KollusPlayerSDK

Ver 2.0.19

2024. 3. 4

[문서화클래스 문서화 xxiii](#_목차000004)

[com.kollus.sdk.media.content.BandwidthItem 클래스 참조 xxiii](#_목차000005)

[상세한 설명 xxiii](#_목차000006)

[생성자 & 소멸자 문서화 xxiii](#_목차000007)

[매개변수 xxiii](#_목차000008)

[멤버 함수 문서화 xxiii](#_목차000009)

[반환값 xxiii](#_목차000010)

[String com.kollus.sdk.media.content.BandwidthItem.getBandwidthName () xxiii](#_목차000011)

[반환값 xxiii](#_목차000012)

[com.kollus.sdk.media.content.KollusContent.ChattingInfo 클래스 참조 24](#_목차000013)

[상세한 설명 24](#_목차000014)

[멤버 데이터 문서화 24](#_목차000015)

[String com.kollus.sdk.media.content.KollusContent.ChattingInfo.chatServer 24](#_목차000016)

[String com.kollus.sdk.media.content.KollusContent.ChattingInfo.helloMessage 24](#_목차000017)

[boolean com.kollus.sdk.media.content.KollusContent.ChattingInfo.isAdmin 24](#_목차000018)

[boolean com.kollus.sdk.media.content.KollusContent.ChattingInfo.isAnonymous 24](#_목차000019)

[boolean com.kollus.sdk.media.content.KollusContent.ChattingInfo.isVisible 24](#_목차000020)

[String com.kollus.sdk.media.content.KollusContent.ChattingInfo.kind 24](#_목차000021)

[String com.kollus.sdk.media.content.KollusContent.ChattingInfo.mainUrl 24](#_목차000022)

[String com.kollus.sdk.media.content.KollusContent.ChattingInfo.photoUrl 24](#_목차000023)

[String com.kollus.sdk.media.content.KollusContent.ChattingInfo.position 25](#_목차000024)

[String com.kollus.sdk.media.content.KollusContent.ChattingInfo.roomId 25](#_목차000025)

[String com.kollus.sdk.media.content.KollusContent.ChattingInfo.userId 25](#_목차000026)

[String com.kollus.sdk.media.content.KollusContent.ChattingInfo.userName 25](#_목차000027)

[com.kollus.sdk.media.content.KollusContent.ContentType Enum Reference 26](#_목차000028)

[상세한 설명 26](#_목차000029)

[멤버 데이터 문서화 26](#_목차000030)

[com.kollus.sdk.media.content.KollusContent.ContentType.KOLLUS\_CONTENT 26](#_목차000031)

[com.kollus.sdk.media.util.CpuInfo 클래스 참조 27](#_목차000032)

[정적 Public 멤버 함수 27](#_목차000033)

[멤버 함수 문서화 27](#_목차000034)

[반환값 27](#_목차000035)

[String com.kollus.sdk.media.util.CpuInfo.getCpuName () 27](#_목차000036)

[반환값 27](#_목차000037)

[Vector< String > com.kollus.sdk.media.util.CpuInfo.getFrequence () 27](#_목차000038)

[반환값 27](#_목차000039)

[int com.kollus.sdk.media.util.CpuInfo.getFrequenceWithInt () 27](#_목차000040)

[반환값 27](#_목차000041)

[static synchronized CpuInfo com.kollus.sdk.media.util.CpuInfo.getInstance ()[static] 27](#_목차000042)

[반환값 27](#_목차000043)

[boolean com.kollus.sdk.media.util.CpuInfo.hasFeature (String feature) 27](#_목차000044)

[매개변수 27](#_목차000045)

[반환값 27](#_목차000046)

[boolean com.kollus.sdk.media.util.CpuInfo.isSupportDevice () 27](#_목차000047)

[반환값 28](#_목차000048)

[com.kollus.sdk.media.util.Utils.DecoderType Enum Reference 29](#_목차000049)

[상세한 설명 29](#_목차000050)

[멤버 데이터 문서화 29](#_목차000051)

[com.kollus.sdk.media.util.Utils.DecoderType.SW\_CODEC 29](#_목차000052)

[com.kollus.sdk.media.util.Utils.DEVICE\_TYPE Enum Reference 30](#_목차000053)

[상세한 설명 30](#_목차000054)

[멤버 데이터 문서화 30](#_목차000055)

[com.kollus.sdk.media.util.Utils.DEVICE\_TYPE.DEVICE\_TABLET 30](#_목차000056)

[com.kollus.sdk.media.util.Utils.DEVICE\_TYPE.DEVICE\_TV 30](#_목차000057)

[com.kollus.sdk.media.KollusStorage.DrmSessionManagerListener 인터페이스 참조 31](#_목차000058)

[com.kollus.sdk.media.EmulatorCheckerListener 인터페이스 참조 32](#_목차000059)

[멤버 함수 문서화 32](#_목차000060)

[void com.kollus.sdk.media.EmulatorCheckerListener.onDetectRooting () 32](#_목차000061)

[void com.kollus.sdk.media.EmulatorCheckerListener.onRunningEmulator () 32](#_목차000062)

[com.kollus.sdk.media.util.ErrorCodes 클래스 참조 33](#_목차000063)

[정적 Public 멤버 함수 33](#_목차000064)

[정적 Public 속성 33](#_목차000065)

[Protected 속성 36](#_목차000066)

[멤버 함수 문서화 36](#_목차000067)

[매개변수 37](#_목차000068)

[반환값 37](#_목차000069)

[static synchronized ErrorCodes com.kollus.sdk.media.util.ErrorCodes.getInstance (Context context)[static] 37](#_목차000070)

[매개변수 37](#_목차000071)

[반환값 37](#_목차000072)

[멤버 데이터 문서화 37](#_목차000073)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ABNORMAL\_DRM\_INFO = -8655[static] 37](#_목차000074)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ABNORMAL\_DRM\_PLAY = -8657[static] 37](#_목차000075)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ABNORMAL\_PLAY\_CALLBACK\_INFO = -8673[static] 37](#_목차000076)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ABNORMAL\_PLAY\_CALLBACK\_INFO\_PLAY = -8675[static] 37](#_목차000077)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ABNORMAL\_UPLOAD\_POSTER\_INFO = -8676[static] 37](#_목차000078)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ACCESS\_DENIED\_FOLDER = -8463[static] 37](#_목차000079)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ACCESS\_DENIED\_MEMCACHE\_SERVER = -8478[static] 37](#_목차000080)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ALREADY\_DOWNLOADED = -1631[static] 37](#_목차000081)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ALREADY\_DOWNLOADING = -8631[static] 38](#_목차000082)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ALTER\_TABLE = -8663[static] 38](#_목차000083)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_BAD\_GATEWAY = -8502[static] 38](#_목차000084)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_BAD\_REQUEST = -8400[static] 38](#_목차000085)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CANCEL\_DOWNLOAD = -8615[static] 38](#_목차000086)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CANNOT\_CALL\_API = -8661[static] 38](#_목차000087)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CHECK\_EXPIRATION\_DATE = -8687[static] 38](#_목차000088)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CHECK\_SETTING\_NOT\_SW\_CODEC = -1104[static] 38](#_목차000089)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CODEC\_DECODE = -1103[static] 38](#_목차000090)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CODEC\_INIT = -1102[static] 38](#_목차000091)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CONTENT\_NOT\_PUBLIC\_CHANNEL = -8474[static] 38](#_목차000092)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CREATE\_DIRECTORY = -8600[static] 38](#_목차000093)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CREATE\_FILE = -8601[static] 38](#_목차000094)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CREATE\_NO\_MEDIA\_FILE = -8647[static] 38](#_목차000095)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CREATE\_TABLE = -8604[static] 38](#_목차000096)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CURLE\_COULDNT\_CONNECT = -8007[static] 39](#_목차000097)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CURLE\_COULDNT\_RESOLVE\_HOST = -8006[static] 39](#_목차000098)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CURLE\_OPERATION\_TIMEDOUT = -8028[static] 39](#_목차000099)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CURLE\_PARTIAL\_FILE = -8018[static] 39](#_목차000100)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CURLE\_RECV\_ERROR = -8056[static] 39](#_목차000101)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DECODE\_HEADER = -8622[static] 39](#_목차000102)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DECRYPT = -8649[static] 39](#_목차000103)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DECRYPT\_COMPANY\_AUTH\_INFO = -8634[static] 39](#_목차000104)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DECRYPT\_CONTENT\_INFO\_LINK = -8632[static] 39](#_목차000105)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DECRYPT\_PLAY\_CALLBACK\_INFO = -8671[static] 39](#_목차000106)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DELETE\_RECORD = -8606[static] 39](#_목차000107)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DIFFERENT\_ES\_KEY = -8477[static] 39](#_목차000108)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DIFFERENT\_EXPIRE\_RESET = -8681[static] 39](#_목차000109)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DIFFERENT\_TIME = -8664[static] 39](#_목차000110)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DUPLICATION\_BLOCK\_MEDIA\_KEY = -8479[static] 40](#_목차000111)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DUPLICATION\_BLOCK\_PROCESS = -8464[static] 40](#_목차000112)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EMPTY\_COMPANY\_AUTH\_INFO = -8651[static] 40](#_목차000113)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EMPTY\_CONTENT\_INFO = -8652[static] 40](#_목차000114)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EMPTY\_PLAY\_CALLBACK\_INFO = -8670[static] 40](#_목차000115)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EMPTY\_PLAYLIST\_INFO = -8802[static] 40](#_목차000116)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EMPTY\_POSTER\_UPLOAD\_FILE\_INFO = -8665[static] 40](#_목차000117)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ENCRYPT\_KEY = -8679[static] 40](#_목차000118)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EXPIRATION\_COUNT = -8644[static] 40](#_목차000119)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EXPIRATION\_DATE = -8643[static] 40](#_목차000120)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EXPIRATION\_PLAY\_TIME = -8683[static] 40](#_목차000121)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EXPIRED\_KEY = -2[static] 40](#_목차000122)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_FORBIDDEN = -8403[static] 40](#_목차000123)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_FORCE\_DELETE = -8680[static] 40](#_목차000124)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_FORCE\_EXPIRED = -8658[static] 41](#_목차000125)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GATEWAY\_TIMEOUT = -8504[static] 41](#_목차000126)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GEO\_RESTRICTION = -8471[static] 41](#_목차000127)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_CONTENT\_LAST\_MODIFIED = -8678[static] 41](#_목차000128)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_CONTENT\_TYPE = -8677[static] 41](#_목차000129)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_CONTENTS\_LENGTH = -8617[static] 41](#_목차000130)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_ENCODE\_LEVEL = -8618[static] 41](#_목차000131)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_MEDIA\_ID = -8668[static] 41](#_목차000132)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_MEDIA\_URL = -8619[static] 41](#_목차000133)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_USER\_IP = -8462[static] 41](#_목차000134)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_HEAVY\_REQUEST = -8469[static] 41](#_목차000135)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_HTTP\_VERSION\_NOT\_SUPPORTED = -8505[static] 41](#_목차000136)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_INSERT\_RECORD = -8607[static] 41](#_목차000137)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_INTERNAL\_SERVER = -8500[static] 41](#_목차000138)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_INTRO\_CANNOT\_DOWNLOAD = -8803[static] 41](#_목차000139)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_INVALID\_KEY = -1[static] 42](#_목차000140)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_INVALID\_MEDIA\_CONTENT\_KEY = -4001[static] 42](#_목차000141)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_LIVE\_NOT\_BROADCASTING = -3012[static] 42](#_목차000142)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_MAX\_REQUEST\_SIZE = -8476[static] 42](#_목차000143)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_MEDIA\_DOMAIN\_VALIDATE = -8468[static] 42](#_목차000144)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_MEDIA\_KEY\_TIME\_EXPIRE = -8467[static] 42](#_목차000145)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_METHOD\_NOT\_ALLOWED = -8405[static] 42](#_목차000146)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_MULTI\_DRM\_EXPIRED = -7000[static] 42](#_목차000147)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NETWORK\_CON\_TIMEOUT = -4105[static] 42](#_목차000148)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NO\_USE\_CONTENT\_URL = -8637[static] 42](#_목차000149)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_ALLOW\_DOWNLOAD = -8650[static] 42](#_목차000150)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_DOWNLOAD\_COMPLETE\_ALL = -8629[static] 42](#_목차000151)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_DOWNLOAD\_TYPE = -8630[static] 42](#_목차000152)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_EXIST\_DOWNLOADED\_CONTENTS = -8638[static] 42](#_목차000153)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_EXIST\_FILE = -8404[static] 43](#_목차000154)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_COMMON\_INFO = -8800[static] 43](#_목차000155)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_DATA = -8620[static] 43](#_목차000156)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_DOMAIN = -8653[static] 43](#_목차000157)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_DRM\_KIND = -8682[static] 43](#_목차000158)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_ID = -8621[static] 43](#_목차000159)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_ITEM\_IN\_PLAYLIST = -8801[static] 43](#_목차000160)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_MOBILE\_LOGO\_FILENAME = -8660[static] 43](#_목차000161)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_POSTER\_URL = -8667[static] 43](#_목차000162)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_RECORD = -8628[static] 43](#_목차000163)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_SNAPSHOT\_FILENAME = -8626[static] 43](#_목차000164)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_SUBTITLE\_FILENAME = -8669[static] 43](#_목차000165)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_THUMBNAIL\_FILENAME = -8627[static] 44](#_목차000166)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_THUMBNAIL\_URL = -8662[static] 44](#_목차000167)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_IMPLEMENTED = -8501[static] 44](#_목차000168)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_LOAD = -8614[static] 44](#_목차000169)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_NORMAL\_REQUEST\_VIDEO\_GATEWAY = -8480[static] 44](#_목차000170)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_RECEIVE\_DOMAIN = -8654[static] 44](#_목차000171)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_RECEIVE\_ENC\_DATA = -8659[static] 44](#_목차000172)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_VAILD\_COMPANY\_AUTH\_INFO = -8636[static] 44](#_목차000173)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_VAILD\_CONTENTS\_INFO = -8646[static] 44](#_목차000174)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_OK = 0[static] 44](#_목차000175)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_OPEN\_DB = -8603[static] 44](#_목차000176)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_OPEN\_FILE = -8608[static] 44](#_목차000177)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_OUT\_OF\_MEMORY = -8609[static] 44](#_목차000178)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PARAM\_VALUE = -8613[static] 45](#_목차000179)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PARSE\_COMPANY\_AUTH\_INFO = -8635[static] 45](#_목차000180)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PARSE\_CONTENT\_INFO\_LINK = -8633[static] 45](#_목차000181)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PARSE\_PLAY\_CALLBACK\_INFO = -8672[static] 45](#_목차000182)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PARSE\_POSTER\_UPLOAD\_FILE\_INFO = -8666[static] 45](#_목차000183)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PLAY\_CALLBACK\_EXPIRATION\_PLAY\_TIME = -8686[static] 45](#_목차000184)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PLAY\_CALLBACK\_EXPIRED = -8674[static] 45](#_목차000185)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PLYMENT = -8470[static] 45](#_목차000186)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PREVIEW\_ERROR = -8472[static] 45](#_목차000187)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_READ\_FILE = -8610[static] 45](#_목차000188)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_REQEST\_URL\_BLOCK = -8466[static] 45](#_목차000189)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_REQUEST\_CONTENT\_URL = -8645[static] 45](#_목차000190)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_REQUEST\_URL = -8461[static] 45](#_목차000191)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_SAVE\_DATA = -8602[static] 46](#_목차000192)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_SELECT\_TABLE = -8605[static] 46](#_목차000193)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_SERVER\_BLACK\_OUT = -1105[static] 46](#_목차000194)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_SERVICE\_UNAVAILABLE = -8503[static] 46](#_목차000195)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_STOAGE\_MANAGER\_BASE = -8000[static] 46](#_목차000196)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_STORAGE\_FULL = -8623[static] 46](#_목차000197)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_TMCM\_INIT = -2001[static] 46](#_목차000198)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UNAUTHORIZED\_1 = -8401[static] 46](#_목차000199)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UNAUTHORIZED\_2 = -8402[static] 46](#_목차000200)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UNDEFINED\_CODE = -1000[static] 46](#_목차000201)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UNKNOWN\_MULTI\_DRM = -2100[static] 46](#_목차000202)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UNSUPPORTED = -1010[static] 46](#_목차000203)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UNSUPPORTED\_DEVICE = -1015[static] 46](#_목차000204)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UPDATE\_RECORD = -8624[static] 46](#_목차000205)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_USED\_MEDIA\_KEY = -8475[static] 46](#_목차000206)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_USER\_KEY = -8473[static] 47](#_목차000207)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_VERIFY\_MEDIA\_KEY = -8465[static] 47](#_목차000208)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_VIDEO\_GATEWAY\_ABNORMAL\_RESPONSE = -8900[static] 47](#_목차000209)

[final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_WRITE\_FILE = -8611[static] 47](#_목차000210)

[final int com.kollus.sdk.media.util.ErrorCodes.INFO\_NETWORK\_TIMEOUT = -8899[static] 47](#_목차000211)

[final int com.kollus.sdk.media.util.ErrorCodes.MEDIA\_ERROR\_IO = -1004[static] 47](#_목차000212)

[final int com.kollus.sdk.media.util.ErrorCodes.MEDIA\_ERROR\_MALFORMED = -1007[static] 47](#_목차000213)

[com.kollus.sdk.media.MediaPlayer.EXTERNAL\_DISPLAY\_MODE Enum Reference 48](#_목차000214)

[상세한 설명 48](#_목차000215)

[멤버 데이터 문서화 48](#_목차000216)

[com.kollus.sdk.media.MediaPlayer.EXTERNAL\_DISPLAY\_MODE.EXTERNAL\_WIFI\_DISPLAY 48](#_목차000217)

[com.kollus.sdk.media.ForensicWatermarkView 클래스 참조 49](#_목차000218)

[Protected 멤버 함수 49](#_목차000219)

[생성자 & 소멸자 문서화 49](#_목차000220)

[매개변수 49](#_목차000221)

[멤버 함수 문서화 49](#_목차000222)

[매개변수 49](#_목차000223)

[com.kollus.sdk.media.content.KollusBookmark 클래스 참조 50](#_목차000224)

[정적 Public 멤버 함수 50](#_목차000225)

[정적 Public 속성 50](#_목차000226)

[생성자 & 소멸자 문서화 50](#_목차000227)

[멤버 함수 문서화 50](#_목차000228)

[매개변수 50](#_목차000229)

[반환값 50](#_목차000230)

[String com.kollus.sdk.media.content.KollusBookmark.getLabel () 50](#_목차000231)

[반환값 50](#_목차000232)

[int com.kollus.sdk.media.content.KollusBookmark.getLevel () 50](#_목차000233)

[반환값 50](#_목차000234)

[Bitmap com.kollus.sdk.media.content.KollusBookmark.getThumbnail () 51](#_목차000235)

[반환값 51](#_목차000236)

[int com.kollus.sdk.media.content.KollusBookmark.getTime () 51](#_목차000237)

[반환값 51](#_목차000238)

[String com.kollus.sdk.media.content.KollusBookmark.getTitle () 51](#_목차000239)

[반환값 51](#_목차000240)

[void com.kollus.sdk.media.content.KollusBookmark.setLabel (String label) 51](#_목차000241)

[매개변수 51](#_목차000242)

[void com.kollus.sdk.media.content.KollusBookmark.setLevel (int level) 51](#_목차000243)

[매개변수 51](#_목차000244)

[void com.kollus.sdk.media.content.KollusBookmark.setThumbnail (Bitmap bm) 51](#_목차000245)

[매개변수 51](#_목차000246)

[void com.kollus.sdk.media.content.KollusBookmark.setTime (int time) 51](#_목차000247)

[매개변수 51](#_목차000248)

[void com.kollus.sdk.media.content.KollusBookmark.setTitle (String title) 51](#_목차000249)

[매개변수 51](#_목차000250)

[멤버 데이터 문서화 51](#_목차000251)

[final int com.kollus.sdk.media.content.KollusBookmark.PROVIDER\_LEVEL = 1[static] 51](#_목차000252)

[final int com.kollus.sdk.media.content.KollusBookmark.USER\_LEVEL = 0[static] 52](#_목차000253)

[com.kollus.sdk.media.content.KollusContent 클래스 참조 53](#_목차000254)

[Public 멤버 함수 53](#_목차000255)

[정적 Public 속성 55](#_목차000256)

[Protected 속성 55](#_목차000257)

[상세한 설명 55](#_목차000258)

[생성자 & 소멸자 문서화 55](#_목차000259)

[com.kollus.sdk.media.content.KollusContent.KollusContent (Parcel in) 55](#_목차000260)

[매개변수 55](#_목차000261)

[멤버 함수 문서화 55](#_목차000262)

[매개변수 55](#_목차000263)

[String com.kollus.sdk.media.content.KollusContent.getAiScriptPath () 55](#_목차000264)

[String com.kollus.sdk.media.content.KollusContent.getAudioWaterMark () 55](#_목차000265)

[반환값 55](#_목차000266)

[int com.kollus.sdk.media.content.KollusContent.getBitrate () 55](#_목차000267)

[반환값 55](#_목차000268)

[String com.kollus.sdk.media.content.KollusContent.getCaptionStyle () 55](#_목차000269)

[반환값 55](#_목차000270)

[ChattingInfo com.kollus.sdk.media.content.KollusContent.getChattingInfo () 56](#_목차000271)

[반환값 56](#_목차000272)

[String com.kollus.sdk.media.content.KollusContent.getCompany () 56](#_목차000273)

[반환값 56](#_목차000274)

[ContentType com.kollus.sdk.media.content.KollusContent.getContentType () 56](#_목차000275)

[반환값 56](#_목차000276)

[참고 56](#_목차000277)

[String com.kollus.sdk.media.content.KollusContent.getCourse () 56](#_목차000278)

[반환값 56](#_목차000279)

[String com.kollus.sdk.media.content.KollusContent.getDetailInfoUrl () 56](#_목차000280)

[반환값 56](#_목차000281)

[boolean com.kollus.sdk.media.content.KollusContent.getDisablePlayRate () 56](#_목차000282)

[반환값 56](#_목차000283)

[boolean com.kollus.sdk.media.content.KollusContent.getDisableTvOut () 56](#_목차000284)

[반환값 56](#_목차000285)

[boolean com.kollus.sdk.media.content.KollusContent.getDownloadError () 56](#_목차000286)

[반환값 56](#_목차000287)

[int com.kollus.sdk.media.content.KollusContent.getDownloadPercent () 56](#_목차000288)

[반환값 56](#_목차000289)

[int com.kollus.sdk.media.content.KollusContent.getDuration () 57](#_목차000290)

[반환값 57](#_목차000291)

[int com.kollus.sdk.media.content.KollusContent.getExpirationCount () 57](#_목차000292)

[반환값 57](#_목차000293)

[int com.kollus.sdk.media.content.KollusContent.getExpirationDate () 57](#_목차000294)

[반환값 57](#_목차000295)

[int com.kollus.sdk.media.content.KollusContent.getExpirationPlaytime () 57](#_목차000296)

[반환값 57](#_목차000297)

[boolean com.kollus.sdk.media.content.KollusContent.getExpirationRefreshPopup () 57](#_목차000298)

[반환값 57](#_목차000299)

[long com.kollus.sdk.media.content.KollusContent.getFileSize () 57](#_목차000300)

[반환값 57](#_목차000301)

[boolean com.kollus.sdk.media.content.KollusContent.getForceNScreen () 57](#_목차000302)

[반환값 57](#_목차000303)

[String com.kollus.sdk.media.content.KollusContent.getForensicWaterMark () 57](#_목차000304)

[반환값 57](#_목차000305)

[String com.kollus.sdk.media.content.KollusContent.getLicenseKey () 57](#_목차000306)

[반환값 57](#_목차000307)

[String com.kollus.sdk.media.content.KollusContent.getLicenseToken () 57](#_목차000308)

[반환값 58](#_목차000309)

[String com.kollus.sdk.media.content.KollusContent.getLicenseUrl () 58](#_목차000310)

[반환값 58](#_목차000311)

[String com.kollus.sdk.media.content.KollusContent.getLocalMediaUrl () 58](#_목차000312)

[반환값 58](#_목차000313)

[int com.kollus.sdk.media.content.KollusContent.getMaxPlaybackRate () 58](#_목차000314)

[반환값 58](#_목차000315)

[String com.kollus.sdk.media.content.KollusContent.getMediaContentKey () 58](#_목차000316)

[반환값 58](#_목차000317)

[String com.kollus.sdk.media.content.KollusContent.getMediaContentKeyMD5 () 58](#_목차000318)

[반환값 58](#_목차000319)

[String com.kollus.sdk.media.content.KollusContent.getMediaUrl () 58](#_목차000320)

[반환값 58](#_목차000321)

[boolean com.kollus.sdk.media.content.KollusContent.getMute () 58](#_목차000322)

[반환값 58](#_목차000323)

[String com.kollus.sdk.media.content.KollusContent.getNextEpisodeCallbackData () 58](#_목차000324)

[반환값 58](#_목차000325)

[String com.kollus.sdk.media.content.KollusContent.getNextEpisodeCallbackUrl () 58](#_목차000326)

[반환값 58](#_목차000327)

[int com.kollus.sdk.media.content.KollusContent.getNextEpisodeShowTime () 59](#_목차000328)

[반환값 59](#_목차000329)

[String com.kollus.sdk.media.content.KollusContent.getOriginThumbnailPath () 59](#_목차000330)

[int com.kollus.sdk.media.content.KollusContent.getPlaySectionEnd () 59](#_목차000331)

[반환값 59](#_목차000332)

[int com.kollus.sdk.media.content.KollusContent.getPlaySectionStart () 59](#_목차000333)

[반환값 59](#_목차000334)

[int com.kollus.sdk.media.content.KollusContent.getPlaytime () 59](#_목차000335)

[반환값 59](#_목차000336)

[long com.kollus.sdk.media.content.KollusContent.getReceivedSize () 59](#_목차000337)

[반환값 59](#_목차000338)

[long com.kollus.sdk.media.content.KollusContent.getReceivingSize () 59](#_목차000339)

[반환값 59](#_목차000340)

[int com.kollus.sdk.media.content.KollusContent.getRepeatPlaySectionEnd () 59](#_목차000341)

[반환값 59](#_목차000342)

[int com.kollus.sdk.media.content.KollusContent.getRepeatPlaySectionStart () 59](#_목차000343)

[반환값 59](#_목차000344)

[String com.kollus.sdk.media.content.KollusContent.getScreenShotPath () 59](#_목차000345)

[반환값 60](#_목차000346)

[boolean com.kollus.sdk.media.content.KollusContent.getSeekable () 60](#_목차000347)

[반환값 60](#_목차000348)

[int com.kollus.sdk.media.content.KollusContent.getSeekableEnd () 60](#_목차000349)

[반환값 60](#_목차000350)

[String com.kollus.sdk.media.content.KollusContent.getServiceProviderMessage () 60](#_목차000351)

[참고 60](#_목차000352)

[반환값 60](#_목차000353)

[String com.kollus.sdk.media.content.KollusContent.getSkinString () 60](#_목차000354)

[반환값 60](#_목차000355)

[int com.kollus.sdk.media.content.KollusContent.getSkipSec () 60](#_목차000356)

[반환값 60](#_목차000357)

[int com.kollus.sdk.media.content.KollusContent.getStartAt () 60](#_목차000358)

[반환값 60](#_목차000359)

[String com.kollus.sdk.media.content.KollusContent.getSubCourse () 60](#_목차000360)

[반환값 60](#_목차000361)

[Vector< SubtitleInfo > com.kollus.sdk.media.content.KollusContent.getSubtitleInfo () 60](#_목차000362)

[반환값 60](#_목차000363)

[Vector< SubtitleInfo > com.kollus.sdk.media.content.KollusContent.getSubtitleSubInfo () 61](#_목차000364)

[반환값 61](#_목차000365)

[String com.kollus.sdk.media.content.KollusContent.getSynopsis () 61](#_목차000366)

[반환값 61](#_목차000367)

[String com.kollus.sdk.media.content.KollusContent.getTeacher () 61](#_목차000368)

[반환값 61](#_목차000369)

[String com.kollus.sdk.media.content.KollusContent.getThumbnailPath () 61](#_목차000370)

[반환값 61](#_목차000371)

[int com.kollus.sdk.media.content.KollusContent.getTotalExpirationCount () 61](#_목차000372)

[반환값 61](#_목차000373)

[int com.kollus.sdk.media.content.KollusContent.getTotalExpirationPlaytime () 61](#_목차000374)

