Bencoder2
<<Java Class>>
Ⓖ Bencoder2
edu.rutgers.cs.cs352.bt.util

- INVALID: int
- DICTIONARY: int
- INTEGER: int
- STRING: int
- LIST: int

- Bencoder2()
- getInfoBytes(byte[]):ByteBuffer
- decode(byte[]):Object
- decode(byte[],int):Object[]
- decodeInteger(byte[],int):Object[]
- decodeString(byte[],int):Object[]
- decodeList(byte[],int):Object[]
- decodeDictionary(byte[],int):Object[]
- nextObject(byte[],int):int
- encode(Object):byte[]
- encodeString(ByteBuffer):byte[]
- encodeInteger(Integer):byte[]
- encodeList(ArrayList):byte[]
- encodeDictionary(HashMap<ByteBuffer,Object>):byte[]

ToolKit
<<Java Class>>
Ⓖ ToolKit
edu.rutgers.cs.cs352.bt.util

- ToolKit()
- print(Object):void
- print(Object,int):void
- printString(byte[],boolean,int):void
- printString(ByteBuffer,boolean,int):void
- printInteger(Integer,int):void
- printList(AbstractList,int):void
- printMap(Map,int):void

RUBTClient
<<Java Class>>
Ⓖ RUBTClient
(default package)

- RUBTClient()
- main(String[]):void

Torrent
<<Java Class>>
Ⓖ Torrent
(default package)

- encodedInfoHash: String
- dataFile: RandomAccessFile
- fileByteBuffer: MappedByteBuffer
- infoMap: HashMap<String,Object>
- peerId: String
- runLock: Object
- running: boolean
- port: int
- uploaded: int
- downloaded: int
- left: int
- fileName: String
- fileLock: Object

- Torrent(TorrentInfo,String)
- stop():void
- run():void
- peerDying(Peer):void
- putPiece(Piece):void
- generatePieces():ArrayList<Piece>
- generateId():String
- getPeers():ArrayList<HashMap<String,Object>>
- sendCompleteEvent():void
- encodeInfoHash(byte[]):String

BencodeWrapper
<<Java Class>>
Ⓖ BencodeWrapper
(default package)

- BencodeWrapper()
- decode(byte[]):Object
- fixStupidByteBuffers(ByteBuffer):String
- fixStupidHashMaps(HashMap<ByteBuffer,Object>):HashMap<String,Object>
- fixStupidArrayLists(ArrayList<Object>):ArrayList<Object>

BencodingException
<<Java Class>>
Ⓖ BencodingException
edu.rutgers.cs.cs352.bt.exceptions

- serialVersionUID: long
- message: String

- BencodingException()
- BencodingException(String)
- toString():String

Peer
<<Java Class>>
Ⓖ Peer
(default package)

- PROTOCOL_HEADER: String
- KEEP_ALIVE: byte[]
- CHOKE: byte[]
- UNCHOKE: byte[]
- INTERESTED: byte[]
- NOT_INTERESTED: byte[]
- HAVE: byte[]
- REQUEST: byte[]
- PIECE: byte[]
- peerInfo: HashMap<String,Object>
- infoHash: ByteBuffer
- peerId: ByteBuffer
- handshake: ByteBuffer
- availablePieces: BitSet
- running: boolean
- die: boolean
- messages: ConcurrentLinkedQueue<ByteBuffer>
- sock: Socket

- Peer(HashMap<String,Object>,Torrent,ByteBuffer,ByteBuffer)
- run():void
- stop():void
- handleMessage(ByteBuffer):void
- recvMessage(ByteBuffer):void
- createHandshake():ByteBuffer
- canGetPiece(int):boolean
- getPiece(Piece):void
- getRequestMessage(int,int,int):ByteBuffer
- getHaveMessage(int):ByteBuffer

TorrentInfo
<<Java Class>>
Ⓖ TorrentInfo
edu.rutgers.cs.cs352.bt

- KEY_INFO: ByteBuffer
- KEY_LENGTH: ByteBuffer
- KEY_PIECES: ByteBuffer
- KEY_NAME: ByteBuffer
- KEY_PIECE_LENGTH: ByteBuffer
- KEY_ANNOUNCE: ByteBuffer
- torrent_file_bytes: byte[]
- torrent_file_map: Map<ByteBuffer,Object>
- info_map: Map<ByteBuffer,Object>
- info_hash: ByteBuffer
- announce_url: URL
- piece_length: int
- file_name: String
- file_length: int
- piece_hashes: ByteBuffer[]

- TorrentInfo(byte[])

PeerSocketRunnable
<<Java Class>>
Ⓖ PeerSocketRunnable
(default package)

- sock: Socket
- ip: String
- port: int
- messages: ConcurrentLinkedQueue<ByteBuffer>

- PeerSocketRunnable(Peer,String,int)
- run():void
- sendMessage(ByteBuffer):void
- shutdown():void

PeerState
<<Java Enumeration>>
Ⓔ PeerState
(default package)

- BEGIN: PeerState
- HANDSHAKE: PeerState
- CHOKED: PeerState
- UNCHOKED: PeerState
- DOWNLOADING: PeerState

- PeerState()

Piece
<<Java Class>>
Ⓖ Piece
(default package)

- index: int
- size: int
- hash: String
- slices: BitSet
- data: byte[]

- Piece(int,int,ByteBuffer,Torrent)
- getIndex():int
- getSize():int
- getHash():String
- getOwner():Torrent
- getByteBuffer():ByteBuffer
- putSlice(int):void
- getNextSlice():int

-owner 0..1
-owner 0..1
-busyPeers 0
-freePeers 0..*
-torrentInfo 0..1
-socketRunner 0..1   -p 0..1
-state 0..1
-busyPieces 0..*
-currentPiece 0..1