

AddressWhitelist

AddressWhitelist()
void add(AddressRange range)
void remove(AddressRange range)
void clear()
boolean isWhitelisted(InetAddress address)

AddressRange

static *AddressRange* parseCidrNotation(
 String cidrNotation
)
boolean contains(*InetAddress* address)
boolean equals(*Object* obj)
int hashCode()

Peer

<< implements *AutoCloseable* >>

static final *int* DEFAULT_PORT = 7777
Peer(*int* localPort, *Path* exportedDirectoryPath)
 throws *IOException*
State getState()
int getLocalPort()
Path getExportedDirectoryPath()
AddressWhitelist getPeerWhitelist()
void open()
void close()
void runJobs(
 List< *Job* > jobs,
 Consumer< *List*< *JobState* > > onJobStatesUpdated
)
)

JobState

JobState(*Job* job)
Job getJob()
Phase getPhase()
boolean hasFinished()
int getProgressPercentage()
long getImmediateThroughput()
long getOverallThroughput()
String getErrorMessage()
void start(*long* totalBytes)
void addToTransferredBytes(
 long bytes
)
void succeed()
void fail(
 Endpoint peerEndpoint,
 String errorMessage
)
JobState copy()

Job

Job(
 JobType type,
 List< *Endpoint* > peerEndpoints,
 Path localFilePath,
 Path remoteFilePath
)
JobType getType()
List< *Endpoint* > getPeerEndpoints()
Path getLocalFilePath()
Path getRemoteFilePath()

JobType

<< enum >>
GET
PUT

ExportedDirectory

ExportedDirectory(*Path* directoryPath) throws *IOException*
Path getDirectoryPath()
Path resolveFilePath(*Path* path, *boolean* mustExist) throws *IOException*
RandomAccessFile openFileForReading(*Path* path) throws *IOException*
TemporaryRandomAccessFile openFileForWriting(*Path* path) throws *IOException*