[반환값 61](#_목차000375)

[String com.kollus.sdk.media.content.KollusContent.getUploadFileKey () 61](#_목차000376)

[반환값 61](#_목차000377)

[int com.kollus.sdk.media.content.KollusContent.getUriIndex () 61](#_목차000378)

[반환값 61](#_목차000379)

[int com.kollus.sdk.media.content.KollusContent.getVideoHeight () 61](#_목차000380)

[반환값 61](#_목차000381)

[int com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkAlpha () 61](#_목차000382)

[반환값 62](#_목차000383)

[String com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkCode () 62](#_목차000384)

[반환값 62](#_목차000385)

[int com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkFontColor () 62](#_목차000386)

[반환값 62](#_목차000387)

[int com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkFontSize () 62](#_목차000388)

[반환값 62](#_목차000389)

[int com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkHideTime () 62](#_목차000390)

[반환값 62](#_목차000391)

[int com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkShowTime () 62](#_목차000392)

[반환값 62](#_목차000393)

[int com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkSpace () 62](#_목차000394)

[반환값 62](#_목차000395)

[int com.kollus.sdk.media.content.KollusContent.getVideoWidth () 62](#_목차000396)

[반환값 62](#_목차000397)

[boolean com.kollus.sdk.media.content.KollusContent.hasAudioWaterMark () 62](#_목차000398)

[반환값 62](#_목차000399)

[boolean com.kollus.sdk.media.content.KollusContent.hasForensicWaterMark () 62](#_목차000400)

[반환값 62](#_목차000401)

[boolean com.kollus.sdk.media.content.KollusContent.isAudioFile () 63](#_목차000402)

[반환값 63](#_목차000403)

[boolean com.kollus.sdk.media.content.KollusContent.isCompleted () 63](#_목차000404)

[반환값 63](#_목차000405)

[boolean com.kollus.sdk.media.content.KollusContent.isContentExpirated () 63](#_목차000406)

[반환값 63](#_목차000407)

[boolean com.kollus.sdk.media.content.KollusContent.isDownloadCanceled () 63](#_목차000408)

[반환값 63](#_목차000409)

[boolean com.kollus.sdk.media.content.KollusContent.isDownloading () 63](#_목차000410)

[반환값 63](#_목차000411)

[boolean com.kollus.sdk.media.content.KollusContent.isDrm () 63](#_목차000412)

[반환값 63](#_목차000413)

[boolean com.kollus.sdk.media.content.KollusContent.isIntro () 63](#_목차000414)

[반환값 63](#_목차000415)

[boolean com.kollus.sdk.media.content.KollusContent.IsInvalidFileSize () 63](#_목차000416)

[반환값 63](#_목차000417)

[boolean com.kollus.sdk.media.content.KollusContent.isLive () 63](#_목차000418)

[반환값 63](#_목차000419)

[boolean com.kollus.sdk.media.content.KollusContent.isLoaded () 63](#_목차000420)

[반환값 64](#_목차000421)

[boolean com.kollus.sdk.media.content.KollusContent.isSection () 64](#_목차000422)

[반환값 64](#_목차000423)

[boolean com.kollus.sdk.media.content.KollusContent.isThumbnailDownloadSync () 64](#_목차000424)

[반환값 64](#_목차000425)

[boolean com.kollus.sdk.media.content.KollusContent.isThumbnailEnable () 64](#_목차000426)

[반환값 64](#_목차000427)

[boolean com.kollus.sdk.media.content.KollusContent.isVideoWaterMarkRandom () 64](#_목차000428)

[반환값 64](#_목차000429)

[boolean com.kollus.sdk.media.content.KollusContent.isVmCheck () 64](#_목차000430)

[반환값 64](#_목차000431)

[boolean com.kollus.sdk.media.content.KollusContent.isVr () 64](#_목차000432)

[반환값 64](#_목차000433)

[void com.kollus.sdk.media.content.KollusContent.setByJson (String jsonString) 64](#_목차000434)

[매개변수 64](#_목차000435)

[void com.kollus.sdk.media.content.KollusContent.setContentExpired (boolean expired) 64](#_목차000436)

[매개변수 64](#_목차000437)

[void com.kollus.sdk.media.content.KollusContent.setContentType (ContentType type) 64](#_목차000438)

[매개변수 65](#_목차000439)

[참고 65](#_목차000440)

[void com.kollus.sdk.media.content.KollusContent.setDownloadCanceled () 65](#_목차000441)

[void com.kollus.sdk.media.content.KollusContent.setDownloadCompleted (boolean complete) 65](#_목차000442)

[매개변수 65](#_목차000443)

[void com.kollus.sdk.media.content.KollusContent.setDownloadError (boolean bError) 65](#_목차000444)

[매개변수 65](#_목차000445)

[void com.kollus.sdk.media.content.KollusContent.setDownloading (boolean downloading) 65](#_목차000446)

[매개변수 65](#_목차000447)

[void com.kollus.sdk.media.content.KollusContent.setDownloadPercent (int percent) 65](#_목차000448)

[매개변수 65](#_목차000449)

[void com.kollus.sdk.media.content.KollusContent.setFileSize (long fileSize) 65](#_목차000450)

[void com.kollus.sdk.media.content.KollusContent.setForDownload (Parcel in) 65](#_목차000451)

[매개변수 65](#_목차000452)

[void com.kollus.sdk.media.content.KollusContent.setForPlay (Parcel in) 65](#_목차000453)

[매개변수 65](#_목차000454)

[void com.kollus.sdk.media.content.KollusContent.setLoaded (boolean bLoaded) 65](#_목차000455)

[매개변수 66](#_목차000456)

[void com.kollus.sdk.media.content.KollusContent.setLocalMediaUrl (String mediaUrl) 66](#_목차000457)

[void com.kollus.sdk.media.content.KollusContent.setMediaContentKey (String key) 66](#_목차000458)

[매개변수 66](#_목차000459)

[void com.kollus.sdk.media.content.KollusContent.setPlaytime (int playtime) 66](#_목차000460)

[매개변수 66](#_목차000461)

[void com.kollus.sdk.media.content.KollusContent.setReceivedSize (long receiveSize) 66](#_목차000462)

[매개변수 66](#_목차000463)

[void com.kollus.sdk.media.content.KollusContent.setReceivingSize (long receiveSize) 66](#_목차000464)

[매개변수 66](#_목차000465)

[void com.kollus.sdk.media.content.KollusContent.setServiceProviderMessage (String message) 66](#_목차000466)

[매개변수 66](#_목차000467)

[void com.kollus.sdk.media.content.KollusContent.setUriIndex (int index) 66](#_목차000468)

[매개변수 66](#_목차000469)

[String com.kollus.sdk.media.content.KollusContent.toString () 66](#_목차000470)

[반환값 66](#_목차000471)

[boolean com.kollus.sdk.media.content.KollusContent.useBlockEmulator () 66](#_목차000472)

[반환값 66](#_목차000473)

[멤버 데이터 문서화 67](#_목차000474)

[com.kollus.sdk.media.KollusPlayerBookmarkListener 인터페이스 참조 68](#_목차000475)

[멤버 함수 문서화 68](#_목차000476)

[매개변수 68](#_목차000477)

[void com.kollus.sdk.media.KollusPlayerBookmarkListener.onBookmarkDeleted (int position, boolean bDeleted) 68](#_목차000478)

[매개변수 68](#_목차000479)

[void com.kollus.sdk.media.KollusPlayerBookmarkListener.onBookmarkUpdated (int position, boolean bUpdated) 68](#_목차000480)

[매개변수 68](#_목차000481)

[void com.kollus.sdk.media.KollusPlayerBookmarkListener.onGetBookmarkError (int nErrorCode) 68](#_목차000482)

[com.kollus.sdk.media.KollusPlayerCallbackListener 인터페이스 참조 69](#_목차000483)

[멤버 함수 문서화 69](#_목차000484)

[매개변수 69](#_목차000485)

[com.kollus.sdk.media.KollusPlayerContentMode 클래스 참조 70](#_목차000486)

[멤버 데이터 문서화 70](#_목차000487)

[final int com.kollus.sdk.media.KollusPlayerContentMode.ScaleAspectFillStretch = 2[static] 70](#_목차000488)

[final int com.kollus.sdk.media.KollusPlayerContentMode.ScaleAspectFit = 0[static] 70](#_목차000489)

[final int com.kollus.sdk.media.KollusPlayerContentMode.ScaleCenter = 3[static] 70](#_목차000490)

[final int com.kollus.sdk.media.KollusPlayerContentMode.ScaleZoom = 4[static] 70](#_목차000491)

[com.kollus.sdk.media.KollusPlayerDRMListener 인터페이스 참조 71](#_목차000492)

[정적 Public 속성 71](#_목차000493)

[멤버 함수 문서화 71](#_목차000494)

[매개변수 71](#_목차000495)

[void com.kollus.sdk.media.KollusPlayerDRMListener.onDRMInfo (KollusContent content, int nInfoCode) 71](#_목차000496)

[매개변수 71](#_목차000497)

[멤버 데이터 문서화 71](#_목차000498)

[final int com.kollus.sdk.media.KollusPlayerDRMListener.DCB\_INFO\_EXPIRED = 1[static] 71](#_목차000499)

[final int com.kollus.sdk.media.KollusPlayerDRMListener.DCB\_INFO\_RESET = 2[static] 71](#_목차000500)

[com.kollus.sdk.media.KollusPlayerDRMUpdateListener 인터페이스 참조 72](#_목차000501)

[멤버 함수 문서화 72](#_목차000502)

[void com.kollus.sdk.media.KollusPlayerDRMUpdateListener.onDRMUpdateProcess (String request, String response) 72](#_목차000503)

[매개변수 72](#_목차000504)

[void com.kollus.sdk.media.KollusPlayerDRMUpdateListener.onDRMUpdateStart () 72](#_목차000505)

[com.kollus.sdk.media.KollusPlayerLMSListener 인터페이스 참조 73](#_목차000506)

[멤버 함수 문서화 73](#_목차000507)

[매개변수 73](#_목차000508)

[com.kollus.sdk.media.KollusPlayerThumbnailListener 인터페이스 참조 74](#_목차000509)

[멤버 함수 문서화 74](#_목차000510)

[매개변수 74](#_목차000511)

[com.kollus.sdk.media.KollusStorage 클래스 참조 75](#_목차000512)

[Public 멤버 함수 75](#_목차000513)

[정적 Public 멤버 함수 75](#_목차000514)

[정적 Public 속성 75](#_목차000515)

[Protected 멤버 함수 76](#_목차000516)

[멤버 함수 문서화 76](#_목차000517)

[int com.kollus.sdk.media.KollusStorage.checkLiveStream (String url) 76](#_목차000518)

[매개변수 76](#_목차000519)

[반환값 76](#_목차000520)

[String com.kollus.sdk.media.KollusStorage.checkVersion () 76](#_목차000521)

[반환값 76](#_목차000522)

[void com.kollus.sdk.media.KollusStorage.clearCache () 76](#_목차000523)

[int com.kollus.sdk.media.KollusStorage.download (String mediaContentKey) 76](#_목차000524)

[매개변수 76](#_목차000525)

[참고 76](#_목차000526)

[반환값 76](#_목차000527)

[void com.kollus.sdk.media.KollusStorage.finish () 76](#_목차000528)

[ArrayList< KollusContent > com.kollus.sdk.media.KollusStorage.getDownloadContentList () 76](#_목차000529)

[반환값 76](#_목차000530)

[KollusContent com.kollus.sdk.media.KollusStorage.getDownloadKollusContent (String path) 77](#_목차000531)

[반환값 77](#_목차000532)

[static synchronized KollusStorage com.kollus.sdk.media.KollusStorage.getInstance (Context context)[static] 77](#_목차000533)

[매개변수 77](#_목차000534)

[반환값 77](#_목차000535)

[boolean com.kollus.sdk.media.KollusStorage.getKollusContent (KollusContent content, String mediaContentKey) throws IllegalStateException 77](#_목차000536)

[반환값 77](#_목차000537)

[String com.kollus.sdk.media.KollusStorage.getLastError () 77](#_목차000538)

[반환값 77](#_목차000539)

[String com.kollus.sdk.media.KollusStorage.getMediaInfoJson (Uri uri)[protected] 77](#_목차000540)

[매개변수 77](#_목차000541)

[반환값 77](#_목차000542)

[long com.kollus.sdk.media.KollusStorage.getNativeInstance () 77](#_목차000543)

[반환값 77](#_목차000544)

[String com.kollus.sdk.media.KollusStorage.getRootPath () 77](#_목차000545)

[반환값 77](#_목차000546)

[long com.kollus.sdk.media.KollusStorage.getUsedSize (int type) 77](#_목차000547)

[매개변수 77](#_목차000548)

[반환값 77](#_목차000549)

[String com.kollus.sdk.media.KollusStorage.getVersion () 78](#_목차000550)

[반환값 78](#_목차000551)

[int com.kollus.sdk.media.KollusStorage.initialize (String key, String expireDate, String packageName) 78](#_목차000552)

[매개변수 78](#_목차000553)

[boolean com.kollus.sdk.media.KollusStorage.isReady () 78](#_목차000554)

[int com.kollus.sdk.media.KollusStorage.load (String path, String extraDrmParam, KollusContent content) 78](#_목차000555)

[매개변수 78](#_목차000556)

[참고 78](#_목차000557)

[반환값 78](#_목차000558)

[void com.kollus.sdk.media.KollusStorage.registerKollusPlayerCallbackListener (KollusPlayerCallbackListener listener) 78](#_목차000559)

[매개변수 78](#_목차000560)

[void com.kollus.sdk.media.KollusStorage.registerKollusPlayerDRMListener (KollusPlayerDRMListener listener) 78](#_목차000561)

[매개변수 78](#_목차000562)

[void com.kollus.sdk.media.KollusStorage.registerKollusPlayerLMSListener (KollusPlayerLMSListener listener)[protected] 78](#_목차000563)

[매개변수 78](#_목차000564)

[void com.kollus.sdk.media.KollusStorage.registerKollusPlayerThumbnailListener (KollusPlayerThumbnailListener listener) 78](#_목차000565)

[매개변수 79](#_목차000566)

[void com.kollus.sdk.media.KollusStorage.registerKollusStorageListener (OnKollusStorageListener listener) 79](#_목차000567)

[매개변수 79](#_목차000568)

[synchronized void com.kollus.sdk.media.KollusStorage.releaseInstance () 79](#_목차000569)

[int com.kollus.sdk.media.KollusStorage.remove (String mediaContentKey) 79](#_목차000570)

[매개변수 79](#_목차000571)

[참고 79](#_목차000572)

[반환값 79](#_목차000573)

[void com.kollus.sdk.media.KollusStorage.sendStoredLMS (StoredLMSListener listener) 79](#_목차000574)

[매개변수 79](#_목차000575)

[void com.kollus.sdk.media.KollusStorage.setCacheSize (int cacheSizeMB) 79](#_목차000576)

[매개변수 79](#_목차000577)

[int com.kollus.sdk.media.KollusStorage.setDevice (String storagePath, String playerId, String playerIdMd5, boolean bTablet) 79](#_목차000578)

[매개변수 79](#_목차000579)

[void com.kollus.sdk.media.KollusStorage.setDeviceASync (String storagePath, String playerId, String playerIdMd5, boolean bTablet, OnKollusStorageDeviceListener listener) 79](#_목차000580)

[매개변수 79](#_목차000581)

[void com.kollus.sdk.media.KollusStorage.setNetworkTimeout (int timeout, int retryCount) 80](#_목차000582)

[매개변수 80](#_목차000583)

[int com.kollus.sdk.media.KollusStorage.unload (String mediaContentKey) 80](#_목차000584)

[매개변수 80](#_목차000585)

[참고 80](#_목차000586)

[반환값 80](#_목차000587)

[void com.kollus.sdk.media.KollusStorage.unregisterKollusPlayerCallbackListener (KollusPlayerCallbackListener listener) 80](#_목차000588)

[매개변수 80](#_목차000589)

[void com.kollus.sdk.media.KollusStorage.unregisterKollusPlayerDRMListener (KollusPlayerDRMListener listener) 80](#_목차000590)

[매개변수 80](#_목차000591)

[void com.kollus.sdk.media.KollusStorage.unregisterKollusPlayerLMSListener (KollusPlayerLMSListener listener)[protected] 80](#_목차000592)

[매개변수 80](#_목차000593)

[void com.kollus.sdk.media.KollusStorage.unregisterKollusPlayerThumbnailListener (KollusPlayerThumbnailListener listener) 80](#_목차000594)

[매개변수 80](#_목차000595)

[void com.kollus.sdk.media.KollusStorage.unregisterKollusStorageListener (OnKollusStorageListener listener) 80](#_목차000596)

[매개변수 81](#_목차000597)

[void com.kollus.sdk.media.KollusStorage.updateDownloadDRMInfo (KollusPlayerDRMUpdateListener listener, ArrayList< String > mckList) 81](#_목차000598)

[매개변수 81](#_목차000599)

[void com.kollus.sdk.media.KollusStorage.updateDownloadDRMInfo (KollusPlayerDRMUpdateListener listener, boolean bAll) 81](#_목차000600)

[매개변수 81](#_목차000601)

[멤버 데이터 문서화 81](#_목차000602)

[final int com.kollus.sdk.media.KollusStorage.TYPE\_DOWNLOAD = 1[static] 81](#_목차000603)

[com.kollus.sdk.media.MediaPlayer 클래스 참조 82](#_목차000604)

[Public 멤버 함수 82](#_목차000605)

[정적 Public 속성 83](#_목차000606)

[Protected 멤버 함수 84](#_목차000607)

[생성자 & 소멸자 문서화 84](#_목차000608)

[매개변수 84](#_목차000609)

[멤버 함수 문서화 84](#_목차000610)

[매개변수 84](#_목차000611)

[참고 84](#_목차000612)

[void com.kollus.sdk.media.MediaPlayer.addTimedTextSource (Context context, boolean bSubTimedText, Uri uri) throws IOException, IllegalArgumentException, IllegalStateException 84](#_목차000613)

[매개변수 84](#_목차000614)

[void com.kollus.sdk.media.MediaPlayer.deleteKollusBookmark (int position) throws IllegalStateException 84](#_목차000615)

[매개변수 84](#_목차000616)

[void com.kollus.sdk.media.MediaPlayer.deselectTrack (boolean bSubTimedText, int index) throws IllegalStateException 84](#_목차000617)

[매개변수 84](#_목차000618)

[void com.kollus.sdk.media.MediaPlayer.destroyDisplay () 84](#_목차000619)

[int com.kollus.sdk.media.MediaPlayer.getBandwidth () 85](#_목차000620)

[반환값 85](#_목차000621)

[int com.kollus.sdk.media.MediaPlayer.getCachedDuration () 85](#_목차000622)

[int com.kollus.sdk.media.MediaPlayer.getCurrentPosition () 85](#_목차000623)

[반환값 85](#_목차000624)

[int com.kollus.sdk.media.MediaPlayer.getDownloadRate () 85](#_목차000625)

[반환값 85](#_목차000626)

[int com.kollus.sdk.media.MediaPlayer.getDuration () 85](#_목차000627)

[반환값 85](#_목차000628)

[String com.kollus.sdk.media.MediaPlayer.getErrorString (int errorCode) 85](#_목차000629)

[매개변수 85](#_목차000630)

[반환값 85](#_목차000631)

[boolean com.kollus.sdk.media.MediaPlayer.getKollusContent (KollusContent content) throws IllegalStateException 85](#_목차000632)

[매개변수 85](#_목차000633)

[반환값 85](#_목차000634)

[int com.kollus.sdk.media.MediaPlayer.getPlayAt () 85](#_목차000635)

[반환값 85](#_목차000636)

[String com.kollus.sdk.media.MediaPlayer.getPlayerName () 85](#_목차000637)

[반환값 85](#_목차000638)

[int com.kollus.sdk.media.MediaPlayer.getPlayerType () 86](#_목차000639)

[반환값 86](#_목차000640)

[MediaPlayerBase.TrackInfo[] com.kollus.sdk.media.MediaPlayer.getTrackInfo () throws IllegalStateException 86](#_목차000641)

[String com.kollus.sdk.media.MediaPlayer.getVersion () 86](#_목차000642)

[반환값 86](#_목차000643)

[String com.kollus.sdk.media.MediaPlayer.getVideoCodecName () 86](#_목차000644)

[반환값 86](#_목차000645)

[int com.kollus.sdk.media.MediaPlayer.getVideoHeight () 86](#_목차000646)

[반환값 86](#_목차000647)

[int com.kollus.sdk.media.MediaPlayer.getVideoWidth () 86](#_목차000648)

[반환값 86](#_목차000649)

[boolean com.kollus.sdk.media.MediaPlayer.isAiPlaybackRateEnabled () 86](#_목차000650)

[반환값 86](#_목차000651)

[boolean com.kollus.sdk.media.MediaPlayer.isLooping () 86](#_목차000652)

[반환값 86](#_목차000653)

[boolean com.kollus.sdk.media.MediaPlayer.isPlaying () 86](#_목차000654)

[반환값 86](#_목차000655)

[void com.kollus.sdk.media.MediaPlayer.pause () throws IllegalStateException 86](#_목차000656)

[void com.kollus.sdk.media.MediaPlayer.prepareAsync () 87](#_목차000657)

[void com.kollus.sdk.media.MediaPlayer.release () 87](#_목차000658)

[void com.kollus.sdk.media.MediaPlayer.releaseWhenFinishing () 87](#_목차000659)

[void com.kollus.sdk.media.MediaPlayer.seekTo (int msec) throws IllegalStateException 87](#_목차000660)

[void com.kollus.sdk.media.MediaPlayer.seekToExact (int msec) throws IllegalStateException 87](#_목차000661)

[void com.kollus.sdk.media.MediaPlayer.selectTrack (boolean bSubTimedText, int index) throws IllegalStateException 87](#_목차000662)

[매개변수 87](#_목차000663)

[void com.kollus.sdk.media.MediaPlayer.setAiPlaybackRateEnable (boolean enable) 87](#_목차000664)

[매개변수 87](#_목차000665)

[void com.kollus.sdk.media.MediaPlayer.setAudioDelay (int timeMs) 87](#_목차000666)

[매개변수 87](#_목차000667)

[void com.kollus.sdk.media.MediaPlayer.setBandwidth (int bandwidth) 87](#_목차000668)

[매개변수 87](#_목차000669)

[void com.kollus.sdk.media.MediaPlayer.setBandwidthName (String bandwidthName) 87](#_목차000670)

[매개변수 87](#_목차000671)

[boolean com.kollus.sdk.media.MediaPlayer.setBufferingRatio (int ratio) 87](#_목차000672)

[매개변수 87](#_목차000673)

[반환값 87](#_목차000674)

[void com.kollus.sdk.media.MediaPlayer.setCaptureDetectLister (CaptureDetectLister listener) 88](#_목차000675)

[매개변수 88](#_목차000676)

[void com.kollus.sdk.media.MediaPlayer.setDataSourceByKey (String mediaContentKey, String extraDrmParam) 88](#_목차000677)

[매개변수 88](#_목차000678)

[void com.kollus.sdk.media.MediaPlayer.setDataSourceByUrl (String path, String extraDrmParam) 88](#_목차000679)

[매개변수 88](#_목차000680)

[void com.kollus.sdk.media.MediaPlayer.setDisplay (SurfaceHolder sh) 88](#_목차000681)

[매개변수 88](#_목차000682)

[void com.kollus.sdk.media.MediaPlayer.setEmulatorCheckerListener (EmulatorCheckerListener listener) 88](#_목차000683)

[매개변수 88](#_목차000684)

[void com.kollus.sdk.media.MediaPlayer.setForensicWatermarkView (ForensicWatermarkView view) 88](#_목차000685)

[매개변수 88](#_목차000686)

[void com.kollus.sdk.media.MediaPlayer.setInitialBandwidth (int bandwidth) 88](#_목차000687)

[매개변수 88](#_목차000688)

[void com.kollus.sdk.media.MediaPlayer.setKollusPlayerBookmarkListener (KollusPlayerBookmarkListener listener) 88](#_목차000689)

[매개변수 89](#_목차000690)

[void com.kollus.sdk.media.MediaPlayer.setKollusPlayerLMSListener (KollusPlayerLMSListener listener) 89](#_목차000691)

[매개변수 89](#_목차000692)

[void com.kollus.sdk.media.MediaPlayer.setKollusStorage (KollusStorage storage)[protected] 89](#_목차000693)

[매개변수 89](#_목차000694)

[void com.kollus.sdk.media.MediaPlayer.setLastOffsetTime (int lastOffsetTimeMs) 89](#_목차000695)

[매개변수 89](#_목차000696)

[void com.kollus.sdk.media.MediaPlayer.setLmsOffDownloadContent (boolean bOn) 89](#_목차000697)

[매개변수 89](#_목차000698)

[void com.kollus.sdk.media.MediaPlayer.setLooping (boolean looping) 89](#_목차000699)

[매개변수 89](#_목차000700)

[void com.kollus.sdk.media.MediaPlayer.setMinDurationForQualityIncreaseMs (int msec) 89](#_목차000701)

[매개변수 89](#_목차000702)

[void com.kollus.sdk.media.MediaPlayer.setMute (boolean mute) 89](#_목차000703)

[매개변수 89](#_목차000704)

[boolean com.kollus.sdk.media.MediaPlayer.setNetworkTimeout (int timeout) 89](#_목차000705)

[매개변수 90](#_목차000706)

[void com.kollus.sdk.media.MediaPlayer.setNotifyLastReport (boolean bOn) 90](#_목차000707)

[매개변수 90](#_목차000708)

[void com.kollus.sdk.media.MediaPlayer.setOnBufferingUpdateListener (OnBufferingUpdateListener listener) 90](#_목차000709)

[매개변수 90](#_목차000710)

[void com.kollus.sdk.media.MediaPlayer.setOnCencDrmListener (OnCencDrmListener listener) 90](#_목차000711)

[매개변수 90](#_목차000712)

[void com.kollus.sdk.media.MediaPlayer.setOnCompletionListener (OnCompletionListener listener) 90](#_목차000713)

[매개변수 90](#_목차000714)

[void com.kollus.sdk.media.MediaPlayer.setOnErrorListener (OnErrorListener listener) 90](#_목차000715)

[매개변수 90](#_목차000716)

[void com.kollus.sdk.media.MediaPlayer.setOnExternalDisplayDetectListener (OnExternalDisplayDetectListener listener) 90](#_목차000717)

[매개변수 90](#_목차000718)

[void com.kollus.sdk.media.MediaPlayer.setOnInfoListener (OnInfoListener listener) 90](#_목차000719)

[매개변수 90](#_목차000720)

[void com.kollus.sdk.media.MediaPlayer.setOnMediaRouteInfoListener (OnMediaRouteInfoListener listener) 90](#_목차000721)

[매개변수 91](#_목차000722)

[void com.kollus.sdk.media.MediaPlayer.setOnPreparedListener (OnPreparedListener listener) 91](#_목차000723)

[매개변수 91](#_목차000724)

[void com.kollus.sdk.media.MediaPlayer.setOnSeekCompleteListener (OnSeekCompleteListener listener) 91](#_목차000725)

[매개변수 91](#_목차000726)

[void com.kollus.sdk.media.MediaPlayer.setOnTimedTextDetectListener (OnTimedTextDetectListener listener) 91](#_목차000727)

[매개변수 91](#_목차000728)

[void com.kollus.sdk.media.MediaPlayer.setOnTimedTextListener (OnTimedTextListener listener) 91](#_목차000729)

[매개변수 91](#_목차000730)

[void com.kollus.sdk.media.MediaPlayer.setOnVideoSizeChangedListener (OnVideoSizeChangedListener listener) 91](#_목차000731)

[매개변수 91](#_목차000732)

[boolean com.kollus.sdk.media.MediaPlayer.setPlayingRate (float playing\_rate) throws IllegalStateException 91](#_목차000733)

[매개변수 91](#_목차000734)

[반환값 91](#_목차000735)

[void com.kollus.sdk.media.MediaPlayer.setRemoteMediaRouteInfo (String castNameSpace) 91](#_목차000736)

[매개변수 91](#_목차000737)

[void com.kollus.sdk.media.MediaPlayer.setScreenOnWhilePlaying (boolean screenOn) 91](#_목차000738)

[매개변수 92](#_목차000739)

[void com.kollus.sdk.media.MediaPlayer.setSurface (Surface surface) 92](#_목차000740)

[매개변수 92](#_목차000741)

[void com.kollus.sdk.media.MediaPlayer.setVideoRending (boolean bRendering) 92](#_목차000742)

