+po
+po
+se
+po
+

ServerConnection

olSize: int = 20
rtNumber: int = 16000
rverSocket: ServerSocket
ol: ExecutorService

Clier

-userNotExist:
-Success: Strin
-Finish: String
-fileNotExist:
-fileExist: Str
-portNumber: in
-address: Strin
-client: Client
-scan: Scanner
-socket: Socket
-retrieve: Stri

+ClientConnecti
+getUserInfo():
+run(): void
+main(args:Stri

+Se
+ru
+ma

ntConnection

String = "UserNotExist"
g = "Success"
 = "finished"
String = "File not exist"
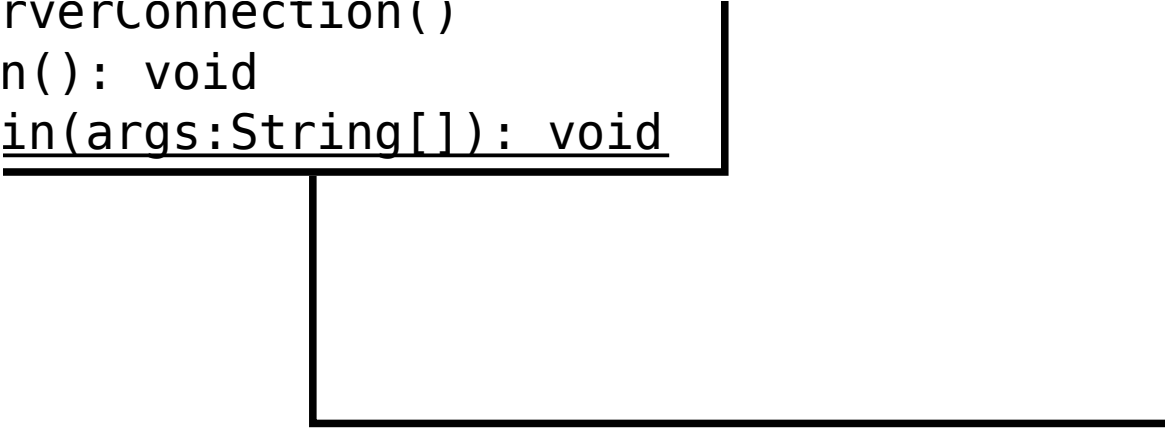ing = "FileExist"
t = 16000
g = "127.0.0.1"

ng

on(socket:Socket)
 void

ng[]): void

Client

```
rverConnection()
n(): void
in(args:String[]): void
```

Clie

```
-userNotExist: St
-Success: String
-Finish: String =
-fileNotExist: St
-fileExist: Strin
-socket: Socket
-server: Server
-retrieve: String
```

```
+ClientHandler(so
+run(): void
```

Runnable

ntHandler

```
ring = "UserNotExist"
= "Success"
= "finished"
ring = "File not exist"
g = "FileExist"
```

```
ocket:Socket)
```

-userID: Stri
-password: St
-key: byte[]
-transmit: Tr
+Client(socke
+getUserID():
+setUserID(us
+getPassword(
+setPassword(
+getKey(): by
+sendKey(): v
+readFile(fna
+sendEncryptM
+readEncryptM
+generateKey(

Client

ing
:ring
~ansmit
et:Socket)
: String
;erID:String): void
): String
(password:String): void
/te[]
/oid
ame:String): void
Message(s:String): void
Message(): String
): byte[]

Tra

+out: DataOutput!
+in: DataInputSt
+encrypt: Encrypt
+Transmit(socket
+setKey(b:byte[]
+sendEncryptMessa
+sendKey(b:byte[

S
-socket: Socket
-check: ServerCheck
-transmit: Transmit
+Server(socket:Socket)
+checkAuthenticate(userID:
+checkFile(fname:String):
+readKey(): void
+sendEncryptMessage(s:Stri
+readEncryptMessage(): Str
+sendFile(fname:String): v

nsmit
Stream
ream
t
:Socket)
): void
age(s:String): void
l): void

+readShadowFile(u:
+checkFile(filena

Server

String,password:String): boolean
boolean

ng): void
ing
oid

ServerCheck

serID:String,password:String): boolean
me:String): boolean