[매개변수 92](#_목차000743)

[void com.kollus.sdk.media.MediaPlayer.setVolumeLevel (int level) 92](#_목차000744)

[매개변수 92](#_목차000745)

[void com.kollus.sdk.media.MediaPlayer.skip () 92](#_목차000746)

[void com.kollus.sdk.media.MediaPlayer.start () throws IllegalStateException 92](#_목차000747)

[void com.kollus.sdk.media.MediaPlayer.stop () throws IllegalStateException 92](#_목차000748)

[boolean com.kollus.sdk.media.MediaPlayer.supportPlaybackrateControl () 92](#_목차000749)

[반환값 92](#_목차000750)

[void com.kollus.sdk.media.MediaPlayer.updateKollusBookmark (int position, String label) throws IllegalStateException 92](#_목차000751)

[매개변수 92](#_목차000752)

[멤버 데이터 문서화 92](#_목차000753)

[참고 92](#_목차000754)

[final int com.kollus.sdk.media.MediaPlayer.MEDIA\_ERROR\_UNKNOWN = 1[static] 93](#_목차000755)

[참고 93](#_목차000756)

[final int com.kollus.sdk.media.MediaPlayer.MEDIA\_INFO\_FRAME\_RATE = 7[static] 93](#_목차000757)

[참고 93](#_목차000758)

[final int com.kollus.sdk.media.MediaPlayer.RENDER\_16\_MULTIPLE = 4[static] 93](#_목차000759)

[final int com.kollus.sdk.media.MediaPlayer.RENDER\_2\_MULTIPLE = 2[static] 93](#_목차000760)

[final int com.kollus.sdk.media.MediaPlayer.RENDER\_2\_POWER = 1[static] 93](#_목차000761)

[final int com.kollus.sdk.media.MediaPlayer.RENDER\_32\_MULTIPLE = 5[static] 93](#_목차000762)

[final int com.kollus.sdk.media.MediaPlayer.RENDER\_4\_MULTIPLE = 3[static] 93](#_목차000763)

[final int com.kollus.sdk.media.MediaPlayer.RENDER\_MODEL = 0[static] 93](#_목차000764)

[com.kollus.sdk.media.MediaPlayer.MediaRouteState Enum Reference 94](#_목차000765)

[상세한 설명 94](#_목차000766)

[com.kollus.sdk.media.MediaPlayer.OnBufferingUpdateListener 인터페이스 참조 95](#_목차000767)

[상세한 설명 95](#_목차000768)

[com.kollus.sdk.media.MediaPlayer.OnCencDrmListener 인터페이스 참조 96](#_목차000769)

[상세한 설명 96](#_목차000770)

[com.kollus.sdk.media.MediaPlayer.OnCompletionListener 인터페이스 참조 97](#_목차000771)

[상세한 설명 97](#_목차000772)

[멤버 함수 문서화 97](#_목차000773)

[매개변수 97](#_목차000774)

[com.kollus.sdk.media.MediaPlayer.OnErrorListener 인터페이스 참조 98](#_목차000775)

[상세한 설명 98](#_목차000776)

[멤버 함수 문서화 98](#_목차000777)

[매개변수 98](#_목차000778)

[반환값 98](#_목차000779)

[com.kollus.sdk.media.MediaPlayer.OnExternalDisplayDetectListener 인터페이스 참조 99](#_목차000780)

[상세한 설명 99](#_목차000781)

[멤버 함수 문서화 99](#_목차000782)

[매개변수 99](#_목차000783)

[com.kollus.sdk.media.MediaPlayer.OnInfoListener 인터페이스 참조 100](#_목차000784)

[상세한 설명 100](#_목차000785)

[멤버 함수 문서화 100](#_목차000786)

[매개변수 100](#_목차000787)

[void com.kollus.sdk.media.MediaPlayer.OnInfoListener.onBufferingStart (MediaPlayer mp) 100](#_목차000788)

[매개변수 100](#_목차000789)

[void com.kollus.sdk.media.MediaPlayer.OnInfoListener.onChangedBandwidth (MediaPlayer mp, BandwidthItem item) 100](#_목차000790)

[매개변수 100](#_목차000791)

[void com.kollus.sdk.media.MediaPlayer.OnInfoListener.onCodecInitFail (MediaPlayer mp, String componentName) 100](#_목차000792)

[매개변수 100](#_목차000793)

[void com.kollus.sdk.media.MediaPlayer.OnInfoListener.onDetectBandwidthList (MediaPlayer mp, List< BandwidthItem > list) 100](#_목차000794)

[매개변수 101](#_목차000795)

[void com.kollus.sdk.media.MediaPlayer.OnInfoListener.onDownloadRate (MediaPlayer mp, int downloadRate) 101](#_목차000796)

[매개변수 101](#_목차000797)

[void com.kollus.sdk.media.MediaPlayer.OnInfoListener.onFrameDrop (MediaPlayer mp) 101](#_목차000798)

[매개변수 101](#_목차000799)

[boolean com.kollus.sdk.media.MediaPlayer.OnInfoListener.onInfo (MediaPlayer mp, int what, int extra) 101](#_목차000800)

[매개변수 101](#_목차000801)

[com.kollus.sdk.media.KollusStorage.OnKollusStorageDeviceListener 인터페이스 참조 102](#_목차000802)

[상세한 설명 102](#_목차000803)

[멤버 함수 문서화 102](#_목차000804)

[void com.kollus.sdk.media.KollusStorage.OnKollusStorageDeviceListener.onDeviceSettingError (int nErrorCode) 102](#_목차000805)

[void com.kollus.sdk.media.KollusStorage.OnKollusStorageDeviceListener.onDeviceSettingStart () 102](#_목차000806)

[com.kollus.sdk.media.KollusStorage.OnKollusStorageListener 인터페이스 참조 103](#_목차000807)

[상세한 설명 103](#_목차000808)

[멤버 함수 문서화 103](#_목차000809)

[매개변수 103](#_목차000810)

[void com.kollus.sdk.media.KollusStorage.OnKollusStorageListener.onError (KollusContent content, int errorCode) 103](#_목차000811)

[매개변수 103](#_목차000812)

[void com.kollus.sdk.media.KollusStorage.OnKollusStorageListener.onProgress (KollusContent content) 103](#_목차000813)

[매개변수 103](#_목차000814)

[com.kollus.sdk.media.MediaPlayer.OnMediaRouteInfoListener 인터페이스 참조 104](#_목차000815)

[상세한 설명 104](#_목차000816)

[com.kollus.sdk.media.MediaPlayer.OnPreparedListener 인터페이스 참조 105](#_목차000817)

[상세한 설명 105](#_목차000818)

[매개변수 105](#_목차000819)

[반환값 105](#_목차000820)

[매개변수 105](#_목차000821)

[반환값 105](#_목차000822)

[멤버 함수 문서화 105](#_목차000823)

[매개변수 105](#_목차000824)

[com.kollus.sdk.media.MediaPlayer.OnSeekCompleteListener 인터페이스 참조 106](#_목차000825)

[상세한 설명 106](#_목차000826)

[멤버 함수 문서화 106](#_목차000827)

[매개변수 106](#_목차000828)

[com.kollus.sdk.media.MediaPlayer.OnTimedTextDetectListener 인터페이스 참조 107](#_목차000829)

[상세한 설명 107](#_목차000830)

[멤버 함수 문서화 107](#_목차000831)

[매개변수 107](#_목차000832)

[com.kollus.sdk.media.MediaPlayer.OnTimedTextListener 인터페이스 참조 108](#_목차000833)

[상세한 설명 108](#_목차000834)

[멤버 함수 문서화 108](#_목차000835)

[매개변수 108](#_목차000836)

[void com.kollus.sdk.media.MediaPlayer.OnTimedTextListener.onTimedImage (MediaPlayer mp, byte[] image, int width, int height) 108](#_목차000837)

[매개변수 108](#_목차000838)

[void com.kollus.sdk.media.MediaPlayer.OnTimedTextListener.onTimedText (MediaPlayer mp, String text) 108](#_목차000839)

[매개변수 108](#_목차000840)

[com.kollus.sdk.media.MediaPlayer.OnVideoSizeChangedListener 인터페이스 참조 109](#_목차000841)

[상세한 설명 109](#_목차000842)

[멤버 함수 문서화 109](#_목차000843)

[매개변수 109](#_목차000844)

[com.kollus.sdk.media.MediaPlayer.RENDER\_MODE Enum Reference 110](#_목차000845)

[상세한 설명 110](#_목차000846)

[멤버 데이터 문서화 110](#_목차000847)

[com.kollus.sdk.media.MediaPlayer.RENDER\_MODE.RENDER\_2\_MULTIPLE 110](#_목차000848)

[com.kollus.sdk.media.MediaPlayer.RENDER\_MODE.RENDER\_2\_POWER 110](#_목차000849)

[com.kollus.sdk.media.MediaPlayer.RENDER\_MODE.RENDER\_32\_MULTIPLE 110](#_목차000850)

[com.kollus.sdk.media.MediaPlayer.RENDER\_MODE.RENDER\_4\_MULTIPLE 110](#_목차000851)

[com.kollus.sdk.media.MediaPlayer.RENDER\_MODE.RENDER\_MODEL 110](#_목차000852)

[com.kollus.sdk.media.content.KollusContent.SubtitleInfo 클래스 참조 111](#_목차000853)

[상세한 설명 111](#_목차000854)

[멤버 데이터 문서화 111](#_목차000855)

[String com.kollus.sdk.media.content.KollusContent.SubtitleInfo.name 111](#_목차000856)

[String com.kollus.sdk.media.content.KollusContent.SubtitleInfo.url 111](#_목차000857)

[com.kollus.sdk.media.util.Utils 클래스 참조 112](#_목차000858)

[정적 Public 멤버 함수 112](#_목차000859)

[정적 Public 속성 112](#_목차000860)

[멤버 함수 문서화 112](#_목차000861)

[매개변수 113](#_목차000862)

[static long com.kollus.sdk.media.util.Utils.getAvailableMemorySize (String path)[static] 113](#_목차000863)

[매개변수 113](#_목차000864)

[반환값 113](#_목차000865)

[static DecoderType com.kollus.sdk.media.util.Utils.getDecoderType (Context context)[static] 113](#_목차000866)

[매개변수 113](#_목차000867)

[반환값 113](#_목차000868)

[참고 113](#_목차000869)

[static DEVICE\_TYPE com.kollus.sdk.media.util.Utils.getDeviceType (Context context)[static] 113](#_목차000870)

[매개변수 113](#_목차000871)

[참고 113](#_목차000872)

[반환값 113](#_목차000873)

[static String com.kollus.sdk.media.util.Utils.getDirectoryJSON (Context context, String prefKey)[static] 113](#_목차000874)

[매개변수 113](#_목차000875)

[반환값 113](#_목차000876)

[static String com.kollus.sdk.media.util.Utils.getDirectoryJSONByPath (Context context, String path, String prefKey)[static] 113](#_목차000877)

[매개변수 113](#_목차000878)

[반환값 113](#_목차000879)

[static String com.kollus.sdk.media.util.Utils.getFileMD5 (File updateFile)[static] 114](#_목차000880)

[매개변수 114](#_목차000881)

[반환값 114](#_목차000882)

[static long com.kollus.sdk.media.util.Utils.getFolderSize (File directory)[static] 114](#_목차000883)

[매개변수 114](#_목차000884)

[반환값 114](#_목차000885)

[static String com.kollus.sdk.media.util.Utils.getPlayerId (Context context)[static] 114](#_목차000886)

[매개변수 114](#_목차000887)

[반환값 114](#_목차000888)

[static String com.kollus.sdk.media.util.Utils.getPlayerIdMd5 (Context context)[static] 114](#_목차000889)

[매개변수 114](#_목차000890)

[반환값 114](#_목차000891)

[static String com.kollus.sdk.media.util.Utils.getPlayerIdSha1 (Context context)[static] 114](#_목차000892)

[매개변수 114](#_목차000893)

[반환값 114](#_목차000894)

[static int com.kollus.sdk.media.util.Utils.getPlayerType ()[static] 114](#_목차000895)

[반환값 114](#_목차000896)

[static int com.kollus.sdk.media.util.Utils.getRenderType (Context context)[static] 114](#_목차000897)

[매개변수 114](#_목차000898)

[반환값 114](#_목차000899)

[static String com.kollus.sdk.media.util.Utils.getStoragePath (Context context)[static] 115](#_목차000900)

[매개변수 115](#_목차000901)

[반환값 115](#_목차000902)

[static String com.kollus.sdk.media.util.Utils.getString (JSONObject json, String key)[static] 115](#_목차000903)

[매개변수 115](#_목차000904)

[반환값 115](#_목차000905)

[static long com.kollus.sdk.media.util.Utils.getTotalMemorySize (String path)[static] 115](#_목차000906)

[매개변수 115](#_목차000907)

[반환값 115](#_목차000908)

[static Object com.kollus.sdk.media.util.Utils.invoke (Object obj, String methodName, Object[] objList)[static] 115](#_목차000909)

[매개변수 115](#_목차000910)

[반환값 115](#_목차000911)

[static boolean com.kollus.sdk.media.util.Utils.isRooting ()[static] 115](#_목차000912)

[반환값 115](#_목차000913)

[static boolean com.kollus.sdk.media.util.Utils.makeDirectory (String path)[static] 115](#_목차000914)

[매개변수 115](#_목차000915)

[반환값 115](#_목차000916)

[static String com.kollus.sdk.media.util.Utils.makeMD5 (String param)[static] 115](#_목차000917)

[매개변수 116](#_목차000918)

[반환값 116](#_목차000919)

[static String com.kollus.sdk.media.util.Utils.makeSHA1 (String param)[static] 116](#_목차000920)

[매개변수 116](#_목차000921)

[반환값 116](#_목차000922)

[static void com.kollus.sdk.media.util.Utils.saveDirectoryJSON (Context context, String dbPath, String jsonString)[static] 116](#_목차000923)

[매개변수 116](#_목차000924)

[static void com.kollus.sdk.media.util.Utils.setDecoderType (Context context, DecoderType type)[static] 116](#_목차000925)

[매개변수 116](#_목차000926)

[참고 116](#_목차000927)

[static void com.kollus.sdk.media.util.Utils.setPlayerId (String id)[static] 116](#_목차000928)

[매개변수 116](#_목차000929)

[static void com.kollus.sdk.media.util.Utils.setRenderType (Context context, @MediaPlayerBase.RENDER\_MODE int mode)[static] 116](#_목차000930)

[매개변수 116](#_목차000931)

[참고 116](#_목차000932)

[static void com.kollus.sdk.media.util.Utils.setStoragePath (Context context, String path)[static] 116](#_목차000933)

[매개변수 117](#_목차000934)

[static String com.kollus.sdk.media.util.Utils.stringForTime (int timeMs)[static] 117](#_목차000935)

[매개변수 117](#_목차000936)

[반환값 117](#_목차000937)

[static String com.kollus.sdk.media.util.Utils.stringForTime (String dayString, String hourString, String minString, String secString, int timeMs)[static] 117](#_목차000938)

[매개변수 117](#_목차000939)

[반환값 117](#_목차000940)

[static String com.kollus.sdk.media.util.Utils.stringForTimeHHMMSS (int timeMs)[static] 117](#_목차000941)

[매개변수 117](#_목차000942)

[반환값 117](#_목차000943)

[static String com.kollus.sdk.media.util.Utils.stringForTimeHMMSS (int timeMs)[static] 117](#_목차000944)

[매개변수 117](#_목차000945)

[반환값 117](#_목차000946)

[static String com.kollus.sdk.media.util.Utils.stringForTimeMMSS (int timeMs)[static] 117](#_목차000947)

[매개변수 117](#_목차000948)

[반환값 117](#_목차000949)

[멤버 데이터 문서화 117](#_목차000950)

[final int com.kollus.sdk.media.util.Utils.PLAYER\_TYPE\_KOLLUS = 2[static] 118](#_목차000951)

[final int com.kollus.sdk.media.util.Utils.PLAYER\_TYPE\_NATIVE = 1[static] 118](#_목차000952)

[final int com.kollus.sdk.media.util.Utils.PLAYER\_TYPE\_NONE = 0[static] 118](#_목차000953)

[색인 119](#_목차000954)

# 

# 문서화클래스 문서화

## com.kollus.sdk.media.content.BandwidthItem 클래스 참조

com.kollus.sdk.media.content.BandwidthItem**Public 멤버 함수**

**BandwidthItem** (int bandwidth, String bandwidthName)

int **getBandwidth** ()

String **getBandwidthName** ()

### 상세한 설명

ABR 컨텐츠의 대역폭 정보를 관리하는 인터페이스

### 생성자 & 소멸자 문서화

**com.kollus.sdk.media.content.BandwidthItem.BandwidthItem (int *bandwidth*, String *bandwidthName*)**

대역폭 정보를 관리하는 인터페이스의 생성자. SDK내부에서 관리합니다.

##### 매개변수

| *bandwidth* | 대역폭 |
| --- | --- |
| *bandwidthName* | 대역폭 이름 |

### 멤버 함수 문서화

**int com.kollus.sdk.media.content.BandwidthItem.getBandwidth ()**

onChangedBandwidth 콜백된 경우 대역폭을 알고자 하는 경우 호출하시면 됩니다.

##### 반환값

대역폭

#### String com.kollus.sdk.media.content.BandwidthItem.getBandwidthName ()

onChangedBandwidth 콜백된 경우 해당 대역폭의 이름을 알고자 하는 경우 호출하시면 됩니다. 비디오 화면 사이즈가 정보가 있는 경우 세로에 P가 붙고 알 수 없으면 대역폭을 kbps단위로 명명합니다.

##### 반환값

대역폭 이름

## com.kollus.sdk.media.content.KollusContent.ChattingInfo 클래스 참조

com.kollus.sdk.media.content.KollusContent.ChattingInfo**Public 속성**

String **mainUrl**

String **kind**

String **position**

boolean **isVisible**

boolean **isAdmin**

boolean **isAnonymous**

boolean **bDisableDefaultProfilePhoto**

String **roomId**

String **chatServer**

String **helloMessage**

String **userName**

String **userId**

String **photoUrl**

### 상세한 설명

채팅 정보 인터페이스

### 멤버 데이터 문서화

**boolean com.kollus.sdk.media.content.KollusContent.ChattingInfo.bDisableDefaultProfilePhoto**

사용자 대표이미지 보이기 유무

#### String com.kollus.sdk.media.content.KollusContent.ChattingInfo.chatServer

채팅 서버

#### String com.kollus.sdk.media.content.KollusContent.ChattingInfo.helloMessage

채팅 입장 메세지

#### boolean com.kollus.sdk.media.content.KollusContent.ChattingInfo.isAdmin

관리자인지 여부

#### boolean com.kollus.sdk.media.content.KollusContent.ChattingInfo.isAnonymous

익명인지 여부

#### boolean com.kollus.sdk.media.content.KollusContent.ChattingInfo.isVisible

채팅 보이는지 여부

#### String com.kollus.sdk.media.content.KollusContent.ChattingInfo.kind

채팅 종류

#### String com.kollus.sdk.media.content.KollusContent.ChattingInfo.mainUrl

채팅 메인 주소

#### String com.kollus.sdk.media.content.KollusContent.ChattingInfo.photoUrl

사용자 대표 이미지

#### String com.kollus.sdk.media.content.KollusContent.ChattingInfo.position

채팅 위치

#### String com.kollus.sdk.media.content.KollusContent.ChattingInfo.roomId

채팅방 번호

#### String com.kollus.sdk.media.content.KollusContent.ChattingInfo.userId

사용자 아이디

#### String com.kollus.sdk.media.content.KollusContent.ChattingInfo.userName

사용자 이름

## com.kollus.sdk.media.content.KollusContent.ContentType Enum Reference

com.kollus.sdk.media.content.KollusContent.ContentType**Public 속성**

**KOLLUS\_CONTENTEXT\_DRM\_CONTENT**

### 상세한 설명

컨텐츠 타입관련 인터페이스

### 멤버 데이터 문서화

**com.kollus.sdk.media.content.KollusContent.ContentType.EXT\_DRM\_CONTENT**

Multi DRM 컨텐츠

#### com.kollus.sdk.media.content.KollusContent.ContentType.KOLLUS\_CONTENT

Kollus DRM 컨텐츠

## com.kollus.sdk.media.util.CpuInfo 클래스 참조

com.kollus.sdk.media.util.CpuInfo**Public 멤버 함수**

String **getCpuName** ()

boolean **isSupportDevice** ()

Vector< String > **getFrequence** ()

int **getFrequenceWithInt** ()

int **getCpuCount** ()

boolean **hasFeature** (String feature)

### 정적 Public 멤버 함수

static synchronized **CpuInfo** **getInstance** ()

### 멤버 함수 문서화

**int com.kollus.sdk.media.util.CpuInfo.getCpuCount ()**

CPU의 코어 수를 가져오는 함수

##### 반환값

CPU 코어 수

#### String com.kollus.sdk.media.util.CpuInfo.getCpuName ()

CPU의 이름을 가져오는 함수

##### 반환값

프로세스 이름

#### Vector< String > com.kollus.sdk.media.util.CpuInfo.getFrequence ()

CPU의 freequency를 String형으로 가져오는 함수

##### 반환값

CPU 클럭 리스트

#### int com.kollus.sdk.media.util.CpuInfo.getFrequenceWithInt ()

CPU의 freequency를 int형으로 가져오는 함수

##### 반환값

CPU 최대 클럭

#### static synchronized CpuInfo com.kollus.sdk.media.util.CpuInfo.getInstance ()[static]

Singleton 생성자

##### 반환값

CpuInfo 인스턴스

#### boolean com.kollus.sdk.media.util.CpuInfo.hasFeature (String *feature*)

CPU에 주어진 feature가 있는 지 여부를 알 수 있는 함수

##### 매개변수

| *feature* | 확인을 원하는 feature |
| --- | --- |

##### 반환값

feature가 있는 지 여부

#### boolean com.kollus.sdk.media.util.CpuInfo.isSupportDevice ()

지원하는 단말인 지 확인하는 함수

##### 반환값

지원 여부

## com.kollus.sdk.media.util.Utils.DecoderType Enum Reference

com.kollus.sdk.media.util.Utils.DecoderType**Public 속성**

**HW\_CODECSW\_CODEC**

### 상세한 설명

코덱 타입

### 멤버 데이터 문서화

**com.kollus.sdk.media.util.Utils.DecoderType.HW\_CODEC**

하드웨어 코덱

#### com.kollus.sdk.media.util.Utils.DecoderType.SW\_CODEC

소프트웨어 코덱

## com.kollus.sdk.media.util.Utils.DEVICE\_TYPE Enum Reference

com.kollus.sdk.media.util.Utils.DEVICE\_TYPE**Public 속성**

**DEVICE\_MOBILEDEVICE\_TABLET**

**DEVICE\_TV**

### 상세한 설명

디바이스 타입

### 멤버 데이터 문서화

**com.kollus.sdk.media.util.Utils.DEVICE\_TYPE.DEVICE\_MOBILE**

모바일

#### com.kollus.sdk.media.util.Utils.DEVICE\_TYPE.DEVICE\_TABLET

테블릿

#### com.kollus.sdk.media.util.Utils.DEVICE\_TYPE.DEVICE\_TV

TV

## com.kollus.sdk.media.KollusStorage.DrmSessionManagerListener 인터페이스 참조

com.kollus.sdk.media.KollusStorage.DrmSessionManagerListener**Public 멤버 함수**

void **onError** (int errorCode)

## com.kollus.sdk.media.EmulatorCheckerListener 인터페이스 참조

com.kollus.sdk.media.EmulatorCheckerListener**Public 멤버 함수**

void **onDetectRooting** ()

void **onRunningEmulator** ()

void **onDetectDebuggingMode** ()

### 멤버 함수 문서화

**void com.kollus.sdk.media.EmulatorCheckerListener.onDetectDebuggingMode ()**

재생 중에 디버깅 모드가 켜져 있는 경우 호출됩니다.

#### void com.kollus.sdk.media.EmulatorCheckerListener.onDetectRooting ()

재생 중에 루팅이 감지되었을 때 호출됩니다.

#### void com.kollus.sdk.media.EmulatorCheckerListener.onRunningEmulator ()

에뮬레이터에서 재생했을 때 호출됩니다.

## com.kollus.sdk.media.util.ErrorCodes 클래스 참조

com.kollus.sdk.media.util.ErrorCodes**Public 멤버 함수**

String **getErrorString** (int errorCode)

### 정적 Public 멤버 함수

static synchronized **ErrorCodes** **getInstance** (Context context)

### 정적 Public 속성

static final int **ERROR\_EXPIRED\_KEY** = -2

static final int **ERROR\_INVALID\_KEY** = -1

static final int **ERROR\_OK** = 0

static final int **ERROR\_UNDEFINED\_CODE** = -1000

static final int **MEDIA\_ERROR\_IO** = -1004

static final int **MEDIA\_ERROR\_MALFORMED** = -1007

static final int **ERROR\_UNSUPPORTED** = -1010

static final int **ERROR\_UNSUPPORTED\_DEVICE** = -1015

static final int **ERROR\_CODEC\_INIT** = -1102

static final int **ERROR\_CODEC\_DECODE** = -1103

static final int **ERROR\_CHECK\_SETTING\_NOT\_SW\_CODEC** = -1104

static final int **ERROR\_SERVER\_BLACK\_OUT** = -1105

static final int **ERROR\_ALREADY\_DOWNLOADED** = -1631

static final int **ERROR\_TMCM\_INIT** = -2001

static final int **ERROR\_TMCM\_ROOTING** = -2002

static final int **ERROR\_TMCM\_EMURATOR** = -2003

static final int **ERROR\_EMURATOR** = -2004

static final int **ERROR\_EMURATOR\_VIA\_DEVICE** = -2005

static final int **ERROR\_UNKNOWN\_MULTI\_DRM** = -2100

static final int **ERROR\_LIVE\_NOT\_BROADCASTING** = -3012

static final int **ERROR\_DUPLICATION\_BLOCK\_OR\_EXPIRED\_TIME\_MEDIA\_KEY** = -3013

static final int **ERROR\_INVALID\_MEDIA\_CONTENT\_KEY** = -4001

static final int **ERROR\_INVALID\_SECURITY\_KEY** = -4002

static final int **ERROR\_INVALID\_CHANNEL\_KEY** = -4003

static final int **ERROR\_INVALID\_USER\_KEY** = -4004

static final int **ERROR\_INVALID\_TRANSCODING\_FILE\_PATH** = -4005

static final int **ERROR\_INVALID\_ALIAS\_KEY** = -4006

static final int **ERROR\_INVALID\_JWT** = -4007

static final int **ERROR\_INVALID\_MEDIA\_TOKEN** = -4008

static final int **ERROR\_NO\_TRANSCODING\_FILE** = -4011

static final int **ERROR\_NO\_MEDIA\_CONTENT** = -4012

static final int **ERROR\_NO\_CONTENT\_OWNER** = -4013

static final int **ERROR\_NO\_CONTENT\_DISTRIBUTOR** = -4014

static final int **ERROR\_NO\_CHANNEL** = -4015

static final int **ERROR\_NO\_PAYMENT** = -4016

static final int **ERROR\_NO\_DEFAULT\_MAIN\_SITE** = -4019

static final int **ERROR\_NO\_MAIN\_MEDIA\_CONTENT** = -4020

static final int **ERROR\_UNAVAILABLE\_TRANSCODING\_FILE** = -4021

static final int **ERROR\_UNAVAILABLE\_MEDIA\_CONTENT** = -4022

static final int **ERROR\_UNAVAILABLE\_CONTENT\_OWNER** = -4023

static final int **ERROR\_UNAVAILABLE\_CONTENT\_DISTRIBUTOR** = -4024

static final int **ERROR\_UNAVAILABLE\_CHANNEL** = -4025

static final int **ERROR\_UNAVAILABLE\_CHANNEL\_DISTRIBUTION** = -4026

static final int **ERROR\_NOT\_ALLOWED\_REFERER** = -4031

static final int **ERROR\_NOT\_ALLOWED\_ACCESS\_FOR\_BLOCKING\_CAPTURE** = -4032

static final int **ERROR\_NOT\_PUBLIC\_SHARED\_CHANNEL** = -4081

static final int **ERROR\_MISMATCH\_USER\_KEY** = -4082

static final int **ERROR\_ACCESS\_WITHOUT\_MEDIA\_CONTENT\_KEY** = -4083

static final int **ERROR\_NOT\_SUPPORT\_DEVICE** = -4084

static final int **ERROR\_TOKEN\_EXPIRED** = -4085

static final int **ERROR\_NETWORK\_CON\_TIMEOUT** = -4105

static final int **ERROR\_MULTI\_DRM\_EXPIRED** = -7000

static final int **ERROR\_INVALID\_CRYPTO\_KEY** = -7001

static final int **ERROR\_WIDEVINE\_NOT\_SUPPORT** = -7109

static final int **ERROR\_STOAGE\_MANAGER\_BASE** = -8000

static final int **ERROR\_CURLE\_COULDNT\_RESOLVE\_HOST** = -8006

static final int **ERROR\_CURLE\_COULDNT\_CONNECT** = -8007

static final int **ERROR\_CURLE\_PARTIAL\_FILE** = -8018

static final int **ERROR\_CURLE\_OPERATION\_TIMEDOUT** = -8028

static final int **ERROR\_CURLE\_RECV\_ERROR** = -8056

static final int **ERROR\_BAD\_REQUEST** = -8400

static final int **ERROR\_UNAUTHORIZED\_1** = -8401

static final int **ERROR\_UNAUTHORIZED\_2** = -8402

static final int **ERROR\_FORBIDDEN** = -8403

static final int **ERROR\_NOT\_EXIST\_FILE** = -8404

static final int **ERROR\_METHOD\_NOT\_ALLOWED** = -8405

static final int **ERROR\_INTERNAL\_SERVER** = -8500

static final int **ERROR\_NOT\_IMPLEMENTED** = -8501

static final int **ERROR\_BAD\_GATEWAY** = -8502

static final int **ERROR\_SERVICE\_UNAVAILABLE** = -8503

static final int **ERROR\_GATEWAY\_TIMEOUT** = -8504

static final int **ERROR\_HTTP\_VERSION\_NOT\_SUPPORTED** = -8505

static final int **ERROR\_REQUEST\_URL** = -8461

static final int **ERROR\_GET\_USER\_IP** = -8462

static final int **ERROR\_ACCESS\_DENIED\_FOLDER** = -8463

static final int **ERROR\_DUPLICATION\_BLOCK\_PROCESS** = -8464

static final int **ERROR\_VERIFY\_MEDIA\_KEY** = -8465

static final int **ERROR\_REQEST\_URL\_BLOCK** = -8466

static final int **ERROR\_MEDIA\_KEY\_TIME\_EXPIRE** = -8467

static final int **ERROR\_MEDIA\_DOMAIN\_VALIDATE** = -8468

static final int **ERROR\_HEAVY\_REQUEST** = -8469

static final int **ERROR\_PLYMENT** = -8470

static final int **ERROR\_GEO\_RESTRICTION** = -8471

static final int **ERROR\_PREVIEW\_ERROR** = -8472

static final int **ERROR\_USER\_KEY** = -8473

static final int **ERROR\_CONTENT\_NOT\_PUBLIC\_CHANNEL** = -8474

static final int **ERROR\_USED\_MEDIA\_KEY** = -8475

static final int **ERROR\_MAX\_REQUEST\_SIZE** = -8476

static final int **ERROR\_DIFFERENT\_ES\_KEY** = -8477

static final int **ERROR\_ACCESS\_DENIED\_MEMCACHE\_SERVER** = -8478

static final int **ERROR\_DUPLICATION\_BLOCK\_MEDIA\_KEY** = -8479

static final int **ERROR\_NOT\_NORMAL\_REQUEST\_VIDEO\_GATEWAY** = -8480

static final int **ERROR\_EDGE\_RADIS\_WRITING** = -8481

static final int **ERROR\_EDGE\_RADIS\_READING** = -8482

static final int **ERROR\_CREATE\_DIRECTORY** = -8600

static final int **ERROR\_CREATE\_FILE** = -8601

static final int **ERROR\_SAVE\_DATA** = -8602

static final int **ERROR\_OPEN\_DB** = -8603

static final int **ERROR\_CREATE\_TABLE** = -8604

static final int **ERROR\_SELECT\_TABLE** = -8605

static final int **ERROR\_DELETE\_RECORD** = -8606

static final int **ERROR\_INSERT\_RECORD** = -8607

static final int **ERROR\_OPEN\_FILE** = -8608

static final int **ERROR\_OUT\_OF\_MEMORY** = -8609

static final int **ERROR\_READ\_FILE** = -8610

static final int **ERROR\_WRITE\_FILE** = -8611

static final int **ERROR\_PARAM\_VALUE** = -8613

static final int **ERROR\_NOT\_LOAD** = -8614

static final int **ERROR\_CANCEL\_DOWNLOAD** = -8615

static final int **ERROR\_GET\_CONTENTS\_LENGTH** = -8617

static final int **ERROR\_GET\_ENCODE\_LEVEL** = -8618

static final int **ERROR\_GET\_MEDIA\_URL** = -8619

static final int **ERROR\_NOT\_FOUND\_DATA** = -8620

static final int **ERROR\_NOT\_FOUND\_ID** = -8621

static final int **ERROR\_DECODE\_HEADER** = -8622

static final int **ERROR\_STORAGE\_FULL** = -8623

static final int **ERROR\_UPDATE\_RECORD** = -8624

static final int **ERROR\_NOT\_FOUND\_SNAPSHOT\_FILENAME** = -8626

static final int **ERROR\_NOT\_FOUND\_THUMBNAIL\_FILENAME** = -8627

static final int **ERROR\_NOT\_FOUND\_RECORD** = -8628

static final int **ERROR\_NOT\_DOWNLOAD\_COMPLETE\_ALL** = -8629

static final int **ERROR\_NOT\_DOWNLOAD\_TYPE** = -8630

static final int **ERROR\_ALREADY\_DOWNLOADING** = -8631

static final int **ERROR\_DECRYPT\_CONTENT\_INFO\_LINK** = -8632

static final int **ERROR\_PARSE\_CONTENT\_INFO\_LINK** = -8633

static final int **ERROR\_DECRYPT\_COMPANY\_AUTH\_INFO** = -8634

static final int **ERROR\_PARSE\_COMPANY\_AUTH\_INFO** = -8635

static final int **ERROR\_NOT\_VAILD\_COMPANY\_AUTH\_INFO** = -8636

static final int **ERROR\_NO\_USE\_CONTENT\_URL** = -8637

static final int **ERROR\_NOT\_EXIST\_DOWNLOADED\_CONTENTS** = -8638

static final int **ERROR\_EXPIRATION\_DATE** = -8643

static final int **ERROR\_EXPIRATION\_COUNT** = -8644

static final int **ERROR\_REQUEST\_CONTENT\_URL** = -8645

static final int **ERROR\_NOT\_VAILD\_CONTENTS\_INFO** = -8646

static final int **ERROR\_CREATE\_NO\_MEDIA\_FILE** = -8647

static final int **ERROR\_DECRYPT** = -8649

static final int **ERROR\_NOT\_ALLOW\_DOWNLOAD** = -8650

static final int **ERROR\_EMPTY\_COMPANY\_AUTH\_INFO** = -8651

static final int **ERROR\_EMPTY\_CONTENT\_INFO** = -8652

static final int **ERROR\_NOT\_FOUND\_DOMAIN** = -8653

static final int **ERROR\_NOT\_RECEIVE\_DOMAIN** = -8654

static final int **ERROR\_ABNORMAL\_DRM\_INFO** = -8655

static final int **ERROR\_ABNORMAL\_DRM\_COMPLETE** = -8656

static final int **ERROR\_ABNORMAL\_DRM\_PLAY** = -8657

static final int **ERROR\_FORCE\_EXPIRED** = -8658

static final int **ERROR\_NOT\_RECEIVE\_ENC\_DATA** = -8659

static final int **ERROR\_NOT\_FOUND\_MOBILE\_LOGO\_FILENAME** = -8660

static final int **ERROR\_CANNOT\_CALL\_API** = -8661

static final int **ERROR\_NOT\_FOUND\_THUMBNAIL\_URL** = -8662

static final int **ERROR\_ALTER\_TABLE** = -8663

static final int **ERROR\_DIFFERENT\_TIME** = -8664

static final int **ERROR\_EMPTY\_POSTER\_UPLOAD\_FILE\_INFO** = -8665

static final int **ERROR\_PARSE\_POSTER\_UPLOAD\_FILE\_INFO** = -8666

static final int **ERROR\_NOT\_FOUND\_POSTER\_URL** = -8667

static final int **ERROR\_GET\_MEDIA\_ID** = -8668

static final int **ERROR\_NOT\_FOUND\_SUBTITLE\_FILENAME** = -8669

static final int **ERROR\_EMPTY\_PLAY\_CALLBACK\_INFO** = -8670

static final int **ERROR\_DECRYPT\_PLAY\_CALLBACK\_INFO** = -8671

static final int **ERROR\_PARSE\_PLAY\_CALLBACK\_INFO** = -8672

static final int **ERROR\_ABNORMAL\_PLAY\_CALLBACK\_INFO** = -8673

static final int **ERROR\_PLAY\_CALLBACK\_EXPIRED** = -8674

static final int **ERROR\_ABNORMAL\_PLAY\_CALLBACK\_INFO\_PLAY** = -8675

static final int **ERROR\_ABNORMAL\_UPLOAD\_POSTER\_INFO** = -8676

static final int **ERROR\_GET\_CONTENT\_TYPE** = -8677

static final int **ERROR\_GET\_CONTENT\_LAST\_MODIFIED** = -8678

static final int **ERROR\_ENCRYPT\_KEY** = -8679

static final int **ERROR\_FORCE\_DELETE** = -8680

static final int **ERROR\_DIFFERENT\_EXPIRE\_RESET** = -8681

static final int **ERROR\_NOT\_FOUND\_DRM\_KIND** = -8682

static final int **ERROR\_EXPIRATION\_PLAY\_TIME** = -8683

static final int **ERROR\_PLAY\_CALLBACK\_EXPIRATION\_PLAY\_TIME** = -8686

static final int **ERROR\_CHECK\_EXPIRATION\_DATE** = -8687

static final int **ERROR\_NOT\_FOUND\_COMMON\_INFO** = -8800

static final int **ERROR\_NOT\_FOUND\_ITEM\_IN\_PLAYLIST** = -8801

static final int **ERROR\_EMPTY\_PLAYLIST\_INFO** = -8802

static final int **ERROR\_INTRO\_CANNOT\_DOWNLOAD** = -8803

static final int **ERROR\_DUPLICATION\_NETWORK\_ERR** = -8804

static final int **INFO\_NETWORK\_TIMEOUT** = -8899

static final int **ERROR\_VIDEO\_GATEWAY\_ABNORMAL\_RESPONSE** = -8900

static final int **ERROR\_DRM\_CALLBACK\_BASE** = -9000

static final int **ERROR\_PLAY\_CALLBACK\_BASE** = -10000

static final int **ERROR\_VWM\_BASE** = -11000

static final int **ERROR\_FORENSIC\_CURLE\_COULDNT\_RESOLVE\_HOST** = -11006

static final int **ERROR\_FORENSIC\_CURLE\_COULDNT\_CONNECT** = -11007

static final int **ERROR\_FORENSIC\_CURLE\_PARTIAL\_FILE** = -11018

static final int **ERROR\_FORENSIC\_CURLE\_OPERATION\_TIMEDOUT** = -11028

static final int **ERROR\_FORENSIC\_CURLE\_RECV\_ERROR** = -11056

static final int **ERROR\_FORENSIC\_INSTANCE\_INITIALIZE** = -11099

static final int **ERROR\_FORENSIC\_MESSAGE\_ID\_RANGE** = -11100

static final int **ERROR\_FORENSIC\_CURLE\_CONNECT** = -11101

static final int **ERROR\_FORENSIC\_CURLE\_HEADER** = -11102

static final int **ERROR\_FORENSIC\_DESCRYPT\_INSTANCE** = -11103

static final int **ERROR\_FORENSIC\_DESCRYPT\_FUNCTION** = -11104

static final int **ERROR\_FORENSIC\_DESCRYPT\_KEY** = -11105

static final int **ERROR\_FORENSIC\_DESCRYPT\_CURL\_DATA** = -11106

static final int **ERROR\_FORENSIC\_RECRIVED\_DATA\_BROKEN** = -11107

static final int **ERROR\_FORENSIC\_ENCRYPT\_CACHE\_DATA** = -11108

static final int **ERROR\_FORENSIC\_DESCRYPT\_CACHE\_DATA\_SIZE** = -11109

static final int **ERROR\_FORENSIC\_EXPIRED\_LICENSE** = -11110

static final int **ERROR\_FORENSIC\_DESCRYPT\_WATERMARK\_PATTERN\_DATA** = -11111

static final int **ERROR\_FORENSIC\_BAD\_REQUEST** = -11400

static final int **ERROR\_FORENSIC\_UNAUTHORIZED\_1** = -11401

static final int **ERROR\_FORENSIC\_UNAUTHORIZED\_2** = -11402

static final int **ERROR\_FORENSIC\_NOT\_EXIST\_FILE** = -11404

static final int **ERROR\_FORENSIC\_INTERNAL\_SERVER** = -11500

static final int **ERROR\_FORENSIC\_NOT\_IMPLEMENTED** = -11501

static final int **ERROR\_FORENSIC\_BAD\_GATEWAY** = -11502

static final int **ERROR\_FORENSIC\_SERVICE\_UNAVAILABLE** = -11503

static final int **ERROR\_FORENSIC\_GATEWAY\_TIMEOUT** = -11504

static final int **ERROR\_FORENSIC\_HTTP\_VERSION\_NOT\_SUPPORTED** = -11505

### Protected 속성

final Context **mContext**

final Map< Integer, CodeBook > **mCodeBooks**

### 멤버 함수 문서화

**String com.kollus.sdk.media.util.ErrorCodes.getErrorString (int *errorCode*)**

Integer 코드에 해당하는 에러문자를 가져오는 함수

##### 매개변수

| *errorCode* | 에러코드 |
| --- | --- |

##### 반환값

에러코드에 해당하는 메세지

#### static synchronized ErrorCodes com.kollus.sdk.media.util.ErrorCodes.getInstance (Context *context*)[static]

ErrorCodes 인스턴스를 가져오는 함수

##### 매개변수

| *context* | App Context |
| --- | --- |

##### 반환값

ErrorCodes 인스턴스

### 멤버 데이터 문서화

**final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ABNORMAL\_DRM\_COMPLETE = -8656[static]**

DRM 컨텐츠 다운로드 완료하였으나, 결과값으로 비정상 값을 받은 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ABNORMAL\_DRM\_INFO = -8655[static]

DRM 컨텐츠 정보 요청시, 결과값으로 비정상 값을 받은 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ABNORMAL\_DRM\_PLAY = -8657[static]

DRM 컨텐츠 재생하려는 경우, 결과값으로 비정상 값을 받은 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ABNORMAL\_PLAY\_CALLBACK\_INFO = -8673[static]

Play 콜백 호출(Kind 1)하고 응답 받은 내용의 result 값이 0(에러)인 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ABNORMAL\_PLAY\_CALLBACK\_INFO\_PLAY = -8675[static]

Play 콜백 호출(Kind 3)하고 응답 받은 내용의 result 값이 0(에러)인 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ABNORMAL\_UPLOAD\_POSTER\_INFO = -8676[static]

poster 업로드 후 서버로 응답 받은 내용중 error 값이 ture인 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ACCESS\_DENIED\_FOLDER = -8463[static]

인증이 필요한 폴더에 접근시도

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ACCESS\_DENIED\_MEMCACHE\_SERVER = -8478[static]

로컬 멤캐쉬서버 접속 오류로 키 세팅 불가

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ALREADY\_DOWNLOADED = -1631[static]

이미 다운로드한 컨텐츠

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ALREADY\_DOWNLOADING = -8631[static]

다운로드 하고 있는데 다운로드 api를 호출시

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ALTER\_TABLE = -8663[static]

DB 테이블 변경에러

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_BAD\_GATEWAY = -8502[static]

게이트웨이 상태 나쁨 - 이 에러는 게이트웨이 상태가 나쁘거나 서버의 과부하 상태일 때 발생

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_BAD\_REQUEST = -8400[static]

문법상 오류가 있어서 서버가 요청 사항을 이해하지 못함

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CANCEL\_DOWNLOAD = -8615[static]

컨텐츠 다운로드 취소

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CANNOT\_CALL\_API = -8661[static]

API를 호출할 수 없습니다.(내부 객체를 종료하는 시점)

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CHECK\_EXPIRATION\_DATE = -8687[static]

DRM 다운로드 컨텐츠 재생시 체크시간이 지났을 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CHECK\_SETTING\_NOT\_SW\_CODEC = -1104[static]

소프트웨어 코덱이 아님

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CODEC\_DECODE = -1103[static]

디코딩 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CODEC\_INIT = -1102[static]

코덱 초기화 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CONTENT\_NOT\_PUBLIC\_CHANNEL = -8474[static]

해당 컨텐츠의 채널이 공개모드가 아님

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CREATE\_DIRECTORY = -8600[static]

디렉토리 생성 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CREATE\_FILE = -8601[static]

파일 생성 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CREATE\_NO\_MEDIA\_FILE = -8647[static]

.nomedia 파일 생성 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CREATE\_TABLE = -8604[static]

DB 테이블 생성 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CURLE\_COULDNT\_CONNECT = -8007[static]

네트워크 연결 오류

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CURLE\_COULDNT\_RESOLVE\_HOST = -8006[static]

호스트명 오류

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CURLE\_OPERATION\_TIMEDOUT = -8028[static]

네트워크 타임 아웃

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CURLE\_PARTIAL\_FILE = -8018[static]

서버에 요청한 사이즈와 응답으로 받은 데이터의 사이즈가 상이한 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_CURLE\_RECV\_ERROR = -8056[static]

Receive 오류

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DECODE\_HEADER = -8622[static]

암호화 데이타 복호화 실패시 media url link로 인증서버에 요청후 응답 받은 헤더의 XHTTP\_ENC\_DATA 데이타에 대해 복호화 실패시

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DECRYPT = -8649[static]

암호화 문자열 복호화 실패시

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DECRYPT\_COMPANY\_AUTH\_INFO = -8634[static]

업체 인증 정보 복호화 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DECRYPT\_CONTENT\_INFO\_LINK = -8632[static]

컨텐츠 정보 링크 복호화 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DECRYPT\_PLAY\_CALLBACK\_INFO = -8671[static]

Play 콜백 호출후 응답받은 데이타를 복호화 실패시

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DELETE\_RECORD = -8606[static]

DB 레코드 삭제 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DIFFERENT\_ES\_KEY = -8477[static]

요청된 키가 시스템에 저장된 값과 다름

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DIFFERENT\_EXPIRE\_RESET = -8681[static]

DRM 콜백 세션이 일치하지 않아 content\_expire\_reset 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DIFFERENT\_TIME = -8664[static]

기기와 서버간의 시간이 일치하지 않아 다운로드를 허용하지 않습니다.

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DUPLICATION\_BLOCK\_MEDIA\_KEY = -8479[static]

중복 차단으로 거부된 media\_key

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_DUPLICATION\_BLOCK\_PROCESS = -8464[static]

중복 차단 프로세스 진행중 오류 발생

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EMPTY\_COMPANY\_AUTH\_INFO = -8651[static]

고객 인증 서버로 부터 AUTH 정보를 얻지 못한 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EMPTY\_CONTENT\_INFO = -8652[static]

컨텐츠 정보를 얻지 못한 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EMPTY\_PLAY\_CALLBACK\_INFO = -8670[static]

Play 콜백 호출후 응답은 받았지만 수신 데이타가 없는 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EMPTY\_PLAYLIST\_INFO = -8802[static]

플래이 리스트 요청후 응답은 받았지만 수신 데이타가 없는 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EMPTY\_POSTER\_UPLOAD\_FILE\_INFO = -8665[static]

포스터 파일 업로드 후, 서버로 부터 응답은 받았지만 수신 데이타가 없는 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_ENCRYPT\_KEY = -8679[static]

암호화 키 생성실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EXPIRATION\_COUNT = -8644[static]

재생 유효횟수를 모두 소진했을 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EXPIRATION\_DATE = -8643[static]

단말기 타입값이 유효하지 않습니다. 단말기 레밸값이 유효하지 않습니다. 추가 블록 다운로드 완료 에러는 아니고 이 코드가 API로 반환도 되지 않는다 네트웍 라이브러리 초기화 실패 재생 유효기간이 지났을 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EXPIRATION\_PLAY\_TIME = -8683[static]

DRM 다운로드 컨텐츠 재생 유효 재생시간을 모두 소진했을 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_EXPIRED\_KEY = -2[static]

인증키 만료

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_FORBIDDEN = -8403[static]

권한 없음 - 접속 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_FORCE\_DELETE = -8680[static]

DRM 콜백 서버로 부터 삭제 요청을 받은 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_FORCE\_EXPIRED = -8658[static]

DRM 콜백 URL로 kind:3로 전달후, 강제 expire 정보를 내려받은 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GATEWAY\_TIMEOUT = -8504[static]

네트워크 타임 아웃

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GEO\_RESTRICTION = -8471[static]

해당 국가 차단

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_CONTENT\_LAST\_MODIFIED = -8678[static]

엣지 서버로 부터 LAST-MODIFIED 정보 얻기 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_CONTENT\_TYPE = -8677[static]

엣지 서버로 부터 Content-Type 정보 얻기 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_CONTENTS\_LENGTH = -8617[static]

네트웍 문제 발생 서버로 부터 컨텐츠 길이 얻기 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_ENCODE\_LEVEL = -8618[static]

서버로 부터 컨텐츠 인코딩 레밸 정보 얻기 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_MEDIA\_ID = -8668[static]

엣지 서버로 부터 미디어 ID를 얻기 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_MEDIA\_URL = -8619[static]

서버로 부터 미디어 URL 얻기 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_GET\_USER\_IP = -8462[static]

사용자의 유효한 IP 정보 확인 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_HEAVY\_REQUEST = -8469[static]

Heavy request error

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_HTTP\_VERSION\_NOT\_SUPPORTED = -8505[static]

브라우저의 설정에 Http1.1 설정이 되어 있는지 확인

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_INSERT\_RECORD = -8607[static]

DB 레코드 추가 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_INTERNAL\_SERVER = -8500[static]

서버 내부 오류 - 이 에러는 웹 서버가 요청사항을 수행할 수 없을 경우에 발생함

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_INTRO\_CANNOT\_DOWNLOAD = -8803[static]

인트로 컨텐츠는 다운로드 할 수 없음

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_INVALID\_KEY = -1[static]

인증키 오류

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_INVALID\_MEDIA\_CONTENT\_KEY = -4001[static]

비디오 게이트 웨이 에러코드

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_LIVE\_NOT\_BROADCASTING = -3012[static]

라이브시 네트워크 오류

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_MAX\_REQUEST\_SIZE = -8476[static]

인증 확인 최대 Request 길이 오류

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_MEDIA\_DOMAIN\_VALIDATE = -8468[static]

미디어 Domain validate error

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_MEDIA\_KEY\_TIME\_EXPIRE = -8467[static]

미디어키 Time Expire

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_METHOD\_NOT\_ALLOWED = -8405[static]

해당 요청이 불가능한 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_MULTI\_DRM\_EXPIRED = -7000[static]

외부 DRM 에러

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NETWORK\_CON\_TIMEOUT = -4105[static]

네트워크 타임아웃

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NO\_USE\_CONTENT\_URL = -8637[static]

처리할 수 없는 URL 요청입니다.

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_ALLOW\_DOWNLOAD = -8650[static]

해당 URL 링크는 다운로드를 허용하지 않습니다.

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_DOWNLOAD\_COMPLETE\_ALL = -8629[static]

다운로드 타입의 컨텐츠를 전부 받지 않은 경우 다운로드 타입의 컨텐츠를 다 받지 않은 상태에서 재생한 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_DOWNLOAD\_TYPE = -8630[static]

다운로드 타입의 컨텐츠가 아님 다운로드 타입이 아닌 컨텐츠 ID로 다운로드 관련 API를 호출해서 사용하려는 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_EXIST\_DOWNLOADED\_CONTENTS = -8638[static]

다운로드된 컨텐츠가 존재하지 않습니다.

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_EXIST\_FILE = -8404[static]

물리적인 파일이 존재하지 않는 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_COMMON\_INFO = -8800[static]

컨텐츠 목록에서 공통 정보를 찾지 못함

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_DATA = -8620[static]

저장된 데이타를 찾지 못한 경우 원하는 블록 정보를 찾지 못한 경우 발생하나, API 기능을 구현을 위한 내부 에러 코드 API호출시 반환값으로는 받아 볼 수 없다.

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_DOMAIN = -8653[static]

엣지 서버에 요청시 사용한 URL 도메인이 엣지서버의 응답 정보의 도메인 목록과 일치하지 않는 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_DRM\_KIND = -8682[static]

DRM 콜백 요청시 전달한 항목에 대한 응답 정보를 받지 못한 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_ID = -8621[static]

해당 컨텐츠 ID를 찾지 못한 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_ITEM\_IN\_PLAYLIST = -8801[static]

컨텐츠 목록에서 해당 항목을 찾지 못한 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_MOBILE\_LOGO\_FILENAME = -8660[static]

모바일 로고 url에서 파일명을 찾지 못함

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_POSTER\_URL = -8667[static]

비디오 게이트웨이로 Media info 정보를 얻은 항목중에 poster url을 찾지 못함

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_RECORD = -8628[static]

DB 레코드를 찾지 못함

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_SNAPSHOT\_FILENAME = -8626[static]

지울 캐시가 없는 경우 동작중인 캐시 컨텐츠를 제외하고, 사용하지 않는 캐시가 더 이상 없는 경우 API 기능을 구현을 위한 내부 에러 코드 API호출시 반환값으로는 받아 볼 수 없다. 스냅샷 url에서 파일명을 찾지 못함

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_SUBTITLE\_FILENAME = -8669[static]

비디오 게이트웨이로 Media info 정보를 얻은 항목중에 자막 url 문자열에서 파일명을 찾지 못함

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_THUMBNAIL\_FILENAME = -8627[static]

썸내일 url에서 파일명을 찾지 못함

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_FOUND\_THUMBNAIL\_URL = -8662[static]

다운로드할 썸내일 url이 없는 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_IMPLEMENTED = -8501[static]

적용 안 됨 - 이 에러는 웹 서버가 요청사항을 수행하는데 필요한 기능을 지원하지 않는 경우에 발생

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_LOAD = -8614[static]

컨텐츠를 로드하지 않아 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_NORMAL\_REQUEST\_VIDEO\_GATEWAY = -8480[static]

비디오게이트웨이를 통한 정상적 요청이 아님

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_RECEIVE\_DOMAIN = -8654[static]

엣지 서버에서 도메인 목록을 받지 못한 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_RECEIVE\_ENC\_DATA = -8659[static]

엣지 서버로 XHTTP\_ENC\_DATA 정보를 받지 못한 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_VAILD\_COMPANY\_AUTH\_INFO = -8636[static]

업체 인증 정보가 유효하지 않음 업체 인증 정보를 받은 것을 체크시 유효기간이나 유효 재생수 값이 음수인 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_NOT\_VAILD\_CONTENTS\_INFO = -8646[static]

컨텐츠 관련 정보가 손상되었습니다.

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_OK = 0[static]

정상

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_OPEN\_DB = -8603[static]

DB 파일을 열기 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_OPEN\_FILE = -8608[static]

파일 열기 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_OUT\_OF\_MEMORY = -8609[static]

메모리 부족

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PARAM\_VALUE = -8613[static]

url 링크 정보 요청시, 미디어 정보를 얻기 실패시 이 경우 사용한 함수의 인자중에 ERROR\_DETAIL\_INFO 구조체 타입의 변수에서 strMessage 맴버변수의 값을 얻어 서버로 부터 받은 에러내용을 확인한다. 함수 인자값이 적절하지 않음

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PARSE\_COMPANY\_AUTH\_INFO = -8635[static]

업체 인증 정보 JSON 파싱 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PARSE\_CONTENT\_INFO\_LINK = -8633[static]

컨텐츠 정보 링크 JSON 파싱 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PARSE\_PLAY\_CALLBACK\_INFO = -8672[static]

Play 콜백 호출후 응답받은 데이타를 JSON 파싱 실패시

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PARSE\_POSTER\_UPLOAD\_FILE\_INFO = -8666[static]

포스터 파일 업로드 후, 서버로 부터 응답받은 데이타를 JSON 파싱 실패시

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PLAY\_CALLBACK\_EXPIRATION\_PLAY\_TIME = -8686[static]

Play Callback에서 전달받은 재생 유효시간이 만료된 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PLAY\_CALLBACK\_EXPIRED = -8674[static]

Play 콜백 호출(Kind 3)하고 응답 받은 내용의 expire 값이 1인(강제만료) 경우

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PLYMENT = -8470[static]

Payment error

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_PREVIEW\_ERROR = -8472[static]

preview error

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_READ\_FILE = -8610[static]

파일 읽기 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_REQEST\_URL\_BLOCK = -8466[static]

요청 URL 차단

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_REQUEST\_CONTENT\_URL = -8645[static]

컨텐츠를 이어받으려면 컨텐츠 url로 요청해야 합니다.

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_REQUEST\_URL = -8461[static]

request url 분석 오류

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_SAVE\_DATA = -8602[static]

파일 데이타 저장 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_SELECT\_TABLE = -8605[static]

DB 테이블 select 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_SERVER\_BLACK\_OUT = -1105[static]

라이브시 네트워크 오류

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_SERVICE\_UNAVAILABLE = -8503[static]

서비스 불가능 - 이 에러는 서비스가 현재 멈춘 상태 또는 현재 일시적인 과부하 또는 관리 상황일 때 발생될 수 있다.

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_STOAGE\_MANAGER\_BASE = -8000[static]

스토리지 매니저에서 나오는 기본값

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_STORAGE\_FULL = -8623[static]

디바이스에 저장할 저장공간 부족시

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_TMCM\_INIT = -2001[static]

테르텐 에뮬레이터 초기화 오류

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UNAUTHORIZED\_1 = -8401[static]

권한 없음 - 접속 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UNAUTHORIZED\_2 = -8402[static]

권한 없음 - 접속 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UNDEFINED\_CODE = -1000[static]

알 수 없는 오류

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UNKNOWN\_MULTI\_DRM = -2100[static]

Multi DRM 오류

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UNSUPPORTED = -1010[static]

미지원 컨텐츠

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UNSUPPORTED\_DEVICE = -1015[static]

미지원 단말

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_UPDATE\_RECORD = -8624[static]

DB 레코드 갱신 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_USED\_MEDIA\_KEY = -8475[static]

해당 미디어키를 이미 사용한적이 있음

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_USER\_KEY = -8473[static]

UserKey error

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_VERIFY\_MEDIA\_KEY = -8465[static]

미디어키 검증 오류

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_VIDEO\_GATEWAY\_ABNORMAL\_RESPONSE = -8900[static]

비디오 게이트웨이에 Media info를 요청했을 때 error값이 true인 경우 (정상 재생이 불가한 상황임)

#### final int com.kollus.sdk.media.util.ErrorCodes.ERROR\_WRITE\_FILE = -8611[static]

파일 쓰기 실패

#### final int com.kollus.sdk.media.util.ErrorCodes.INFO\_NETWORK\_TIMEOUT = -8899[static]

네트웍 타임아웃시 재시도를 위한 반환값으로 API 반환값은 아님(에러 메세지에 추가할 사항 아님)

#### final int com.kollus.sdk.media.util.ErrorCodes.MEDIA\_ERROR\_IO = -1004[static]

파일 또는 네트워크 IO 관련 오류

#### final int com.kollus.sdk.media.util.ErrorCodes.MEDIA\_ERROR\_MALFORMED = -1007[static]

스트림이 표준에 맞지 않는 경우

## com.kollus.sdk.media.MediaPlayer.EXTERNAL\_DISPLAY\_MODE Enum Reference

com.kollus.sdk.media.MediaPlayer.EXTERNAL\_DISPLAY\_MODE**Public 속성**

**EXTERNAL\_HDMI\_DISPLAYEXTERNAL\_WIFI\_DISPLAY**

### 상세한 설명

외부 모니터가 감지되었을 때 타입

### 멤버 데이터 문서화

**com.kollus.sdk.media.MediaPlayer.EXTERNAL\_DISPLAY\_MODE.EXTERNAL\_HDMI\_DISPLAY**

HDMI가 연결되거나 끊겼을 때

#### com.kollus.sdk.media.MediaPlayer.EXTERNAL\_DISPLAY\_MODE.EXTERNAL\_WIFI\_DISPLAY

Wifi Display가 연결되거나 끊겼을 때

## com.kollus.sdk.media.ForensicWatermarkView 클래스 참조

com.kollus.sdk.media.ForensicWatermarkView**Public 멤버 함수**

**ForensicWatermarkView** (Context context, **KollusStorage** storage, String key, String expireDate, boolean isDownload)

void **setStrength** (int alpha)

### Protected 멤버 함수

void **onLayout** (boolean changed, int l, int t, int r, int b)

int **start** (long message)

void **stop** ()

### 생성자 & 소멸자 문서화

**com.kollus.sdk.media.ForensicWatermarkView.ForensicWatermarkView (Context *context*, KollusStorage *storage*, String *key*, String *expireDate*, boolean *isDownload*)**

포렌식 워터마크를 표시하는 View로 최상위 ViewGroup에 addView해야 합니다.

##### 매개변수

| *context* | App Context |
| --- | --- |
| *storage* | KollusStorage 인스턴스 |
| *key* | 발급된 키 |
| *expireDate* | 발급된 키의 만료 기한 |
| *isDownload* | 다운로드 컨텐츠 여부 |

### 멤버 함수 문서화

**void com.kollus.sdk.media.ForensicWatermarkView.setStrength (int *alpha*)**

포렌식 강도를 지정하는 함수입니다. UI에 addView전에 설정해야 반영이 됩니다.

##### 매개변수

| *alpha* | 범위 0~255이며 기본값은 2이며 이보다 낮으면 검출이 안될 수 있고 너무 높으면 워터마크가 보여 시인성이 떨어집니다. |
| --- | --- |

## com.kollus.sdk.media.content.KollusBookmark 클래스 참조

com.kollus.sdk.media.content.KollusBookmark**Public 멤버 함수**

**KollusBookmark** ()

int **getLevel** ()

void **setLevel** (int level)

void **setLabel** (String label)

String **getLabel** ()

void **setTitle** (String title)

String **getTitle** ()

void **setTime** (int time)

int **getTime** ()

void **setThumbnail** (Bitmap bm)

Bitmap **getThumbnail** ()

int **describeContents** ()

void **writeToParcel** (Parcel dest, int flags)

### 정적 Public 멤버 함수

static ArrayList< **KollusBookmark** > **getBookmarkInfo** (String jsonString)

### 정적 Public 속성

static final int **NO\_LEVEL** = -1

static final int **USER\_LEVEL** = 0

static final int **PROVIDER\_LEVEL** = 1

static final int **MAX\_BOOKMARK** = 1000

### 생성자 & 소멸자 문서화

**com.kollus.sdk.media.content.KollusBookmark.KollusBookmark ()**

SDK내부에서 관리합니다.

### 멤버 함수 문서화

**static ArrayList< KollusBookmark > com.kollus.sdk.media.content.KollusBookmark.getBookmarkInfo (String *jsonString*)[static]**

SDK내부에서 관리합니다.

##### 매개변수

| *jsonString* | JSON형식의 문자인 북마크 데이터 |
| --- | --- |

##### 반환값

북마크 리스트

#### String com.kollus.sdk.media.content.KollusBookmark.getLabel ()

북마크 라벨을 가져올 때 사용합니다.

##### 반환값

북마크 라벨

#### int com.kollus.sdk.media.content.KollusBookmark.getLevel ()

북마크 레벨를 가져오는 함수

##### 반환값

북마크 레벨 [USER\_LEVEL, PROVIDER\_LEVEL]

#### Bitmap com.kollus.sdk.media.content.KollusBookmark.getThumbnail ()

북마크의 이미지를 가져올 때 사용합니다.

##### 반환값

북마크 이미지

#### int com.kollus.sdk.media.content.KollusBookmark.getTime ()

북마크의 시간을 가져올 때 사용합니다.

##### 반환값

북마크 시간(초)

#### String com.kollus.sdk.media.content.KollusBookmark.getTitle ()

북마크명을 가져올 때 사용합니다.

##### 반환값

북마크명

#### void com.kollus.sdk.media.content.KollusBookmark.setLabel (String *label*)

북마크 라벨을 설정합니다.

##### 매개변수

| *label* | 북마크 라벨 |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusBookmark.setLevel (int *level*)

북마크 레벨을 설정합니다.

##### 매개변수

| *level* | 북마크 레벨 |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusBookmark.setThumbnail (Bitmap *bm*)

북마크의 이미지를 설정합니다.

##### 매개변수

| *bm* | 북마크 이미지 |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusBookmark.setTime (int *time*)

북마크 시간을 설정합니다.

##### 매개변수

| *time* | 북마크의 시간(초) |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusBookmark.setTitle (String *title*)

북마크 명을 설정합니다.

##### 매개변수

| *title* | 북마크명 |
| --- | --- |

### 멤버 데이터 문서화

**final int com.kollus.sdk.media.content.KollusBookmark.MAX\_BOOKMARK = 1000[static]**

최대 북마크 개수

#### final int com.kollus.sdk.media.content.KollusBookmark.PROVIDER\_LEVEL = 1[static]

Service Provider가 추가한 북마크

#### final int com.kollus.sdk.media.content.KollusBookmark.USER\_LEVEL = 0[static]

사용자가 추가한 북마크

## com.kollus.sdk.media.content.KollusContent 클래스 참조

com.kollus.sdk.media.content.KollusContent**클래스**

class **ChattingInfo**enum **ContentType**

class **SubtitleInfo**

### Public 멤버 함수

**KollusContent** ()

**KollusContent** (Parcel in)

void **setForPlay** (Parcel in)

void **setForDownload** (Parcel in)

void **setByJson** (String jsonString)

void **copy** (**KollusContent** content)

boolean **isSection** ()

String **getMediaContentKeyMD5** ()

void **setMediaContentKey** (String key)

String **getMediaContentKey** ()

String **getUploadFileKey** ()

void **setContentType** (**ContentType** type)

**ContentType** **getContentType** ()

String **getLicenseUrl** ()

String **getLicenseKey** ()

String **getLicenseToken** ()

String **getCompany** ()

String **getThumbnailPath** ()

String **getOriginThumbnailPath** ()

String **getScreenShotPath** ()

String **getAiScriptPath** ()

String **getCourse** ()

String **getSubCourse** ()

String **getTeacher** ()

String **getSynopsis** ()

String **getDetailInfoUrl** ()

int **getUriIndex** ()

void **setUriIndex** (int index)

void **setPlaytime** (int playtime)

int **getPlaytime** ()

int **getStartAt** ()

int **getDuration** ()

void **setReceivedSize** (long receiveSize)

long **getReceivedSize** ()

void **setReceivingSize** (long receiveSize)

long **getReceivingSize** ()

void **setFileSize** (long fileSize)

long **getFileSize** ()

boolean **IsInvalidFileSize** ()

String **getMediaUrl** ()

void **setLocalMediaUrl** (String mediaUrl)

String **getLocalMediaUrl** ()

int **getPlaySectionStart** ()

int **getPlaySectionEnd** ()

int **getRepeatPlaySectionStart** ()

int **getRepeatPlaySectionEnd** ()

boolean **getDisablePlayRate** ()

int **getSeekableEnd** ()

String **getCaptionStyle** ()

boolean **isVmCheck** ()

void **setDownloadPercent** (int percent)

int **getDownloadPercent** ()

void **setDownloadCompleted** (boolean complete)

boolean **isCompleted** ()

void **setDownloadError** (boolean bError)

boolean **getDownloadError** ()

void **setLoaded** (boolean bLoaded)

boolean **isLoaded** ()

String **getNextEpisodeCallbackUrl** ()

String **getNextEpisodeCallbackData** ()

int **getNextEpisodeShowTime** ()

boolean **getForceNScreen** ()

int **getMaxPlaybackRate** ()

boolean **useBlockEmulator** ()

boolean **getMute** ()

boolean **getDisableTvOut** ()

boolean **isAudioFile** ()

boolean **isIntro** ()

boolean **getSeekable** ()

int **getSkipSec** ()

String **getAudioWaterMark** ()

String **getForensicWaterMark** ()

boolean **hasAudioWaterMark** ()

boolean **hasForensicWaterMark** ()

boolean **isThumbnailEnable** ()

boolean **isThumbnailDownloadSync** ()

String **getVideoWaterMarkCode** ()

int **getVideoWaterMarkAlpha** ()

int **getVideoWaterMarkFontSize** ()

int **getVideoWaterMarkFontColor** ()

int **getVideoWaterMarkShowTime** ()

int **getVideoWaterMarkHideTime** ()

int **getVideoWaterMarkSpace** ()

boolean **isVideoWaterMarkRandom** ()

boolean **isLive** ()

boolean **isVr** ()

int **getExpirationDate** ()

int **getTotalExpirationCount** ()

int **getExpirationCount** ()

int **getTotalExpirationPlaytime** ()

int **getExpirationPlaytime** ()

boolean **getExpirationRefreshPopup** ()

void **setContentExpired** (boolean expired)

boolean **isContentExpirated** ()

int **getVideoWidth** ()

int **getVideoHeight** ()

int **getBitrate** ()

boolean **isDrm** ()

void **setDownloading** (boolean downloading)

boolean **isDownloading** ()

void **setDownloadCanceled** ()

boolean **isDownloadCanceled** ()

String **getSkinString** ()

**ChattingInfo** **getChattingInfo** ()

Vector< **SubtitleInfo** > **getSubtitleInfo** ()

Vector< **SubtitleInfo** > **getSubtitleSubInfo** ()

String **getServiceProviderMessage** ()

void **setServiceProviderMessage** (String message)

String **toString** ()

### 정적 Public 속성

static final Parcelable.Creator< **KollusContent** > **CREATOR**

### Protected 속성

boolean **mInvalidFileSize**

### 상세한 설명

컨텐츠 정보를 관리하는 인터페이스

### 생성자 & 소멸자 문서화

**com.kollus.sdk.media.content.KollusContent.KollusContent ()**

기본 생성자

#### com.kollus.sdk.media.content.KollusContent.KollusContent (Parcel *in*)

SDK 내부에서 사용하는 생성자

##### 매개변수

| *in* | 컨텐츠 정보 |
| --- | --- |

### 멤버 함수 문서화

**void com.kollus.sdk.media.content.KollusContent.copy (KollusContent *content*)**

주어진 인자에 컨텐츠 정보를 복사하는 함수

##### 매개변수

| *content* | 컨텐츠 정보를 복사할 대상 KollusContent |
| --- | --- |

#### String com.kollus.sdk.media.content.KollusContent.getAiScriptPath ()

AI SCript 주소를 가져오는 함수로 경로가 빈문자가 아니면 AI 배속이 가능한 컨텐츠입니다.

#### String com.kollus.sdk.media.content.KollusContent.getAudioWaterMark ()

오디오 워터 마크를 가져오는 함수

##### 반환값

오디오 워터 마크

#### int com.kollus.sdk.media.content.KollusContent.getBitrate ()

다운로드 컨텐츠 bitrate를 가져오는 함수

##### 반환값

컨텐츠 bitrate

#### String com.kollus.sdk.media.content.KollusContent.getCaptionStyle ()

자막 스타일 정보를 가져오는 함수

##### 반환값

자막 스타일

#### ChattingInfo com.kollus.sdk.media.content.KollusContent.getChattingInfo ()

채팅 설정에 필요한 정보를 가져오는 함수

##### 반환값

채팅 정보

#### String com.kollus.sdk.media.content.KollusContent.getCompany ()

사용안함

##### 반환값

회사명

#### ContentType com.kollus.sdk.media.content.KollusContent.getContentType ()

컨텐츠 타입을 가져오는 함수

##### 반환값

컨텐츠 타입

##### 참고

ContentType

#### String com.kollus.sdk.media.content.KollusContent.getCourse ()

사용 안함

##### 반환값

카테고리

#### String com.kollus.sdk.media.content.KollusContent.getDetailInfoUrl ()

다운로드 컨텐츠 상세정보 URL 정보를 가져오는 함수

##### 반환값

상세정보 URL

#### boolean com.kollus.sdk.media.content.KollusContent.getDisablePlayRate ()

배속이 불가능한 지 여부를 가져오는 함수

##### 반환값

true:배속 조절 불가능, false:배속 조절 가능

#### boolean com.kollus.sdk.media.content.KollusContent.getDisableTvOut ()

외부 출력을 못하도록 하게 설정이 되어 있는지 여부를 가져오는 함수

##### 반환값

true:외부 출력 허용 안됨, false:외부 출력 가능

#### boolean com.kollus.sdk.media.content.KollusContent.getDownloadError ()

다운로드 진행중에 에러가 발생했는 지를 알 수 있는 함수

##### 반환값

에러 여부

#### int com.kollus.sdk.media.content.KollusContent.getDownloadPercent ()

다운로드 진행률을 가져오는 함수

##### 반환값

다운로드 진행률

#### int com.kollus.sdk.media.content.KollusContent.getDuration ()

컨텐츠 길이를 가져오는 함수

##### 반환값

컨텐츠 길이(밀리초)

#### int com.kollus.sdk.media.content.KollusContent.getExpirationCount ()

다운로드 컨텐츠에 대해 남아 있는 플레이 가능 회수를 가져오는 함수

##### 반환값

남아 있는 재생 가능 회수

#### int com.kollus.sdk.media.content.KollusContent.getExpirationDate ()

다운로드 컨텐츠에 대해 재생 만료 시간을 가져오는 함수

##### 반환값

만료 시간

#### int com.kollus.sdk.media.content.KollusContent.getExpirationPlaytime ()

다운로드 컨텐츠에 대해 남아 있는 플레이 가능 시간을 가져오는 함수

##### 반환값

플레이 가능 시간(초)

#### boolean com.kollus.sdk.media.content.KollusContent.getExpirationRefreshPopup ()

만료된 컨텐츠에 대해 갱신 시 팝업창을 띄울 지 여부를 가져오는 함수

##### 반환값

팝업창을 띄울 지 여부

#### long com.kollus.sdk.media.content.KollusContent.getFileSize ()

다운로드가 완료된 컨텐츠의 파일 사이즈

##### 반환값

파일 사이즈

#### boolean com.kollus.sdk.media.content.KollusContent.getForceNScreen ()

이어보기 팝업창을 띄울 지 여부를 가져오는 함수

##### 반환값

팝업창 띄울 지 여부

#### String com.kollus.sdk.media.content.KollusContent.getForensicWaterMark ()

포렌식 워터 마크를 가져하는 함수

##### 반환값

포렌식 워터 마크

#### String com.kollus.sdk.media.content.KollusContent.getLicenseKey ()

Multi DRM 라이센스 키를 가져오는 함수

##### 반환값

라이센스 키

#### String com.kollus.sdk.media.content.KollusContent.getLicenseToken ()

Multi DRM 라이센스 토크을 가져오는 함수

##### 반환값

라이센스 토큰

#### String com.kollus.sdk.media.content.KollusContent.getLicenseUrl ()

Multi DRM 라이센스 URL를 가져오는 함수

##### 반환값

라이센스 URL

#### String com.kollus.sdk.media.content.KollusContent.getLocalMediaUrl ()

컨텐츠의 원래 경로를 가져오는 함수

##### 반환값

컨텐츠 주소

#### int com.kollus.sdk.media.content.KollusContent.getMaxPlaybackRate ()

최대 배속 가져오는 함수를 가져오는 함수

##### 반환값

최대 배속

#### String com.kollus.sdk.media.content.KollusContent.getMediaContentKey ()

미디어 컨텐츠 키를 가져오는 함수

##### 반환값

미디어 컨텐츠 키

#### String com.kollus.sdk.media.content.KollusContent.getMediaContentKeyMD5 ()

미디어 컨텐츠 키의 체크섬을 가져오는 함수

##### 반환값

미디어 컨텐츠 키의 체크섬

#### String com.kollus.sdk.media.content.KollusContent.getMediaUrl ()

컨텐츠 주소 가져오는 함수.

##### 반환값

컨텐츠 경로

#### boolean com.kollus.sdk.media.content.KollusContent.getMute ()

음소거인 지를 가져오는 함수

##### 반환값

음소거 여부

#### String com.kollus.sdk.media.content.KollusContent.getNextEpisodeCallbackData ()

다음회차 Callback URL 호출 시 post 데이터 가져오는 함수

##### 반환값

다음회차 Callback URL 호출 시 post 데이터

#### String com.kollus.sdk.media.content.KollusContent.getNextEpisodeCallbackUrl ()

다음회차 Callback URL 가져오는 함수

##### 반환값

다음회차 Callback URL

#### int com.kollus.sdk.media.content.KollusContent.getNextEpisodeShowTime ()

다음회차가 있는 경우 UI 버튼을 보여줄 시간 가져오는 함수

##### 반환값

보여줄 시간

#### String com.kollus.sdk.media.content.KollusContent.getOriginThumbnailPath ()

원래 설정된 대표 이미지를 가져오는 함수

#### int com.kollus.sdk.media.content.KollusContent.getPlaySectionEnd ()

재생 구간 종료 시간을 가져오는 함수

##### 반환값

재생 구간 종료 시간 (밀리초)

#### int com.kollus.sdk.media.content.KollusContent.getPlaySectionStart ()

재생 구간 시작 시간을 가져오는 함수

##### 반환값

재생 구간 시작 시간 (밀리초)

#### int com.kollus.sdk.media.content.KollusContent.getPlaytime ()

다운로드 컨텐츠에 대해 마지막 본 위치(밀리초)

##### 반환값

이어보기 정보(밀리초)

#### long com.kollus.sdk.media.content.KollusContent.getReceivedSize ()

다운로드된 컨텐츠의 파일 사이즈를 가져오는 함수 getCurrentReceiveSize와 비교하여 다운로드 bitrate으로 남은 시간을 계산하는데 사용하는 함수

##### 반환값

받은 사이즈

#### long com.kollus.sdk.media.content.KollusContent.getReceivingSize ()

현재까지 다운로드된 컨텐츠의 파일 사이즈

##### 반환값

다운로드 된 사이즈

#### int com.kollus.sdk.media.content.KollusContent.getRepeatPlaySectionEnd ()

반복 재생 종료 시간을 가져오는 함수

##### 반환값

반복 재생 종료 시간 (밀리초)

#### int com.kollus.sdk.media.content.KollusContent.getRepeatPlaySectionStart ()

반복 재생 시작 시간을 가져오는 함수

##### 반환값

반복 재생 시작 시간 (밀리초)

#### String com.kollus.sdk.media.content.KollusContent.getScreenShotPath ()

탐색 시 보여주는 스냅샵 이미지 주소 naming rule 패스.width.height.count.확장자 처음 3장은 의미없는 데이터임. 따라서 마지막 3장은 같은 데이터로 처리해야 한다. count는 최대 1200개임 duration이 1200초 이하면 초당 1장

##### 반환값

스크릿 샷 주소

#### boolean com.kollus.sdk.media.content.KollusContent.getSeekable ()

탐색이 가능한 지 여부를 알 수 있는 함수

##### 반환값

탐색 가능 여부

#### int com.kollus.sdk.media.content.KollusContent.getSeekableEnd ()

탐색이 가능한 시간을 가져오는 함수. 탐색이 불가능으로 설정되어 있는 경우라도 리턴 되는 값만큼은 탐색이 가능합니다.

##### 반환값

탐색이 가능한 시간(초)

#### String com.kollus.sdk.media.content.KollusContent.getServiceProviderMessage ()

KollusPlayerDRMListener의 onDRMInfo가 발생 시 서비스 프로바이더의 메세지를 확인할 수 있다.

##### 참고

**com.kollus.sdk.media.KollusPlayerDRMListener**

##### 반환값

String 서비스 프로바이더의 메세지

#### String com.kollus.sdk.media.content.KollusContent.getSkinString ()

플레이어 스킨을 가져오는 함수

##### 반환값

String JSON형식의 스킨 데이터

#### int com.kollus.sdk.media.content.KollusContent.getSkipSec ()

얼마 동안 seek가 안되지는 지 설정된 값을 가져오는 함수

##### 반환값

seek 불가능한 초. SDK내에서는 자동으로 Skip를 막지는 않습니다. 해당 초만큼 UI에서 Skip를 막아야 합니다.

#### int com.kollus.sdk.media.content.KollusContent.getStartAt ()

서버에 접속한 시간을 가져오는 시간

##### 반환값

접속한 UTC 시간

#### String com.kollus.sdk.media.content.KollusContent.getSubCourse ()

컨텐츠 제목을 가져오는 함수

##### 반환값

컨텐츠 제목

#### Vector< SubtitleInfo > com.kollus.sdk.media.content.KollusContent.getSubtitleInfo ()

자막 정보 리스트를 가져오는 함수

##### 반환값

자막 정보 리스트

#### Vector< SubtitleInfo > com.kollus.sdk.media.content.KollusContent.getSubtitleSubInfo ()

부가 자막 정보 리스트를 가져오는 함수

##### 반환값

자막 정보 리스트

#### String com.kollus.sdk.media.content.KollusContent.getSynopsis ()

사용 안함

##### 반환값

시놉시스

#### String com.kollus.sdk.media.content.KollusContent.getTeacher ()

사용 안함

##### 반환값

선생님 이름

#### String com.kollus.sdk.media.content.KollusContent.getThumbnailPath ()

썸네일 주소를 가져오는 함수

##### 반환값

썸네일 주소

#### int com.kollus.sdk.media.content.KollusContent.getTotalExpirationCount ()

다운로드 컨텐츠에 대해 원래 주어진 재생 가능 회수 가져오는 함수

##### 반환값

재생 가능 회수

#### int com.kollus.sdk.media.content.KollusContent.getTotalExpirationPlaytime ()

다운로드 컨텐츠에 대해 원래 주어진 재생 만료 시간을 가져오는 함수

##### 반환값

원래 주어진 재생 만료 시간(초)

#### String com.kollus.sdk.media.content.KollusContent.getUploadFileKey ()

업로드 파일 키를 가져오는 함수

##### 반환값

업로드 파일 키

#### int com.kollus.sdk.media.content.KollusContent.getUriIndex ()

컨텐츠 인덱스를 가져오는 함수

##### 반환값

컨텐츠 인덱스

#### int com.kollus.sdk.media.content.KollusContent.getVideoHeight ()

다운로드 컨텐츠의 세로 사이즈를 가져오는 함수

##### 반환값

세로 사이즈

#### int com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkAlpha ()

비디오 워터마크 투명도를 가져오는 함수

##### 반환값

비디오 워터마크 투명도

#### String com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkCode ()

비디오 워터마크 코드 가져오는 함수

##### 반환값

비디오 워터마크 코드

#### int com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkFontColor ()

비디오 워터마크 색상 가져오는 함수

##### 반환값

비디오 워터마크 색상

#### int com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkFontSize ()

비디오 워터마크 폰트 사이즈를 가져오는 함수

##### 반환값

폰트 크기

#### int com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkHideTime ()

비디오 워터마크 숨기는 시간을 가져오는 함수

##### 반환값

숨기는 시간(초)

#### int com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkShowTime ()

비디오 워터마크 보여주는 시간을 가져오는 함수

##### 반환값

보여주는 시간(초)

#### int com.kollus.sdk.media.content.KollusContent.getVideoWaterMarkSpace ()

비디오 워터마크 공간 분할 개수 가져오는 함수

##### 반환값

분할 개수

#### int com.kollus.sdk.media.content.KollusContent.getVideoWidth ()

다운로드 컨텐츠의 가로 사이즈를 가져오는 함수

##### 반환값

가로 사이즈

#### boolean com.kollus.sdk.media.content.KollusContent.hasAudioWaterMark ()

오디오 워터마크가 적용된 컨텐츠인 지 여부

##### 반환값

적용 여부

#### boolean com.kollus.sdk.media.content.KollusContent.hasForensicWaterMark ()

워터마크가 적용된 컨텐츠인 지 여부

##### 반환값

적용 여부

#### boolean com.kollus.sdk.media.content.KollusContent.isAudioFile ()

오디오 컨텐츠인 지 여부를 알 수 있는 함수

##### 반환값

true:오디오 파일, false:비디오도 있는 파일

#### boolean com.kollus.sdk.media.content.KollusContent.isCompleted ()

다운로드 완료 여부를 가져오는 함수

##### 반환값

true:다운로드 완료, false:다운로드 미완료

#### boolean com.kollus.sdk.media.content.KollusContent.isContentExpirated ()

다운로드 컨텐츠가 만료됐는 지 여부를 가져오는 함수

##### 반환값

컨텐츠 만료 여부

#### boolean com.kollus.sdk.media.content.KollusContent.isDownloadCanceled ()

다운로드 취소 여부를 가져오는 함수

##### 반환값

다운로드 취소여부

#### boolean com.kollus.sdk.media.content.KollusContent.isDownloading ()

다운로드 중인 지 가져오는 함수

##### 반환값

다운로드 중인 지 여부

#### boolean com.kollus.sdk.media.content.KollusContent.isDrm ()

다운로드 컨텐츠에 대해 drm 여부 확인

##### 반환값

drm 여부

#### boolean com.kollus.sdk.media.content.KollusContent.isIntro ()

인트로 영상인 지 여부를 알 수 있는 함수

##### 반환값

인트로 영상 여부

#### boolean com.kollus.sdk.media.content.KollusContent.IsInvalidFileSize ()

다운로드가 완료된 컨텐츠의 파일 사이즈 유효성

##### 반환값

true:유효한 파일, false:유효하지 않은 파일

#### boolean com.kollus.sdk.media.content.KollusContent.isLive ()

라이브 영상 여부를 가져오는 함수

##### 반환값

라이브 영상 여부

#### boolean com.kollus.sdk.media.content.KollusContent.isLoaded ()

다운로드를 위해 로드를 했는지 여부를 가져오는 함수

##### 반환값

로드 여부

#### boolean com.kollus.sdk.media.content.KollusContent.isSection ()

사용 안함

##### 반환값

#### boolean com.kollus.sdk.media.content.KollusContent.isThumbnailDownloadSync ()

load 시 썸네일 다운로드가 Sync(no thread)인지 여부를 가져오는 함수

##### 반환값

true:load시 썸네일 다운로드를 같이 함, false:썸네일 다운로드를 병렬 처리함

#### boolean com.kollus.sdk.media.content.KollusContent.isThumbnailEnable ()

썸네일 활성화 여부를 가져오는 함수

##### 반환값

썸네일 활성화 여부

#### boolean com.kollus.sdk.media.content.KollusContent.isVideoWaterMarkRandom ()

비디오 워터마크 표시가 랜덤인지 여부를 가져오는 함수

##### 반환값

랜덤 여부

#### boolean com.kollus.sdk.media.content.KollusContent.isVmCheck ()

가상머신에서 재생을 허용하는 여부를 가져오는 함수

##### 반환값

true:가상머신에서 재생 허용, false:가상머신에서 재생 허용 안함

#### boolean com.kollus.sdk.media.content.KollusContent.isVr ()

VR 영상 여부를 가져오는 함수

##### 반환값

VR 영상 여부

#### void com.kollus.sdk.media.content.KollusContent.setByJson (String *jsonString*)

Json String으로 컨텐츠 정보를 설정하는 함수 SDK 내부에서 사용

##### 매개변수

| *jsonString* | 컨텐츠 정볼를 담은 Json String |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setContentExpired (boolean *expired*)

다운로드 컨텐츠가 만료됐는 지 여부를 설정하는 함수

##### 매개변수

| *expired* | 컨텐츠 만료 여부 |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setContentType (ContentType *type*)

컨텐츠 타입을 설정하는 함수

##### 매개변수

| *type* | 컨텐츠 타입 |
| --- | --- |

##### 참고

ContentType

#### void com.kollus.sdk.media.content.KollusContent.setDownloadCanceled ()

다운로드 취소여부를 설정하는 함수 메세지로 주고 받기 때문에 다운로드 취소가 완료가 되어도 다운로드 진행률이 올라올 수 있음.

#### void com.kollus.sdk.media.content.KollusContent.setDownloadCompleted (boolean *complete*)

다운로드 완료 여부를 설정하는 함수

##### 매개변수

| *complete* | true:다운로드 완료, false:다운로드 미완료 |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setDownloadError (boolean *bError*)

다운로드 진행중에 에러가 발생했는 지를 설정하는 함수

##### 매개변수

| *bError* | 에러 여부 |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setDownloading (boolean *downloading*)

다운로드 중인 지 설정하는 함수

##### 매개변수

| *downloading* | 다운로드 중인 지 여부 |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setDownloadPercent (int *percent*)

다운로드 진행률을 설정하는 함수

##### 매개변수

| *percent* | 다운로드 진행률 |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setFileSize (long *fileSize*)

다운로드가 완료된 컨텐츠의 파일 사이즈 설정하는 함수

#### void com.kollus.sdk.media.content.KollusContent.setForDownload (Parcel *in*)

Parcel로 다운로드 컨텐츠 정보를 설정하는 함수 SDK 내부에서 사용

##### 매개변수

| *in* | 컨텐츠 정볼를 담은 Parcel |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setForPlay (Parcel *in*)

Parcel로 재생 컨텐츠 정보를 설정하는 함수 SDK 내부에서 사용

##### 매개변수

| *in* | 컨텐츠 정볼를 담은 Parcel |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setLoaded (boolean *bLoaded*)

다운로드를 위해 로드를 했는지 여부를 설정오는 함수

##### 매개변수

| *bLoaded* | 로드 여부 |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setLocalMediaUrl (String *mediaUrl*)

컨텐츠의 원래 경로를 설정하는 함수

#### void com.kollus.sdk.media.content.KollusContent.setMediaContentKey (String *key*)

미디어 컨텐츠 키를 설정하는 함수

##### 매개변수

| *key* | 미디어 컨텐츠 키 |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setPlaytime (int *playtime*)

이어보기 정보를 설정하는 함수

##### 매개변수

| *playtime* | 이어보기 정보(밀리초) |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setReceivedSize (long *receiveSize*)

다운로드된 컨텐츠의 파일 사이즈를 설정하는 함수 getCurrentReceiveSize와 비교하여 다운로드 bitrate으로 남은 시간을 계산하는데 사용하는 함수 계산 후 getReceivingSize로 setReceivedSize을 셋팅해야 합니다.

##### 매개변수

| *receiveSize* | 받은 사이즈 |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setReceivingSize (long *receiveSize*)

현재까지 다운로드 된 사이즈를 설정하는 함수

##### 매개변수

| *receiveSize* | 다운로드 된 사이즈 |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setServiceProviderMessage (String *message*)

KollusPlayerDRMListener의 onDRMInfo가 발생 시 서비스 프로바이더의 메세지를 설정하는 함수

##### 매개변수

| *message* | 서비스 프로바이더의 메세지 |
| --- | --- |

#### void com.kollus.sdk.media.content.KollusContent.setUriIndex (int *index*)

컨텐츠 인덱스를 설정하는 함수

##### 매개변수

| *index* | 컨텐츠 인덱스 |
| --- | --- |

#### String com.kollus.sdk.media.content.KollusContent.toString ()

KollusContent의 내용을 문자화하는 함수

##### 반환값

KollusContent의 내용

#### boolean com.kollus.sdk.media.content.KollusContent.useBlockEmulator ()

에뮬레이터를 사용하고 있는지 여부를 가져오는 함수

##### 반환값

에뮬레이터 사용 여부

### 멤버 데이터 문서화

**final Parcelable.Creator<KollusContent> com.kollus.sdk.media.content.KollusContent.CREATOR[static]**

**초기값:**= new Parcelable.Creator<KollusContent>() {

@Override

public KollusContent createFromParcel(Parcel in) {

return new KollusContent(in);

}

@Override

public KollusContent[] newArray(int size) {

return new KollusContent[size];

}

}

생성자. SDK내에서 사용합니다.

## com.kollus.sdk.media.KollusPlayerBookmarkListener 인터페이스 참조

com.kollus.sdk.media.KollusPlayerBookmarkListener**Public 멤버 함수**

void **onBookmark** (List< **KollusBookmark** > bookmark, boolean bWritable)

void **onGetBookmarkError** (int nErrorCode)

void **onBookmarkUpdated** (int position, boolean bUpdated)

void **onBookmarkDeleted** (int position, boolean bDeleted)

### 멤버 함수 문서화

**void com.kollus.sdk.media.KollusPlayerBookmarkListener.onBookmark (List< KollusBookmark > *bookmark*, boolean *bWritable*)**

강사용 북마크나 사용자 북마크가 있을 경우 리스닝 됩니다.

##### 매개변수

| *bookmark* | 북마크 리스트 |
| --- | --- |
| *bWritable* | false인 경우 사용자 북마크를 할 수 없는 컨텐츠입니다. |

#### void com.kollus.sdk.media.KollusPlayerBookmarkListener.onBookmarkDeleted (int *position*, boolean *bDeleted*)

북마크가 삭제되었을 때 리스닝됩니다. 삭제 액션후 네트워크 상황에 따라 오랜 시간후에 리스닝될 수 있습니다.

##### 매개변수

| *position* | 삭제할 초단위 위치 |
| --- | --- |
| *bDeleted* | 삭제 성공 여부 |

#### void com.kollus.sdk.media.KollusPlayerBookmarkListener.onBookmarkUpdated (int *position*, boolean *bUpdated*)

북마크를 추가하거나 갱신되었을 때 리스닝됩니다. 추가 액션후 네트워크 상황에 따라 오랜 시간후에 리스닝될 수 있습니다.

##### 매개변수

| *position* | 갱신할 초단위 위치 |
| --- | --- |
| *bUpdated* | 갱신 성공 여부 |

#### void com.kollus.sdk.media.KollusPlayerBookmarkListener.onGetBookmarkError (int *nErrorCode*)

네트워크 오류, JSON 데이터 파싱 오류, JSON error 필드가 0이 아닐 때 발생합니다.

## com.kollus.sdk.media.KollusPlayerCallbackListener 인터페이스 참조

com.kollus.sdk.media.KollusPlayerCallbackListener**Public 멤버 함수**

void **onCallbackMessage** (String request, String response)

### 멤버 함수 문서화

**void com.kollus.sdk.media.KollusPlayerCallbackListener.onCallbackMessage (String *request*, String *response*)**

DRM 관련하여 서버에 전송한 데이터와 그 결과값을 String으로 응답받는 함수

##### 매개변수

| *request* | 서버에 전송한 데이터 |
| --- | --- |
| *response* | 서버로부터 응답받은 데이터 |

## com.kollus.sdk.media.KollusPlayerContentMode 클래스 참조

com.kollus.sdk.media.KollusPlayerContentMode**정적 Public 속성**

static final int **ScaleAspectFit** = 0

static final int **ScaleAspectFill** = 1

static final int **ScaleAspectFillStretch** = 2

static final int **ScaleCenter** = 3

static final int **ScaleZoom** = 4

### 멤버 데이터 문서화

**final int com.kollus.sdk.media.KollusPlayerContentMode.ScaleAspectFill = 1[static]**

전체 화면 (비율 유지)

#### final int com.kollus.sdk.media.KollusPlayerContentMode.ScaleAspectFillStretch = 2[static]

전체 화면 (비율 무시)

#### final int com.kollus.sdk.media.KollusPlayerContentMode.ScaleAspectFit = 0[static]

화면 비율 유지

#### final int com.kollus.sdk.media.KollusPlayerContentMode.ScaleCenter = 3[static]

원본 사이즈

#### final int com.kollus.sdk.media.KollusPlayerContentMode.ScaleZoom = 4[static]

사용자 비율

## com.kollus.sdk.media.KollusPlayerDRMListener 인터페이스 참조

com.kollus.sdk.media.KollusPlayerDRMListener**Public 멤버 함수**

void **onDRM** (String request, String response)

void **onDRMInfo** (**KollusContent** content, int nInfoCode)

### 정적 Public 속성

static final int **DCB\_INFO\_DELETE** = 0

static final int **DCB\_INFO\_EXPIRED** = 1

static final int **DCB\_INFO\_RESET** = 2

### 멤버 함수 문서화

**void com.kollus.sdk.media.KollusPlayerDRMListener.onDRM (String *request*, String *response*)**

DRM 관련하여 서버에 전송한 데이터와 그 결과값을 String으로 응답받는 함수

##### 매개변수

| *request* | 서버에 전송한 데이터 |
| --- | --- |
| *response* | 서버로부터 응답받은 데이터 |

#### void com.kollus.sdk.media.KollusPlayerDRMListener.onDRMInfo (KollusContent *content*, int *nInfoCode*)

DRM 콜백 Kind 응답 정보(컨텐츠 삭제, 컨텐츠 수정, 컨텐츠 reset등)

##### 매개변수

| *content* | 컨텐트 정보 |
| --- | --- |
| *nInfoCode* | DCB\_INFO\_DELETE, DCB\_INFO\_EXPIRED, DCB\_INFO\_RESET중 하나의 값 |

### 멤버 데이터 문서화

**final int com.kollus.sdk.media.KollusPlayerDRMListener.DCB\_INFO\_DELETE = 0[static]**

load시에 DRM 콜백 KIND 2, 3로 컨텐츠 삭제 요청을 받고, 실제 컨텐츠 관련 삭제후 알림

#### final int com.kollus.sdk.media.KollusPlayerDRMListener.DCB\_INFO\_EXPIRED = 1[static]

KIND 3에 대한 응답으로 강제 만료를 받고 강제 만료 정보를 컨텐츠에 저장후 알림

#### final int com.kollus.sdk.media.KollusPlayerDRMListener.DCB\_INFO\_RESET = 2[static]

KIND 3에 대한 응답으로 컨텐츠 reset시에 reset 정보 컨텐츠에 저장후 알림

## com.kollus.sdk.media.KollusPlayerDRMUpdateListener 인터페이스 참조

com.kollus.sdk.media.KollusPlayerDRMUpdateListener**Public 멤버 함수**

void **onDRMUpdateStart** ()

void **onDRMUpdateProcess** (String request, String response)

void **onDRMUpdateComplete** ()

### 멤버 함수 문서화

**void com.kollus.sdk.media.KollusPlayerDRMUpdateListener.onDRMUpdateComplete ()**

DRM 갱신이 완료됐을 때 알려주는 함수

#### void com.kollus.sdk.media.KollusPlayerDRMUpdateListener.onDRMUpdateProcess (String *request*, String *response*)

현재 DRM정보 갱신 진행상황을 알려주는 함수 갱신의 경우 void KollusPlayerDRMListener.onDRM(String request, String response)대신에 이 함수를 통해 전달합니다.

##### 매개변수

| *request* | 서버에 전송한 데이터 |
| --- | --- |
| *response* | 서버에서 전송 받은 데이터 |

#### void com.kollus.sdk.media.KollusPlayerDRMUpdateListener.onDRMUpdateStart ()

DRM 갱신 시작을 알려주는 함수

## com.kollus.sdk.media.KollusPlayerLMSListener 인터페이스 참조

com.kollus.sdk.media.KollusPlayerLMSListener**Public 멤버 함수**

void **onLMS** (String request, String response)

### 멤버 함수 문서화

**void com.kollus.sdk.media.KollusPlayerLMSListener.onLMS (String *request*, String *response*)**

진도률을 서버에 전송했을 때 관련 정보

##### 매개변수

| *request* | 서버에 전송한 메세지 |
| --- | --- |
| *response* | 서버에서 응답 받은 메세지 |

## com.kollus.sdk.media.KollusPlayerThumbnailListener 인터페이스 참조

com.kollus.sdk.media.KollusPlayerThumbnailListener**Public 멤버 함수**

void **onCached** (int index, int nErrorCode, String savePath)

### 멤버 함수 문서화

**void com.kollus.sdk.media.KollusPlayerThumbnailListener.onCached (int *index*, int *nErrorCode*, String *savePath*)**

썸네일이 다운로드 방식이 쓰레드 방식일 때 호출

##### 매개변수

| *index* | KollusContent의 getUriIndex 값 |
| --- | --- |
| *nErrorCode* | 에러값 |
| *savePath* | 저장된 경로 |

## com.kollus.sdk.media.KollusStorage 클래스 참조

com.kollus.sdk.media.KollusStorage**클래스**

interface **DrmSessionManagerListener**interface **OnKollusStorageDeviceListener**

interface **OnKollusStorageListener**

### Public 멤버 함수

void **registerDrmSessionManagerListenerList** (**DrmSessionManagerListener** listener)

void **removeDrmSessionManagerListenerList** (**DrmSessionManagerListener** listener)

**KollusStorage** (Context context)

synchronized void **releaseInstance** ()

long **getNativeInstance** ()

String **getVersion** ()

String **checkVersion** ()

int **initialize** (String key, String expireDate, String packageName)

int **checkLiveStream** (String url)

void **finish** ()

boolean **isReady** ()

int **setDevice** (String storagePath, String playerId, String playerIdMd5, boolean bTablet)

void **setDeviceASync** (String storagePath, String playerId, String playerIdMd5, boolean bTablet, **OnKollusStorageDeviceListener** listener)

String **getRootPath** ()

ArrayList< **KollusContent** > **getDownloadContentList** ()

boolean **getKollusContent** (**KollusContent** content, String mediaContentKey) throws IllegalStateException

void **updateDownloadDRMInfo** (**KollusPlayerDRMUpdateListener** listener, boolean bAll)

void **updateDownloadDRMInfo** (**KollusPlayerDRMUpdateListener** listener, ArrayList< String > mckList)

long **getUsedSize** (int type)

void **clearCache** ()

String **decrypt** (String encData)

int **load** (String path, String extraDrmParam, **KollusContent** content)

**KollusContent** **getDownloadKollusContent** (String path)

void **cancelLoad** ()

int **unload** (String mediaContentKey)

int **download** (String mediaContentKey)

String **getLastError** ()

int **remove** (String mediaContentKey)

void **setNetworkTimeout** (int timeout, int retryCount)

void **setCacheSize** (int cacheSizeMB)

void **registerKollusStorageListener** (**OnKollusStorageListener** listener)

void **unregisterKollusStorageListener** (**OnKollusStorageListener** listener)

void **registerKollusPlayerDRMListener** (**KollusPlayerDRMListener** listener)

void **unregisterKollusPlayerDRMListener** (**KollusPlayerDRMListener** listener)

void **registerKollusPlayerThumbnailListener** (**KollusPlayerThumbnailListener** listener)

void **unregisterKollusPlayerThumbnailListener** (**KollusPlayerThumbnailListener** listener)

void **registerKollusPlayerCallbackListener** (**KollusPlayerCallbackListener** listener)

void **unregisterKollusPlayerCallbackListener** (**KollusPlayerCallbackListener** listener)

void **sendStoredLMS** (StoredLMSListener listener)

### 정적 Public 멤버 함수

static synchronized **KollusStorage** **getInstance** (Context context)

### 정적 Public 속성

static final int **TYPE\_CACHE** = 0

static final int **TYPE\_DOWNLOAD** = 1

static final int **TYPE\_ALL** = 2

### Protected 멤버 함수

void **sendCallback** (String customer, String url)

String **getNoKollusStreamPath** ()

String **getNoKollusDownloadPath** ()

int **downloadFile** (String srcPath, String dstPath)

int **downloadThumbnail** (String srcPath, String dstPath)

String **getMediaInfoJson** (Uri uri)

void **registerKollusPlayerLMSListener** (**KollusPlayerLMSListener** listener)

void **unregisterKollusPlayerLMSListener** (**KollusPlayerLMSListener** listener)

String **getCallback** (String customer, String callbackUrl, String postData)

### 멤버 함수 문서화

**void com.kollus.sdk.media.KollusStorage.cancelLoad ()**

스토리지 매니져가 load 작업을 취소할 때 호출 다운로드 취소 시 호출 cancelLoad -> unload

#### int com.kollus.sdk.media.KollusStorage.checkLiveStream (String *url*)

해당 경로로 현재 라이브 방송인 지 체크하는 함수

##### 매개변수

| *url* | 라이브 방송 URL |
| --- | --- |

##### 반환값

#### String com.kollus.sdk.media.KollusStorage.checkVersion ()

라이브러리 버전 정보. 더이상 사용하지 않습니다.

##### 반환값

버전

#### void com.kollus.sdk.media.KollusStorage.clearCache ()

스트리밍 재생 시에 사용한 데이터 삭제

#### int com.kollus.sdk.media.KollusStorage.download (String *mediaContentKey*)

다운로드를 위해 load한 KollusContent의 MediaContentKey를 가지고 다운로드를 시작한다.

##### 매개변수

| *mediaContentKey* | 미디어 컨텐트 키 |
| --- | --- |

##### 참고

ErrorCodes

##### 반환값

에러코드

#### void com.kollus.sdk.media.KollusStorage.finish ()

native 스토리지 매니져 자원 회수. 호출 시점이 모호하면 굳이 호출하지 않아도 됩니다.

#### ArrayList< KollusContent > com.kollus.sdk.media.KollusStorage.getDownloadContentList ()

다운로드 컨텐츠 리스트 정보를 가져온다.

##### 반환값

컨텐츠 리스트

#### KollusContent com.kollus.sdk.media.KollusStorage.getDownloadKollusContent (String *path*)

해당 패스로 다운로드 컨텐츠 정보를 얻어오는 함수

##### 반환값

컨텐츠 정보

#### static synchronized KollusStorage com.kollus.sdk.media.KollusStorage.getInstance (Context *context*)[static]

Singleton 생성자

##### 매개변수

| *context* | App Context |
| --- | --- |

##### 반환값

KollusStorage 인스턴스

#### boolean com.kollus.sdk.media.KollusStorage.getKollusContent (KollusContent *content*, String *mediaContentKey*) throws IllegalStateException

재생 시에 해당 컨텐츠 정보를 알기 원할 때 사용한다.

##### 반환값

성공적으로 컨텐츠 정보를 가져왔는 지 여부

#### String com.kollus.sdk.media.KollusStorage.getLastError ()

에러 발생 시 마지막 에러 메세지를 가져온다.

##### 반환값

에러 메세지

#### String com.kollus.sdk.media.KollusStorage.getMediaInfoJson (Uri *uri*)[protected]

해당 uri로 미디어 정보를 가져오는 함수로 SDK 내부에서만 사용됩니다.

##### 매개변수

| *uri* | 정보를 가져오는 URI |
| --- | --- |

##### 반환값

미디어 정보

#### long com.kollus.sdk.media.KollusStorage.getNativeInstance ()

KollusStorage int형 인스턴스

##### 반환값

#### String com.kollus.sdk.media.KollusStorage.getRootPath ()

저장하는 위치를 가져온다. setDevice(setDeviceAsync)에서 넣어준 저장위치와 동일합니다.

##### 반환값

저장위치

#### long com.kollus.sdk.media.KollusStorage.getUsedSize (int *type*)

해당 타입의 총 용량

##### 매개변수

| *type* | TYPE\_CACHE - 스트리밍 재생에 사용된 캐쉬 용량 TYPE\_DOWNLOAD - 다운로드된 컨텐츠 총 용량 |
| --- | --- |

##### 반환값

용량

#### String com.kollus.sdk.media.KollusStorage.getVersion ()

라이브러리 버전 정보

##### 반환값

버전

#### int com.kollus.sdk.media.KollusStorage.initialize (String *key*, String *expireDate*, String *packageName*)

초기화 함수

##### 매개변수

| *key* | 발급된 키 |
| --- | --- |
| *expireDate* | 키와 같이 발급된 만료기한(YYYY/MM/DD) |

#### boolean com.kollus.sdk.media.KollusStorage.isReady ()

setDevice(setDeviceASync)가 호출이 성공적으로 되어서 사용할 준비가 되었는 지 여부를 알려줍니다.

#### int com.kollus.sdk.media.KollusStorage.load (String *path*, String *extraDrmParam*, KollusContent *content*)

컨텐츠 다운로드 시 호출

##### 매개변수

| *path* | 다운로드할 컨텐츠 경로 |
| --- | --- |
| *extraDrmParam* | 동적 DRMCallback/PlayCallback 인자 |

##### 참고

ErrorCodes

##### 반환값

에러 값

#### void com.kollus.sdk.media.KollusStorage.registerKollusPlayerCallbackListener (KollusPlayerCallbackListener *listener*)

KollusPlayCallbackListener 등록 함수

##### 매개변수

| *listener* | KollusPlayerCallbackListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.KollusStorage.registerKollusPlayerDRMListener (KollusPlayerDRMListener *listener*)

KollusPlayerDRMListener 등록 함수

##### 매개변수

| *listener* | KollusPlayerDRMListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.KollusStorage.registerKollusPlayerLMSListener (KollusPlayerLMSListener *listener*)[protected]

KollusPlayerLMSListener 등록 함수

##### 매개변수

| *listener* | KollusPlayerLMSListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.KollusStorage.registerKollusPlayerThumbnailListener (KollusPlayerThumbnailListener *listener*)

KollusPlayerThumbnailListener 등록 함수

##### 매개변수

| *listener* | KollusPlayerThumbnailListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.KollusStorage.registerKollusStorageListener (OnKollusStorageListener *listener*)

OnStorageManagerListener 등록 함수

##### 매개변수

| *listener* | OnStorageManagerListener 인스턴스 |
| --- | --- |

#### synchronized void com.kollus.sdk.media.KollusStorage.releaseInstance ()

더이상 사용하지 않을 경우 KollusStorage를 해제하는 함수. 호출 시점이 모호하면 굳이 호출하지 않아도 됩니다.

#### int com.kollus.sdk.media.KollusStorage.remove (String *mediaContentKey*)

다운로드된 컨텐츠를 삭제 시 호출한다.

##### 매개변수

| *mediaContentKey* | 미디어 컨텐트 키 |
| --- | --- |

##### 참고

ErrorCodes

##### 반환값

에러코드

#### void com.kollus.sdk.media.KollusStorage.sendStoredLMS (StoredLMSListener *listener*)

보내지 못한 LMS를 한꺼번에 보내는 함수

##### 매개변수

| *listener* | StoredLMSListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.KollusStorage.setCacheSize (int *cacheSizeMB*)

스트리밍 시에 사용될 캐시 사이즈를 설정합니다. setDevice나 setDeviceASync전에 호출해야 합니다.

##### 매개변수

| *cacheSizeMB* | MB단위의 캐시 사이즈 |
| --- | --- |

#### int com.kollus.sdk.media.KollusStorage.setDevice (String *storagePath*, String *playerId*, String *playerIdMd5*, boolean *bTablet*)

디바이스 정보를 설정하는 함수

##### 매개변수

| *storagePath* | 스토리지 매니져가 사용할 루트 패스 |
| --- | --- |
| *playerId* | player id |
| *playerIdMd5* | player id의 MD5 |

#### void com.kollus.sdk.media.KollusStorage.setDeviceASync (String *storagePath*, String *playerId*, String *playerIdMd5*, boolean *bTablet*, OnKollusStorageDeviceListener *listener*)

디바이스 정보를 설정하는 함수

##### 매개변수

| *storagePath* | 스토리지 매니져가 사용할 루트 패스를 설정하는 함수로 ASync하게 동작합니다. |
| --- | --- |
| *playerId* | player id의 md5 값 Utils.createUUIDMD5 |
| *playerIdMd5* | player id의 sha1 값 Utils.createUUIDSHA1 |

#### void com.kollus.sdk.media.KollusStorage.setNetworkTimeout (int *timeout*, int *retryCount*)

네트워크 타임아웃과 재시도 회수를 설정합니다.

##### 매개변수

| *timeout* | 타임 아웃 초 |
| --- | --- |
| *retryCount* | 타임 아웃 시 재시도 회수 |

#### int com.kollus.sdk.media.KollusStorage.unload (String *mediaContentKey*)

load된 index unload한다. 다운로드 취소 시 cancelLoad호출 후 unload를 호출한다.

##### 매개변수

| *mediaContentKey* | 미디어 컨텐트 키 |
| --- | --- |

##### 참고

ErrorCodes

##### 반환값

0보다 작으면 에러 아니면 컨텐츠 인덱스

#### void com.kollus.sdk.media.KollusStorage.unregisterKollusPlayerCallbackListener (KollusPlayerCallbackListener *listener*)

KollusPlayCallbackListener 해제 함수

##### 매개변수

| *listener* | KollusPlayerCallbackListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.KollusStorage.unregisterKollusPlayerDRMListener (KollusPlayerDRMListener *listener*)

KollusPlayerDRMListener 해제 함수

##### 매개변수

| *listener* | KollusPlayerDRMListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.KollusStorage.unregisterKollusPlayerLMSListener (KollusPlayerLMSListener *listener*)[protected]

KollusPlayerLMSListener 해제 함수

##### 매개변수

| *listener* | KollusPlayerLMSListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.KollusStorage.unregisterKollusPlayerThumbnailListener (KollusPlayerThumbnailListener *listener*)

KollusPlayerThumbnailListener 해제 함수

##### 매개변수

| *listener* | KollusPlayerThumbnailListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.KollusStorage.unregisterKollusStorageListener (OnKollusStorageListener *listener*)

OnStorageManagerListener 해제 함수

##### 매개변수

| *listener* | OnStorageManagerListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.KollusStorage.updateDownloadDRMInfo (KollusPlayerDRMUpdateListener *listener*, ArrayList< String > *mckList*)

DRM 일관 갱신을 중지하는 함수 DRM 갱신을 미디어 컨텐츠 키 리스트를 받아서 하는 함수

##### 매개변수

| *listener* | 갱신 상황을 알려주는 인터페이스 |
| --- | --- |
| *mckList* | 갱신할 미디어컨텐츠키 리스트입니다. |

#### void com.kollus.sdk.media.KollusStorage.updateDownloadDRMInfo (KollusPlayerDRMUpdateListener *listener*, boolean *bAll*)

DRM 일괄 갱신을 하는 함수

##### 매개변수

| *listener* | 갱신 상황을 알려주는 인터페이스 |
| --- | --- |
| *bAll* | true이면 모든 컨텐츠를 갱신하고, false이면 만료된 컨텐츠만 갱신합니다. |

### 멤버 데이터 문서화

**final int com.kollus.sdk.media.KollusStorage.TYPE\_CACHE = 0[static]**

스트리밍 타입 컨텐츠

#### final int com.kollus.sdk.media.KollusStorage.TYPE\_DOWNLOAD = 1[static]

다운로드 타입 컨텐츠

## com.kollus.sdk.media.MediaPlayer 클래스 참조

com.kollus.sdk.media.MediaPlayer**클래스**

enum **EXTERNAL\_DISPLAY\_MODE**enum **MediaRouteState**

interface **OnBufferingUpdateListener**

interface **OnCencDrmListener**

interface **OnCompletionListener**

interface **OnErrorListener**

interface **OnExternalDisplayDetectListener**

interface **OnInfoListener**

interface **OnMediaRouteInfoListener**

interface **OnPreparedListener**

interface **OnSeekCompleteListener**

interface **OnTimedTextDetectListener**

interface **OnTimedTextListener**

interface **OnVideoSizeChangedListener**

enum **RENDER\_MODE**

class **TMCMReceiver**

class **TrackInfo**

### Public 멤버 함수

**MediaPlayer** (Context context, **KollusStorage** storage, int port)

void **setRemoteMediaRouteInfo** (String castNameSpace)

void **setDataSourceByUrl** (String path, String extraDrmParam)

void **setDataSourceByKey** (String mediaContentKey, String extraDrmParam)

void **setOnPreparedListener** (**OnPreparedListener** listener)

void **setOnCompletionListener** (**OnCompletionListener** listener)

void **setOnErrorListener** (**OnErrorListener** listener)

void **setOnInfoListener** (**OnInfoListener** listener)

void **setOnMediaRouteInfoListener** (**OnMediaRouteInfoListener** listener)

void **setOnCencDrmListener** (**OnCencDrmListener** listener)

void **setOnBufferingUpdateListener** (**OnBufferingUpdateListener** listener)

void **setOnSeekCompleteListener** (**OnSeekCompleteListener** listener)

void **setOnVideoSizeChangedListener** (**OnVideoSizeChangedListener** listener)

void **setOnTimedTextDetectListener** (**OnTimedTextDetectListener** listener)

void **setOnTimedTextListener** (**OnTimedTextListener** listener)

void **setOnExternalDisplayDetectListener** (**OnExternalDisplayDetectListener** listener)

void **setKollusPlayerBookmarkListener** (**KollusPlayerBookmarkListener** listener)

void **setKollusPlayerLMSListener** (**KollusPlayerLMSListener** listener)

void **setCaptureDetectLister** (CaptureDetectLister listener)

void **setEmulatorCheckerListener** (**EmulatorCheckerListener** listener)

void **setDisplay** (SurfaceHolder sh)

void **setSurface** (Surface surface)

void **destroyDisplay** ()

void **setForensicWatermarkView** (**ForensicWatermarkView** view)

String **getVersion** ()

void **prepareAsync** ()

void **start** () throws IllegalStateException

void **stop** () throws IllegalStateException

void **pause** () throws IllegalStateException

void **setScreenOnWhilePlaying** (boolean screenOn)

int **getVideoWidth** ()

int **getVideoHeight** ()

int **getPlayAt** ()

String **getErrorString** (int errorCode)

boolean **isPlaying** ()

void **seekTo** (int msec) throws IllegalStateException

void **seekToExact** (int msec) throws IllegalStateException

int **getCurrentPosition** ()

int **getDuration** ()

void **release** ()

void **releaseWhenFinishing** ()

void **setLooping** (boolean looping)

void **setVolumeLevel** (int level)

void **setMute** (boolean mute)

boolean **isLooping** ()

MediaPlayerBase.TrackInfo[] **getTrackInfo** () throws IllegalStateException

void **updateKollusBookmark** (int position, String label) throws IllegalStateException

void **deleteKollusBookmark** (int position) throws IllegalStateException

boolean **getKollusContent** (**KollusContent** content) throws IllegalStateException

void **addTimedTextSource** (boolean bSubTimedText, String path) throws IOException, IllegalArgumentException, IllegalStateException, RuntimeException

void **addTimedTextSource** (Context context, boolean bSubTimedText, Uri uri) throws IOException, IllegalArgumentException, IllegalStateException

void **selectTrack** (boolean bSubTimedText, int index) throws IllegalStateException

void **deselectTrack** (boolean bSubTimedText, int index) throws IllegalStateException

boolean **setPlayingRate** (float playing\_rate) throws IllegalStateException

String **getVideoCodecName** ()

int **getCachedDuration** ()

boolean **setNetworkTimeout** (int timeout)

boolean **setBufferingRatio** (int ratio)

void **skip** ()

void **setVideoRending** (boolean bRendering)

String **getPlayerName** ()

int **getPlayerType** ()

boolean **supportPlaybackrateControl** ()

void **setLmsOffDownloadContent** (boolean bOn)

void **setNotifyLastReport** (boolean bOn)

void **setAudioDelay** (int timeMs)

void **setAiPlaybackRateEnable** (boolean enable)

boolean **isAiPlaybackRateEnabled** ()

void **setLastOffsetTime** (int lastOffsetTimeMs)

int **getDownloadRate** ()

int **getBandwidth** ()

void **setBandwidth** (int bandwidth)

void **setBandwidthName** (String bandwidthName)

void **setInitialBandwidth** (int bandwidth)

void **setMinDurationForQualityIncreaseMs** (int msec)

### 정적 Public 속성

static final String **VERSION** = "2.0.19\_r3"

static final int **MEDIA\_ERROR\_UNKNOWN** = 1

static final int **RENDER\_MODEL** = 0

static final int **RENDER\_2\_POWER** = 1

static final int **RENDER\_2\_MULTIPLE** = 2

static final int **RENDER\_4\_MULTIPLE** = 3

static final int **RENDER\_16\_MULTIPLE** = 4

static final int **RENDER\_32\_MULTIPLE** = 5

static final int **MEDIA\_INFO\_FRAME\_RATE** = 7

static final int **MEDIA\_ERROR\_NOT\_VALID\_FOR\_PROGRESSIVE\_PLAYBACK** = 200

static final int **QUICK** = 1

static final int **EXACT** = 2

static final String **CAST\_NAME\_SPACE** = "urn:x-cast:com.kollus.cast"

static final String **CAST\_APP\_ID** = "E956F6B0"

### Protected 멤버 함수

void **setKollusStorage** (**KollusStorage** storage)

### 생성자 & 소멸자 문서화

**com.kollus.sdk.media.MediaPlayer.MediaPlayer (Context *context*, KollusStorage *storage*, int *port*)**

플레이어 생성 클래스

##### 매개변수

| *context* | Context 인스턴스 |
| --- | --- |
| *storage* | KollusStorage 인스턴스 |
| *port* | 내장 플레이어로 생성될 때 필요한 서버 포트 번호 |

### 멤버 함수 문서화

**void com.kollus.sdk.media.MediaPlayer.addTimedTextSource (boolean *bSubTimedText*, String *path*) throws IOException, IllegalArgumentException, IllegalStateException, RuntimeException**

HTTP Url로 자막을 설정하는 함수

##### 매개변수

| *bSubTimedText* | 서브 자막 유무 |
| --- | --- |
| *path* | 자막 http url 경로 |

##### 참고

MediaPlayerBase.OnTimedTextDetectListener

#### void com.kollus.sdk.media.MediaPlayer.addTimedTextSource (Context *context*, boolean *bSubTimedText*, Uri *uri*) throws IOException, IllegalArgumentException, IllegalStateException

Uri로 자막을 설정하는 함수

##### 매개변수

| *context* | App Context |
| --- | --- |
| *bSubTimedText* | 서브 자막 유무 |
| *uri* | 자막 Uri 경로 |

#### void com.kollus.sdk.media.MediaPlayer.deleteKollusBookmark (int *position*) throws IllegalStateException

북마크 정보를 삭제하는 함수 onBookmarkInfoDeleted로 성공여부가 리턴된다.

##### 매개변수

| *position* | 삭제할 북마크 시간(밀리초) |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.deselectTrack (boolean *bSubTimedText*, int *index*) throws IllegalStateException

selectTrack의 반대되는 함수

##### 매개변수

| *index* | 트랙 인덱스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.destroyDisplay ()

비디오 출력 화면이 Destroy됐을 때 호출하는 함수

#### int com.kollus.sdk.media.MediaPlayer.getBandwidth ()

Mpeg-Dash에서 ABR인 경우 현재 다운로드하고 있는 Bandwidth

##### 반환값

현재 다운로드하고 있는 Bandwidth

#### int com.kollus.sdk.media.MediaPlayer.getCachedDuration ()

스트리밍인 경우 현재 캐싱된 Duration일 밀리초 단위로 가져오는 함수

#### int com.kollus.sdk.media.MediaPlayer.getCurrentPosition ()

현재 재생 위치를 밀리초 단위로 가져오는 함수

##### 반환값

현재 재생하고 있는 시간(밀리초)

#### int com.kollus.sdk.media.MediaPlayer.getDownloadRate ()

Mpeg-Dash에서 다운로드 속도를 가져오는 함수

##### 반환값

다운로드 속도

#### int com.kollus.sdk.media.MediaPlayer.getDuration ()

현재 재생 중인 컨텐츠의 duration정보를 밀리초 단위로 가져오는 함수

##### 반환값

컨텐츠의 총 시간(밀리초)

#### String com.kollus.sdk.media.MediaPlayer.getErrorString (int *errorCode*)

errorCode를 String으로 가져오는 함수이나 의미있는 값을 전달하지는 않는다. \*

##### 매개변수

| *errorCode* | 에러 코드 |
| --- | --- |

##### 반환값

에러 스트링

#### boolean com.kollus.sdk.media.MediaPlayer.getKollusContent (KollusContent *content*) throws IllegalStateException

컨텐츠 정보를 가져오는 함수

##### 매개변수

| *content* | 컨텐츠 정보를 가져올 인스턴스 |
| --- | --- |

##### 반환값

true:성공적으로 정보를 가져온 경우, false:실패한 경우

#### int com.kollus.sdk.media.MediaPlayer.getPlayAt ()

이어보기 정보를 가져오는 함수로 N-Screen과 연관된다.

##### 반환값

이어보기 시간(밀리초)

#### String com.kollus.sdk.media.MediaPlayer.getPlayerName ()

MediaPlayerFactory에서 만들어진 플레이어의 이름을 가져옵니다.

##### 반환값

플레이어 이름

#### int com.kollus.sdk.media.MediaPlayer.getPlayerType ()

MediaPlayerFactory에서 만들어진 플레이어의 타입을 가져옵니다.

##### 반환값

플레이어 타입(Utils.PLAYER\_TYPE\_NATIVE, Utils.PLAYER\_TYPE\_KOLLUS, Utils.PLAYER\_TYPE\_EXO)

#### MediaPlayerBase.TrackInfo[] com.kollus.sdk.media.MediaPlayer.getTrackInfo () throws IllegalStateException

현재 재생 중인 컨텐츠의 트랙정보를 가져오는 함수

#### String com.kollus.sdk.media.MediaPlayer.getVersion ()

플레이어 버전을 가져오는 함수

##### 반환값

version 플레이어 버전

#### String com.kollus.sdk.media.MediaPlayer.getVideoCodecName ()

비디오 코덱 이름을 가져오는 함수로 HW코덱 연동을 확인하기 위해 사용

##### 반환값

코덱 이름

#### int com.kollus.sdk.media.MediaPlayer.getVideoHeight ()

컨텐츠의 height를 가져오는 함수 onPrepared 응답을 받은 후 호출할 수 있다.

##### 반환값

비디오 세로 사이즈

#### int com.kollus.sdk.media.MediaPlayer.getVideoWidth ()

컨텐츠의 width를 가져오는 함수 onPrepared 응답을 받은 후 호출할 수 있다.

##### 반환값

비디오 가로 사이즈

#### boolean com.kollus.sdk.media.MediaPlayer.isAiPlaybackRateEnabled ()

AI 배속 여부를 알려주는 함수

##### 반환값

true : AI 배속, false : 일반 배속

#### boolean com.kollus.sdk.media.MediaPlayer.isLooping ()

현재 반복 재생인지 여부를 가져오는 함수

##### 반환값

true:반복 재생 상태, false:반복 상태 아님

#### boolean com.kollus.sdk.media.MediaPlayer.isPlaying ()

현재 재생 중인지 여부를 알 수 있는 함수

##### 반환값

true:재생 상태, false: 재생 상태 아님

#### void com.kollus.sdk.media.MediaPlayer.pause () throws IllegalStateException

재생을 일시 정지 시킬 때 호출하는 함수

#### void com.kollus.sdk.media.MediaPlayer.prepareAsync ()

setDataSource 호출 후 재생을 준비시키는 함수 이 함수를 호출하면 onPrepared로 응답이 옵니다.

#### void com.kollus.sdk.media.MediaPlayer.release ()

자원을 회수하는 함수로 stop후 호출하면 됩니다.

#### void com.kollus.sdk.media.MediaPlayer.releaseWhenFinishing ()

자원을 회수하는 함수로 stop후 호출하면 됩니다.

#### void com.kollus.sdk.media.MediaPlayer.seekTo (int *msec*) throws IllegalStateException

해당 밀리초로 이동하는 함수로 I-Frame단위로 이동한다

#### void com.kollus.sdk.media.MediaPlayer.seekToExact (int *msec*) throws IllegalStateException

해당 밀리초로 이동하는 함수로 시간단위로 이동한다

#### void com.kollus.sdk.media.MediaPlayer.selectTrack (boolean *bSubTimedText*, int *index*) throws IllegalStateException

getTrackInfo로 가져온 트랙정보 중에서 해당 트랙으로 재생을 원할 때 호출하는 함수

##### 매개변수

| *index* | 트랙 인덱스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setAiPlaybackRateEnable (boolean *enable*)

AI 배속 여부 설정하는 함수

##### 매개변수

| *enable* | true : AI 배속, false : 일반 배속 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setAudioDelay (int *timeMs*)

AV 싱크를 조절하는 함수

##### 매개변수

| *timeMs* | 조절할 시간(밀리초) |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setBandwidth (int *bandwidth*)

컨텐츠가 ABR인 경우 bandwidth으로 화질 선택하는 함수

##### 매개변수

| *bandwidth* | 선택할 bandwidth |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setBandwidthName (String *bandwidthName*)

컨텐츠가 ABR인 경우 bandwidthName으로 화질 선택하는 함수

##### 매개변수

| *bandwidthName* | 선택할 bandwidth 명 |
| --- | --- |

#### boolean com.kollus.sdk.media.MediaPlayer.setBufferingRatio (int *ratio*)

기본 버퍼링량의 배수만큼 설정하는 함수로 Utils.PLAYER\_TYPE\_KOLLUS인 경우에만 사용 가능합니다.

##### 매개변수

| *ratio* | - 범위는 1~10입니다. |
| --- | --- |

##### 반환값

true:설정 성공, false:설정 실패

#### void com.kollus.sdk.media.MediaPlayer.setCaptureDetectLister (CaptureDetectLister *listener*)

CaptureDetectLister 설정하는 함수

##### 매개변수

| *listener* | CaptureDetectLister 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setDataSourceByKey (String *mediaContentKey*, String *extraDrmParam*)

다운로드 리스트에서 재생을 원할 때 호출하는 함수

##### 매개변수

| *mediaContentKey* | 컨텐트 미디어 키 |
| --- | --- |
| *extraDrmParam* | 동적 DRMCallback/PlayCallback 인자 |

#### void com.kollus.sdk.media.MediaPlayer.setDataSourceByUrl (String *path*, String *extraDrmParam*)

스트리밍 재생을 원할 때 호출하는 함수

##### 매개변수

| *path* | 스트리밍 컨텐츠 경로 |
| --- | --- |
| *extraDrmParam* | 동적 DRMCallback/PlayCallback 인자 |

#### void com.kollus.sdk.media.MediaPlayer.setDisplay (SurfaceHolder *sh*)

비디오 출력 화면을 설정하는 함수

##### 매개변수

| *sh* | 비디오 화면을 출력할 SurfaceHolder |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setEmulatorCheckerListener (EmulatorCheckerListener *listener*)

Emulator에서 재생시도 시 발생하는 리스너

##### 매개변수

| *listener* | EmulatorCheckerListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setForensicWatermarkView (ForensicWatermarkView *view*)

ForensicWatermarkView를 설정하는 함수로 SurfaceView나 Surface를 포함한 상태에서 호출하면 됩니다. 포렌식 비디오 워터마크를 지원하는 경우 필수로 설정해야 합니다.

##### 매개변수

| *view* | ForensicWatermarkView 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setInitialBandwidth (int *bandwidth*)

시작하는 Bandwidth를 설정하는 함수로 0이면 bandwidth 리스트의 첫 항목부터 시작하고 0보다 큰 값을 설정하면 설정된 값을 넘지 않는 최고의 bandwidth로 시작합니다. prepareAsync전에 호출하시면 됩니다.

##### 매개변수

| *bandwidth* | 초기 시청할 bandwidth |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setKollusPlayerBookmarkListener (KollusPlayerBookmarkListener *listener*)

KollusPlayerBookmarkListener를 설정하는 함수

##### 매개변수

| *listener* | KollusPlayerBookmarkListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setKollusPlayerLMSListener (KollusPlayerLMSListener *listener*)

KollusPlayerLMSListener를 설정하는 함수

##### 매개변수

| *listener* | KollusPlayerLMSListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setKollusStorage (KollusStorage *storage*)[protected]

StorageManager 인스턴스를 설정하는 함수 initialize와 setDataSource사이에 호출하면 됩니다.

##### 매개변수

| *storage* | KollusStorage 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setLastOffsetTime (int *lastOffsetTimeMs*)

Mpeg-Dash에서 timeline 기준으로 마지막부터 어느 위치부터 시청할 것인지 설정하는 함수

##### 매개변수

| *lastOffsetTimeMs* | 시청할 위치 (밀리초) |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setLmsOffDownloadContent (boolean *bOn*)

다운로드 컨텐츠의 LMS데이터 전송여부를 설정하는 함수입니다.

##### 매개변수

| *bOn* | true:전송, false:미전송 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setLooping (boolean *looping*)

반복 재생을 설정하는 함수

##### 매개변수

| *looping* | true:반복 재생 false:반복 재생 안함 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setMinDurationForQualityIncreaseMs (int *msec*)

상위 bandwidth로 올릴 때 남아 있는 버퍼 용량이 설정값이상인 경우만 적용되도록 설정하는 함수로 기본 값은 10초입니다. prepareAsync전에 호출하시면 됩니다.

##### 매개변수

| *msec* | 최소 버퍼링 시간(밀리초) |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setMute (boolean *mute*)

음소거를 설정하는 함수

##### 매개변수

| *mute* | true:음소거 상태, false:음소거 상태 아님 |
| --- | --- |

#### boolean com.kollus.sdk.media.MediaPlayer.setNetworkTimeout (int *timeout*)

네트워크 타임아웃 설정하는 함수로 Utils.PLAYER\_TYPE\_KOLLUS인 경우에만 사용 가능합니다.

##### 매개변수

| *timeout* | 네트워크 타임 아웃 시간(초) |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setNotifyLastReport (boolean *bOn*)

종료시 LMS데이터 관련 사항을 UI로 Notify 받을지 여부를 설정하는 함수입니다. true로 설정하면 종료가 네트워크 상황에 따라 오래걸릴 수 있습니다.

##### 매개변수

| *bOn* | true:전송, false:미전송 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setOnBufferingUpdateListener (OnBufferingUpdateListener *listener*)

OnBufferingUpdateListener를 설정하는 함수

##### 매개변수

| *listener* | OnBufferingUpdateListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setOnCencDrmListener (OnCencDrmListener *listener*)

OnCencDrmListener를 설정하는 함수

##### 매개변수

| *listener* | OnCencDrmListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setOnCompletionListener (OnCompletionListener *listener*)

OnCompletionListener를 설정하는 함수

##### 매개변수

| *listener* | OnCompletionListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setOnErrorListener (OnErrorListener *listener*)

OnErrorListener를 설정하는 함수

##### 매개변수

| *listener* | OnErrorListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setOnExternalDisplayDetectListener (OnExternalDisplayDetectListener *listener*)

OnExternalDisplayDetectListener를 설정하는 함수

##### 매개변수

| *listener* | OnExternalDisplayDetectListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setOnInfoListener (OnInfoListener *listener*)

OnInfoListener를 설정하는 함수

##### 매개변수

| *listener* | OnInfoListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setOnMediaRouteInfoListener (OnMediaRouteInfoListener *listener*)

OnMediaRouteInfoListener 설정하는 함수 \*

##### 매개변수

| *listener* | the callback that will be run |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setOnPreparedListener (OnPreparedListener *listener*)

OnPreparedListener를 설정하는 함수

##### 매개변수

| *listener* | OnPreparedListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setOnSeekCompleteListener (OnSeekCompleteListener *listener*)

OnSeekCompleteListener를 설정하는 함수 \*

##### 매개변수

| *listener* | OnSeekCompleteListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setOnTimedTextDetectListener (OnTimedTextDetectListener *listener*)

OnTimedTextDetectListener를 설정하는 함수

##### 매개변수

| *listener* | OnTimedTextDetectListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setOnTimedTextListener (OnTimedTextListener *listener*)

OnTimedTextListener를 설정하는 함수

##### 매개변수

| *listener* | OnTimedTextListener 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setOnVideoSizeChangedListener (OnVideoSizeChangedListener *listener*)

OnVideoSizeChangedListener를 설정하는 함수

##### 매개변수

| *listener* | OnVideoSizeChangedListener 인스턴스 |
| --- | --- |

#### boolean com.kollus.sdk.media.MediaPlayer.setPlayingRate (float *playing\_rate*) throws IllegalStateException

배속 조정 함수로 (0.5 ~ 10.0)범위내에서 지원합니다. 오디오 워터마크가 있는 경우 최대 1.5배속으로 제한됩니다. 2.0을 넘어가면 화면 drop으로 원활한 시청이 안될 수 있습니다.

##### 매개변수

| *playing\_rate* | 설정할 배속 |
| --- | --- |

##### 반환값

true:배속 설정 성공, false:배속 설정 실패

#### void com.kollus.sdk.media.MediaPlayer.setRemoteMediaRouteInfo (String *castNameSpace*)

크롬 캐스트 리시버와 메세지를 주고 받을 때 사용되는 네임스페이스 설정

##### 매개변수

| *castNameSpace* | Kollus Cast Receiver를 사용하면 MediaPlayer.CAST\_NAME\_SPACE 입력하면 됩니다. prepareAsync전에 호출해야 합니다. |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setScreenOnWhilePlaying (boolean *screenOn*)

재생 중 화면을 항상 On시킬 지 여부를 설정하는 함수 \*

##### 매개변수

| *screenOn* | true:항상 켬, false: 자동 꺼짐 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setSurface (Surface *surface*)

비디오 출력 화면을 설정하는 함수

##### 매개변수

| *surface* | 비디오 화면을 출력할 Surface |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setVideoRending (boolean *bRendering*)

Video Render 여부를 설정하는 함수

##### 매개변수

| *bRendering* | - true는 렌터링을 하고 false는 레터링을 하지 않습니다. |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.setVolumeLevel (int *level*)

현재 볼륨 크기를 설정하는 함수

##### 매개변수

| *level* | 볼륨값 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.skip ()

인트로 영상이 있는 경우 현재 재생하고 있는 컨텐츠를 건너뛰고 싶을 때 호출합니다.

#### void com.kollus.sdk.media.MediaPlayer.start () throws IllegalStateException

onPrepared후 재생을 시작하거나 또는 일시정지 후 재생을 원할때 호출하는 함수

#### void com.kollus.sdk.media.MediaPlayer.stop () throws IllegalStateException

재생을 정지 시킬 때 호출하는 함수

#### boolean com.kollus.sdk.media.MediaPlayer.supportPlaybackrateControl ()

현재 플레이어가 배속 조절을 지원하는 지 여부

##### 반환값

ture:배속 지원, false: 배속 미지원

#### void com.kollus.sdk.media.MediaPlayer.updateKollusBookmark (int *position*, String *label*) throws IllegalStateException

북마크 정보를 추가하는 함수 onBookmarkInfoUpdated로 성공여부가 리턴됩니다.

##### 매개변수

| *position* | 북마크 초단위 위치 |
| --- | --- |
| *label* | 북마크 이름 |

### 멤버 데이터 문서화

**final int com.kollus.sdk.media.MediaPlayer.MEDIA\_ERROR\_NOT\_VALID\_FOR\_PROGRESSIVE\_PLAYBACK = 200[static]**

MPEG4BOX 컨테이너 오류로 재생을 할 수 없는 경우

##### 참고

**com.kollus.sdk.media.MediaPlayer.OnErrorListener**

#### final int com.kollus.sdk.media.MediaPlayer.MEDIA\_ERROR\_UNKNOWN = 1[static]

알 수 없는 이유

##### 참고

**com.kollus.sdk.media.MediaPlayer.OnErrorListener**

#### final int com.kollus.sdk.media.MediaPlayer.MEDIA\_INFO\_FRAME\_RATE = 7[static]

Video Frame Rate 정보를 가져올 때

##### 참고

MediaPlayer.OnInfoListener

#### final int com.kollus.sdk.media.MediaPlayer.RENDER\_16\_MULTIPLE = 4[static]

렌더링 버퍼사이즈 결정 시 16의 배수값으로 합니다.

#### final int com.kollus.sdk.media.MediaPlayer.RENDER\_2\_MULTIPLE = 2[static]

렌더링 버퍼사이즈 결정 시 2의 배수값으로 합니다.

#### final int com.kollus.sdk.media.MediaPlayer.RENDER\_2\_POWER = 1[static]

렌더링 버퍼사이즈 결정 시 2의 승수값으로 합니다.

#### final int com.kollus.sdk.media.MediaPlayer.RENDER\_32\_MULTIPLE = 5[static]

렌더링 버퍼사이즈 결정 시 16의 배수값으로 합니다.

#### final int com.kollus.sdk.media.MediaPlayer.RENDER\_4\_MULTIPLE = 3[static]

렌더링 버퍼사이즈 결정 시 4의 배수값으로 합니다.

#### final int com.kollus.sdk.media.MediaPlayer.RENDER\_MODEL = 0[static]

렌더링 버퍼사이즈 결정 시 모델별로 구분합니다. (기본값)

## com.kollus.sdk.media.MediaPlayer.MediaRouteState Enum Reference

com.kollus.sdk.media.MediaPlayer.MediaRouteState**Public 속성**

**STARTING**

**STARTED**

**RESUME**

**END**

### 상세한 설명

크롬 캐스트 상태

## com.kollus.sdk.media.MediaPlayer.OnBufferingUpdateListener 인터페이스 참조

com.kollus.sdk.media.MediaPlayer.OnBufferingUpdateListener**Public 멤버 함수**

void **onBufferingUpdate** (**MediaPlayer** mp, int percent)

### 상세한 설명

버퍼링 상태를 응답받는 인터페이스. 현재는 사용되지 않습니다.

## com.kollus.sdk.media.MediaPlayer.OnCencDrmListener 인터페이스 참조

com.kollus.sdk.media.MediaPlayer.OnCencDrmListener**Public 멤버 함수**

void **onProxyError** (int nErrorCode, String strErrorMessage)

### 상세한 설명

Widevine등의 DRM서버와 연동관련 인터페이스

## com.kollus.sdk.media.MediaPlayer.OnCompletionListener 인터페이스 참조

com.kollus.sdk.media.MediaPlayer.OnCompletionListener**Public 멤버 함수**

void **onCompletion** (**MediaPlayer** mp)

### 상세한 설명

재생 완료 여부를 응답받는 인터페이스

### 멤버 함수 문서화

**void com.kollus.sdk.media.MediaPlayer.OnCompletionListener.onCompletion (MediaPlayer *mp*)**

재생이 완료되면 호출됩니다.

##### 매개변수

| *mp* | 재생이 완료된 플레이어 인스턴스 |
| --- | --- |

## com.kollus.sdk.media.MediaPlayer.OnErrorListener 인터페이스 참조

com.kollus.sdk.media.MediaPlayer.OnErrorListener**Public 멤버 함수**

boolean **onError** (**MediaPlayer** mp, int what, int extra)

### 상세한 설명

에러 코드를 응답받는 인터페이스

### 멤버 함수 문서화

**boolean com.kollus.sdk.media.MediaPlayer.OnErrorListener.onError (MediaPlayer *mp*, int *what*, int *extra*)**

에러가 발생하면 호출됩니다.

##### 매개변수

| *mp* | 에러가 발생한 플레이어 |
| --- | --- |
| *what* | 에러 코드 1 |
| *extra* | 에러 코드 2 |

##### 반환값

UI에서 에러를 처리했는지 여부를 SDK에 전달하는 값으로 False를 리턴하면 SDK에서 onComplete를 호출됩니다.

## com.kollus.sdk.media.MediaPlayer.OnExternalDisplayDetectListener 인터페이스 참조

com.kollus.sdk.media.MediaPlayer.OnExternalDisplayDetectListener**Public 멤버 함수**

void **onExternalDisplayDetect** (**EXTERNAL\_DISPLAY\_MODE** type, boolean plugged)

### 상세한 설명

외부 모니터를 감지하는 인터페이스

### 멤버 함수 문서화

**void com.kollus.sdk.media.MediaPlayer.OnExternalDisplayDetectListener.onExternalDisplayDetect (EXTERNAL\_DISPLAY\_MODE *type*, boolean *plugged*)**

외부 모니터가 감지되었을 때 호출됩니다.

##### 매개변수

| *type* | 오부 모니터의 종류. [EXTERNAL\_HDMI\_DISPLAY, EXTERNAL\_WIFI\_DISPLAY] |
| --- | --- |
| *plugged* | 외부 모니터가 연결되면 true, 끊기면 false가 리턴됩니다. |

## com.kollus.sdk.media.MediaPlayer.OnInfoListener 인터페이스 참조

com.kollus.sdk.media.MediaPlayer.OnInfoListener**Public 멤버 함수**

boolean **onInfo** (**MediaPlayer** mp, int what, int extra)

void **onBufferingStart** (**MediaPlayer** mp)

void **onBufferingEnd** (**MediaPlayer** mp)

void **onFrameDrop** (**MediaPlayer** mp)

void **onDownloadRate** (**MediaPlayer** mp, int downloadRate)

void **onDetectBandwidthList** (**MediaPlayer** mp, List< **BandwidthItem** > list)

void **onChangedBandwidth** (**MediaPlayer** mp, **BandwidthItem** item)

void **onCodecInitFail** (**MediaPlayer** mp, String componentName)

### 상세한 설명

info/warning을 응답받는 인터페이스

### 멤버 함수 문서화

**void com.kollus.sdk.media.MediaPlayer.OnInfoListener.onBufferingEnd (MediaPlayer *mp*)**

버퍼링이 끝나면 호출됩니다.

##### 매개변수

| *mp* | 버퍼링이 끝난 플레이어 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.OnInfoListener.onBufferingStart (MediaPlayer *mp*)

버퍼링이 시작되면 호출됩니다.

##### 매개변수

| *mp* | 버퍼링이 시작된 플레이어 인스턴스 |
| --- | --- |

#### void com.kollus.sdk.media.MediaPlayer.OnInfoListener.onChangedBandwidth (MediaPlayer *mp*, BandwidthItem *item*)

ABR 컨텐츠인 경우 컨텐츠 bitrate이 바뀌면 호출됩니다.

##### 매개변수

| *mp* | 플레이어 인스턴스 |
| --- | --- |
| *item* | 바뀐 bitrate 정보 |

#### void com.kollus.sdk.media.MediaPlayer.OnInfoListener.onCodecInitFail (MediaPlayer *mp*, String *componentName*)

하드웨어 코덱 초기화 실패시 호출됩니다. 오디오 워터 마크가 있거나 젤리빈 MR2(18)미만인 경우만 호출됩니다.

##### 매개변수

| *mp* | 플레이어 인스턴스 |
| --- | --- |
| *componentName* | 실패가 난 코덱 이름 |

#### void com.kollus.sdk.media.MediaPlayer.OnInfoListener.onDetectBandwidthList (MediaPlayer *mp*, List< BandwidthItem > *list*)

ABR 컨텐츠인 경우 호출됩니다.

##### 매개변수

| *mp* | 플레이어 인스턴스 |
| --- | --- |
| *list* | ABR정보를 가지고 있는 리스트 |

#### void com.kollus.sdk.media.MediaPlayer.OnInfoListener.onDownloadRate (MediaPlayer *mp*, int *downloadRate*)

컨텐츠 파일이 다운로드가 완료되면 해당 파일의 다운로드 bps를 알려줍니다.

##### 매개변수

| *mp* | 플레이어 인스턴스 |
| --- | --- |
| *downloadRate* | 해당 파일의 다운로드 bps |

#### void com.kollus.sdk.media.MediaPlayer.OnInfoListener.onFrameDrop (MediaPlayer *mp*)

플레임 드랍이 발생하면 호출됩니다. 오디오 워터 마크가 있거나 젤리빈 MR2(18)미만인 경우만 호출됩니다.

##### 매개변수

| *mp* | 플레이어 인스턴스 |
| --- | --- |

#### boolean com.kollus.sdk.media.MediaPlayer.OnInfoListener.onInfo (MediaPlayer *mp*, int *what*, int *extra*)

플레이어의 상태 정보가 있을 때 호출됩니다.

##### 매개변수

| *mp* | 플레이어 인스턴스 |
| --- | --- |
| *what* | 정보의 종류 |
| *extra* | 기타 정보 |

## com.kollus.sdk.media.KollusStorage.OnKollusStorageDeviceListener 인터페이스 참조

com.kollus.sdk.media.KollusStorage.OnKollusStorageDeviceListener**Public 멤버 함수**

void **onDeviceSettingStart** ()

void **onDeviceSettingEnd** ()

void **onDeviceSettingError** (int nErrorCode)

### 상세한 설명

setDeviceASync의 인자로 디바이스 셋팅의 과정을 응답받는 인터페이스

### 멤버 함수 문서화

**void com.kollus.sdk.media.KollusStorage.OnKollusStorageDeviceListener.onDeviceSettingEnd ()**

디바이스 셋팅 완료

#### void com.kollus.sdk.media.KollusStorage.OnKollusStorageDeviceListener.onDeviceSettingError (int *nErrorCode*)

디바이스 셋팅 오류 시. 예를들면 주어진 경로로 필수 디렉토리를 생성할 수 없을 때 발생

#### void com.kollus.sdk.media.KollusStorage.OnKollusStorageDeviceListener.onDeviceSettingStart ()

디바이스 셋팅 시작

## com.kollus.sdk.media.KollusStorage.OnKollusStorageListener 인터페이스 참조

com.kollus.sdk.media.KollusStorage.OnKollusStorageListener**Public 멤버 함수**

void **onComplete** (**KollusContent** content)

void **onProgress** (**KollusContent** content)

void **onError** (**KollusContent** content, int errorCode)

### 상세한 설명

스토리지 매니저 상황(에러, 다운로드 중, 다운로드 완료)을 응답받는 인터페이스

### 멤버 함수 문서화

**void com.kollus.sdk.media.KollusStorage.OnKollusStorageListener.onComplete (KollusContent *content*)**

다운로드 완료

##### 매개변수

| *content* | 컨텐츠 정보 |
| --- | --- |

#### void com.kollus.sdk.media.KollusStorage.OnKollusStorageListener.onError (KollusContent *content*, int *errorCode*)

다운로드 에러

##### 매개변수

| *content* | 컨텐츠 정보 |
| --- | --- |
| *errorCode* | 에러 코드 |

#### void com.kollus.sdk.media.KollusStorage.OnKollusStorageListener.onProgress (KollusContent *content*)

다운로드 중

##### 매개변수

| *content* | 컨텐츠 정보 |
| --- | --- |

## com.kollus.sdk.media.MediaPlayer.OnMediaRouteInfoListener 인터페이스 참조

com.kollus.sdk.media.MediaPlayer.OnMediaRouteInfoListener**Public 멤버 함수**

void **onAvailable** (**MediaPlayer** mp, boolean bAvailable)

void **onChangedState** (**MediaPlayer** mp, MediaPlayer.MediaRouteState state)

void **onError** (**MediaPlayer** mp)

### 상세한 설명

ChromeCast 상태를 응답받는 인터페이스

## com.kollus.sdk.media.MediaPlayer.OnPreparedListener 인터페이스 참조

com.kollus.sdk.media.MediaPlayer.OnPreparedListener**Public 멤버 함수**

void **onPrepared** (**MediaPlayer** mp)

### 상세한 설명

멀티뷰 지원을 위해 스트림을 추가하는 함수

##### 매개변수

| *path* | 서브 스트림 |
| --- | --- |
| *videoLayer* | 서브 스트림을 출력할 화면 |

##### 반환값

서브 스트림이 추가된 인덱스 0보다 작으면 에러 멀티뷰에서 메인과 sub 플레이어의 출력 화면을 바꾸는 함수

##### 매개변수

| *role* | 메인 화면으로 바꿀 플레이어의 role |
| --- | --- |

##### 반환값

성공 : Sub 화면으로 이동된 플레이어 role, 실패 : IndexOutOfBoundsException 발생 재생 준비가 완료를 응답받는 인터페이스

### 멤버 함수 문서화

**void com.kollus.sdk.media.MediaPlayer.OnPreparedListener.onPrepared (MediaPlayer *mp*)**

재생 준비가 되면 호출됩니다.

##### 매개변수

| *mp* | 재생 준비가 된 플레이어 |
| --- | --- |

## com.kollus.sdk.media.MediaPlayer.OnSeekCompleteListener 인터페이스 참조

com.kollus.sdk.media.MediaPlayer.OnSeekCompleteListener**Public 멤버 함수**

void **onSeekComplete** (**MediaPlayer** mp)

### 상세한 설명

seek완료 여부를 응답받는 인터페이스

### 멤버 함수 문서화

**void com.kollus.sdk.media.MediaPlayer.OnSeekCompleteListener.onSeekComplete (MediaPlayer *mp*)**

seek가 완료되면 호출됩니다.

##### 매개변수

| *mp* | seek가 완료된 플레이어 |
| --- | --- |

## com.kollus.sdk.media.MediaPlayer.OnTimedTextDetectListener 인터페이스 참조

com.kollus.sdk.media.MediaPlayer.OnTimedTextDetectListener**Public 멤버 함수**

void **onTimedTextDetect** (**MediaPlayer** mp, int trackIndex)

void **onSubTimedTextDetect** (**MediaPlayer** mp, int trackIndex)

### 상세한 설명

addTimedTextSource 또는 자막이 있는 경우 응답받는 인터페이스

### 멤버 함수 문서화

**void com.kollus.sdk.media.MediaPlayer.OnTimedTextDetectListener.onTimedTextDetect (MediaPlayer *mp*, int *trackIndex*)**

자막이 있는 경우 호출됩니다.

##### 매개변수

| *mp* | 해당 콜백과 연관된 플레이어 |
| --- | --- |
| *trackIndex* | 자막의 인덱스 |

## com.kollus.sdk.media.MediaPlayer.OnTimedTextListener 인터페이스 참조

com.kollus.sdk.media.MediaPlayer.OnTimedTextListener**Public 멤버 함수**

void **onTimedText** (**MediaPlayer** mp, String text)

void **onSubTimedText** (**MediaPlayer** mp, String text)

void **onTimedImage** (**MediaPlayer** mp, byte[] image, int width, int height)

### 상세한 설명

해당 시간의 자막을 응답받는 인터페이스

### 멤버 함수 문서화

**void com.kollus.sdk.media.MediaPlayer.OnTimedTextListener.onSubTimedText (MediaPlayer *mp*, String *text*)**

텍스트 기반 부가 자막인 경우

##### 매개변수

| *mp* | 해당 콜백과 연관된 플레이어 |
| --- | --- |
| *text* | 현재 시간의 자막 |

#### void com.kollus.sdk.media.MediaPlayer.OnTimedTextListener.onTimedImage (MediaPlayer *mp*, byte[] *image*, int *width*, int *height*)

이미지 기반 자막인 경우 호출됩니다. VobSub만 지원합니다.

##### 매개변수

| *mp* | 해당 콜백과 연관된 플레이어 |
| --- | --- |
| *image* | 자막 이미지의 비트맵 데이터 |
| *width* | 자막 이미지의 가로 사이즈 |
| *height* | 자막 이미지의 세로 사이즈 |

#### void com.kollus.sdk.media.MediaPlayer.OnTimedTextListener.onTimedText (MediaPlayer *mp*, String *text*)

텍스트 기반 자막인 경우

##### 매개변수

| *mp* | 해당 콜백과 연관된 플레이어 |
| --- | --- |
| *text* | 현재 시간의 자막 |

## com.kollus.sdk.media.MediaPlayer.OnVideoSizeChangedListener 인터페이스 참조

com.kollus.sdk.media.MediaPlayer.OnVideoSizeChangedListener**Public 멤버 함수**

void **onVideoSizeChanged** (**MediaPlayer** mp, int width, int height)

### 상세한 설명

컨텐츠 resolution정보가 변경되었을 때 응답받는 인터페이스 최초 resolution정보를 받을 때도 사용됩니다.

### 멤버 함수 문서화

**void com.kollus.sdk.media.MediaPlayer.OnVideoSizeChangedListener.onVideoSizeChanged (MediaPlayer *mp*, int *width*, int *height*)**

비디오 사이즈가 확인이 되면 호출됩니다.

##### 매개변수

| *mp* | 해당 콜백과 연관된 플레이어 |
| --- | --- |
| *width* | 비디오의 가로 정보 |
| *height* | 비디오의 세로 정보 |

## com.kollus.sdk.media.MediaPlayer.RENDER\_MODE Enum Reference

com.kollus.sdk.media.MediaPlayer.RENDER\_MODE**Public 속성**

**RENDER\_MODELRENDER\_2\_POWER**

**RENDER\_2\_MULTIPLE**

**RENDER\_4\_MULTIPLE**

**RENDER\_16\_MULTIPLE**

**RENDER\_32\_MULTIPLE**

### 상세한 설명

렌더링 모드

### 멤버 데이터 문서화

**com.kollus.sdk.media.MediaPlayer.RENDER\_MODE.RENDER\_16\_MULTIPLE**

렌더링 방식을 16배수로 하는 경우

#### com.kollus.sdk.media.MediaPlayer.RENDER\_MODE.RENDER\_2\_MULTIPLE

렌더링 방식을 2배수로 하는 경우

#### com.kollus.sdk.media.MediaPlayer.RENDER\_MODE.RENDER\_2\_POWER

렌더링 방식을 2승수로 하는 경우

#### com.kollus.sdk.media.MediaPlayer.RENDER\_MODE.RENDER\_32\_MULTIPLE

렌더링 방식을 32배수로 하는 경우

#### com.kollus.sdk.media.MediaPlayer.RENDER\_MODE.RENDER\_4\_MULTIPLE

렌더링 방식을 4배수로 하는 경우

#### com.kollus.sdk.media.MediaPlayer.RENDER\_MODE.RENDER\_MODEL

렌더링 방식을 이미 알고 있는 경우

## com.kollus.sdk.media.content.KollusContent.SubtitleInfo 클래스 참조

com.kollus.sdk.media.content.KollusContent.SubtitleInfo**Public 속성**

String **name**

String **url**

String **languageCode**

### 상세한 설명

자막 정보 인터페이스

### 멤버 데이터 문서화

**String com.kollus.sdk.media.content.KollusContent.SubtitleInfo.languageCode**

자막 언어 코드

#### String com.kollus.sdk.media.content.KollusContent.SubtitleInfo.name

자막 언어명

#### String com.kollus.sdk.media.content.KollusContent.SubtitleInfo.url

자막 경로

## com.kollus.sdk.media.util.Utils 클래스 참조

com.kollus.sdk.media.util.Utils**클래스**

enum **DecoderType**enum **DEVICE\_TYPE**

### 정적 Public 멤버 함수

static< T > T **checkNotNull** (T object)

static void **closeSilently** (Closeable c)

static void **closeSilently** (ParcelFileDescriptor fd)

static void **closeSilently** (Cursor cursor)

static long **getTotalMemorySize** (String path)

static long **getAvailableMemorySize** (String path)

static String **stringForTime** (String dayString, String hourString, String minString, String secString, int timeMs)

static String **stringForTime** (int timeMs)

static String **stringForTimeHHMMSS** (int timeMs)

static String **stringForTimeHMMSS** (int timeMs)

static String **stringForTimeMMSS** (int timeMs)

static String **getFileMD5** (File updateFile)

static String **makeMD5** (String param)

static String **makeSHA1** (String param)

static void **setPlayerId** (String id)

static String **getPlayerId** (Context context)

static String **getPlayerIdMd5** (Context context)

static String **getPlayerIdSha1** (Context context)

static String **getString** (JSONObject json, String key)

static void **setStoragePath** (Context context, String path)

static File[] **getExternalFilesDirs** (@NonNull Context context, @Nullable String type)

static String **getStoragePath** (Context context)

static boolean **isRooting** ()

static **DEVICE\_TYPE** **getDeviceType** (Context context)

static Object **invoke** (Object obj, String methodName, Object[] objList)

static int **getPlayerType** ()

static String **getDirectoryJSON** (Context context, String prefKey)

static String **getDirectoryJSONByPath** (Context context, String path, String prefKey)

static void **saveDirectoryJSON** (Context context, String dbPath, String jsonString)

static void **setDecoderType** (Context context, **DecoderType** type)

static **DecoderType** **getDecoderType** (Context context)

static void **setRenderType** (Context context, @MediaPlayerBase.RENDER\_MODE int mode)

static int **getRenderType** (Context context)

static boolean **makeDirectory** (String path)

static void **deleteDirectory** (String path)

static long **getFolderSize** (File directory)

### 정적 Public 속성

static final int **PLAYER\_TYPE\_NONE** = 0

static final int **PLAYER\_TYPE\_NATIVE** = 1

static final int **PLAYER\_TYPE\_KOLLUS** = 2

static final int **PLAYER\_TYPE\_EXO** = 3

static final boolean **USE\_EXO\_PLAYER** = false

### 멤버 함수 문서화

**static void com.kollus.sdk.media.util.Utils.deleteDirectory (String *path*)[static]**

주어진 경로의 폴더를 삭제하는 함수

##### 매개변수

| *path* | 경로 |
| --- | --- |

#### static long com.kollus.sdk.media.util.Utils.getAvailableMemorySize (String *path*)[static]

주어진 경로의 SD카드의 남아있는 용량을 가져오는 함수

##### 매개변수

| *path* | 남아있는 용량을 가져올 경로 |
| --- | --- |

##### 반환값

남아있는 용량

#### static DecoderType com.kollus.sdk.media.util.Utils.getDecoderType (Context *context*)[static]

코덱 타입을 가져오는 함수

##### 매개변수

| *context* | App Context |
| --- | --- |

##### 반환값

코덱 타입

##### 참고

DecoderType

#### static DEVICE\_TYPE com.kollus.sdk.media.util.Utils.getDeviceType (Context *context*)[static]

디바이스 타입을 확인하는 함수

##### 매개변수

| *context* | App Context |
| --- | --- |

##### 참고

DEVICE\_TYPE

##### 반환값

디바이스 타입

#### static String com.kollus.sdk.media.util.Utils.getDirectoryJSON (Context *context*, String *prefKey*)[static]

디렉토리 구조 JSON 데이터를 불러오는 함수

##### 매개변수

| *context* | App Context |
| --- | --- |
| *prefKey* | 저장된 JSON 데이터의 키 |

##### 반환값

JSON 데이터

#### static String com.kollus.sdk.media.util.Utils.getDirectoryJSONByPath (Context *context*, String *path*, String *prefKey*)[static]

디렉토리 구조 JSON 데이터를 주어진 경로에서 불러오는 함수

##### 매개변수

| *context* | App Context |
| --- | --- |
| *path* | 불러올 경로 |
| *prefKey* | 저장된 JSON 데이터의 키 |

##### 반환값

JSON 데이터

#### static String com.kollus.sdk.media.util.Utils.getFileMD5 (File *updateFile*)[static]

주어진 File의 MD5 체크섬을 가져오는 함수

##### 매개변수

| *updateFile* | 체크섬을 가져올 파일 |
| --- | --- |

##### 반환값

체크섬

#### static long com.kollus.sdk.media.util.Utils.getFolderSize (File *directory*)[static]

주어진 폴더의 사이즈를 가져오는 함수

##### 매개변수

| *directory* | 사이즈를 가져올 폴더 |
| --- | --- |

##### 반환값

사용하는 용량량

#### static String com.kollus.sdk.media.util.Utils.getPlayerId (Context *context*)[static]

플레이어 ID를 가져오는 함수

##### 매개변수

| *context* | App Context |
| --- | --- |

##### 반환값

플레이어 ID

#### static String com.kollus.sdk.media.util.Utils.getPlayerIdMd5 (Context *context*)[static]

내부적으로 사용하는 플레이어 ID를 생성하는 함수 16bytes 헥사스트링임(32자).

##### 매개변수

| *context* | App Context |
| --- | --- |

##### 반환값

플레이어 ID 체크섬

#### static String com.kollus.sdk.media.util.Utils.getPlayerIdSha1 (Context *context*)[static]

대외적으로 사용하는 플레이어 ID를 생성하는 함수 길이 제한이 없음.

##### 매개변수

| *context* | App Context |
| --- | --- |

##### 반환값

플레이어 ID 체크섬

#### static int com.kollus.sdk.media.util.Utils.getPlayerType ()[static]

생성될 플레어 타입을 가져오는 함수

##### 반환값

플레이어 타입 (PLAYER\_TYPE\_NATIVE, PLAYER\_TYPE\_KOLLUS)

#### static int com.kollus.sdk.media.util.Utils.getRenderType (Context *context*)[static]

렌더링 방식을 가져오는 함수

##### 매개변수

| *context* | App Context |
| --- | --- |

##### 반환값

렌더링 방식 @show @MediaPlayer.RENDER\_MODE

#### static String com.kollus.sdk.media.util.Utils.getStoragePath (Context *context*)[static]

플레이어가 사용하는 스토리지 경로를 가져오는 함수

##### 매개변수

| *context* | App Context |
| --- | --- |

##### 반환값

저장된 경로

#### static String com.kollus.sdk.media.util.Utils.getString (JSONObject *json*, String *key*)[static]

JSON데이터에서 주어진 key의 String값을 가져오는 함수

##### 매개변수

| *json* | JSON데이터 |
| --- | --- |
| *key* | 가져올 key |

##### 반환값

주어진 key의 String값

#### static long com.kollus.sdk.media.util.Utils.getTotalMemorySize (String *path*)[static]

주어진 경로의 SD카드의 전체 용량을 가져오는 함수

##### 매개변수

| *path* | 전체 용량을 가져올 경로 |
| --- | --- |

##### 반환값

전체 용량

#### static Object com.kollus.sdk.media.util.Utils.invoke (Object *obj*, String *methodName*, Object[] *objList*)[static]

임의의 클래스의 메서드를 invoke하는 함수

##### 매개변수

| *obj* | invoke를 원하는 클래스 |
| --- | --- |
| *methodName* | invoke할 클래스의 함수 |
| *objList* | Object 리스트 |

##### 반환값

invoke된 함수

#### static boolean com.kollus.sdk.media.util.Utils.isRooting ()[static]

루팅 여부를 확인하는 함수

##### 반환값

루팅 여부

#### static boolean com.kollus.sdk.media.util.Utils.makeDirectory (String *path*)[static]

주어진 경로로 폴더를 만드는 함수

##### 매개변수

| *path* | 경로 |
| --- | --- |

##### 반환값

폴더 생성 여부

#### static String com.kollus.sdk.media.util.Utils.makeMD5 (String *param*)[static]

주어진 문자의 MD5 체크섬을 가져오는 함수

##### 매개변수

| *param* | 체크섬을 가져올 문자 |
| --- | --- |

##### 반환값

체크섬

#### static String com.kollus.sdk.media.util.Utils.makeSHA1 (String *param*)[static]

주어진 문자의 SHA1 체크섬을 가져오는 함수

##### 매개변수

| *param* | 체크섬을 가져올 문자 |
| --- | --- |

##### 반환값

체크섬

#### static void com.kollus.sdk.media.util.Utils.saveDirectoryJSON (Context *context*, String *dbPath*, String *jsonString*)[static]

디렉토리 구조 JSON 데이터를 저장하는 함수

##### 매개변수

| *context* | App Context |
| --- | --- |
| *dbPath* | 저장할 위치 |
| *jsonString* | 저장할 JSON 데이터 |

#### static void com.kollus.sdk.media.util.Utils.setDecoderType (Context *context*, DecoderType *type*)[static]

코덱 타입을 설정하는 함수

##### 매개변수

| *context* | App Context |
| --- | --- |
| *type* | 코덱 타입 |

##### 참고

DecoderType

#### static void com.kollus.sdk.media.util.Utils.setPlayerId (String *id*)[static]

외부에서 플레이어 ID를 설정하는 함수

##### 매개변수

| *id* | 플레이어 ID |
| --- | --- |

#### static void com.kollus.sdk.media.util.Utils.setRenderType (Context *context*, @MediaPlayerBase.RENDER\_MODE int *mode*)[static]

렌더링 방식을 설정하는 함수

##### 매개변수

| *context* | App Context |
| --- | --- |
| *mode* | 렌더링 방식 |

##### 참고

@MediaPlayer.RENDER\_MODE

#### static void com.kollus.sdk.media.util.Utils.setStoragePath (Context *context*, String *path*)[static]

플레이어가 사용할 스토리지 경로를 지정하는 함수

##### 매개변수

| *context* | App Context |
| --- | --- |
| *path* | 지정할 경로 |

#### static String com.kollus.sdk.media.util.Utils.stringForTime (int *timeMs*)[static]

주어진 밀리초를 String형식으로 가져오는 함수 예 : 1:10:59, 10:59

##### 매개변수

| *timeMs* | 시간(밀리초) |
| --- | --- |

##### 반환값

정형화된 시간

#### static String com.kollus.sdk.media.util.Utils.stringForTime (String *dayString*, String *hourString*, String *minString*, String *secString*, int *timeMs*)[static]

주어진 시간을 주어진 구분자로 문자화하는 함수 예 : 1일 1시간 1분 1초

##### 매개변수

| *dayString* | 날짜 구분자 |
| --- | --- |
| *hourString* | 시간 구분자 |
| *minString* | 분 구분자 |
| *secString* | 초 구분자 |
| *timeMs* | 시간(밀리초) |

##### 반환값

정형화된 시간

#### static String com.kollus.sdk.media.util.Utils.stringForTimeHHMMSS (int *timeMs*)[static]

주어진 밀리초를 String형식으로 가져오는 함수 예 : 01:10:59

##### 매개변수

| *timeMs* | 시간(밀리초) |
| --- | --- |

##### 반환값

정형화된 시간

#### static String com.kollus.sdk.media.util.Utils.stringForTimeHMMSS (int *timeMs*)[static]

주어진 밀리초를 String형식으로 가져오는 함수 예 : 1:10:59

##### 매개변수

| *timeMs* | 시간(밀리초) |
| --- | --- |

##### 반환값

정형화된 시간

#### static String com.kollus.sdk.media.util.Utils.stringForTimeMMSS (int *timeMs*)[static]

주어진 밀리초를 String형식으로 가져오는 함수 예 : 10:59

##### 매개변수

| *timeMs* | 시간(밀리초) |
| --- | --- |

##### 반환값

정형화된 시간

### 멤버 데이터 문서화

**final int com.kollus.sdk.media.util.Utils.PLAYER\_TYPE\_EXO = 3[static]**

생성될 플레이어가 ExoPlayer인 경우 젤리빈부터 지원 (android 4.1, API Level 16)

#### final int com.kollus.sdk.media.util.Utils.PLAYER\_TYPE\_KOLLUS = 2[static]

생성될 플레이어가 Kollus 전용 플레이어인 경우

#### final int com.kollus.sdk.media.util.Utils.PLAYER\_TYPE\_NATIVE = 1[static]

생성될 플레이어가 내장 플레이어인 경우

#### final int com.kollus.sdk.media.util.Utils.PLAYER\_TYPE\_NONE = 0[static]

생성된 플레이어가 없는 경우

# 색인

색인INDEX