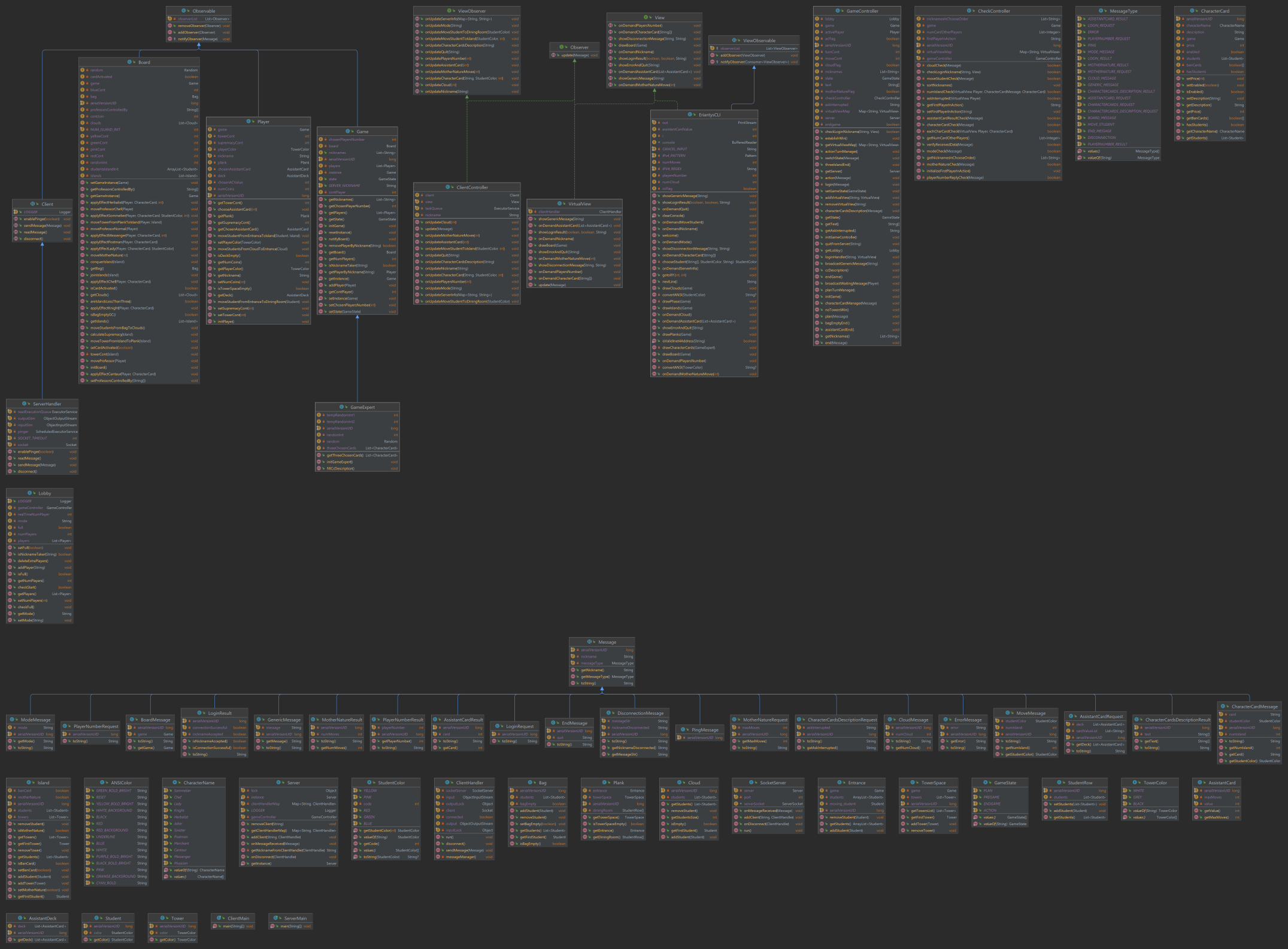
Observable
observerList List<Observer>
removeObserver(Observer) void
addObserver(Observer) void
notifyObserver(Message) void
ViewObserver
onUpdateServerInfo(Map<String, String>) void
onUpdateMode(String) void
onUpdateMoveStudentToDiningRoom(StudentColor) void
onUpdateMoveStudentToIsland(StudentColor, int) void
onUpdateCharacterCardsDescription(String) void
onUpdateQuit(String) void
onUpdatePlayersNumber(int) void
onUpdateAssistantCard(int) void
onUpdateMotherNatureMove(int) void
onUpdateCharacterCard(String, StudentColor, int) void
onUpdateCloud(int) void
onUpdateNickname(String) void
View
onDemandPlayersNumber() void
onDemandCharacterCard(String[]) void
showDisconnectionMessage(String, String) void
drawBoard(Game) void
onDemandNickname() void
showLoginResult(boolean, boolean, String) void
showErrorAndQuit(String) void
onDemandAssistantCard(List<AssistantCard>) void
showGenericMessage(String) void
onDemandMotherNatureMove(int) void
ViewObservable
observerList List<ViewObserver>
addObserver(ViewObserver) void
notifyObserver(Consumer<ViewObserver>) void
Observer
updateMessage() void
GameController
lobby Lobby
game Game
activePlayer Player
actFlag boolean
serialVersionUID long
turnCont int
moveCont int
cloudFlag boolean
nicknames List<String>
state GameState
text String[]
motherNatureFlag boolean
checkController CheckController
askInterrupted String
virtualViewMap Map<String, VirtualView>
server Server
endgame boolean
checkLoginNickname(String, View) boolean
establishWin() void
getVirtualViewMap() Map<String, VirtualView>
actionTurnManager() void
switchDate(Message) void
threeIslandEnd() void
getServer() Server
action(Message) void
login(Message) void
setGameState(GameState) void
addVirtualView(String, VirtualView) void
removeVirtualView(String) void
characterCardsDescription(Message) void
getState() GameState
getText() String[]
getAskInterrupted() String
initGameController() void
quitFromServer(String) void
getLobby() Lobby
loginHandle(String, VirtualView) void
broadcastGeneric(Message(String)) void
ccDescription() void
endGame() void
broadcastWaitingMessage(Player) void
planTurnManager() void
initGame() void
characterCardsManager(Message) void
notTowerWin() void
play(Message) void
bagEmptyEnd() void
assistantCardEnd() void
getNicknames() List<String>
end(Message) void
CheckController
nicknamesinChooseOrder List<String>
game Game
numCardOtherPlayers String
firstPlayerInAction String
serialVersionUID long
virtualViewMap Map<String, VirtualView>
gameController GameController
cloudCheck(Message) boolean
checkLoginNickname(String, View) boolean
moveStudentCheck(Message) boolean
setNicknames() void
numIslandCheck(VirtualView Player, CharacterCardMessage, CharacterCard) boolean
askInterrupted(VirtualView Player) boolean
getFirstPlayerInAction() String
setFirstPlayerInAction(String) void
assistantCardResult(Check(Message) boolean
characterCardCheck(Message) boolean
eachCharCardCheck(VirtualView Player, CharacterCard) boolean
getNumCardOtherPlayers() List<Integer>
verifyReceivedData(Message) boolean
modeCheck(Message) boolean
getNicknamesInChooseOrder() List<String>
motherNatureCheck(Message) boolean
initializeFirstPlayerInAction() void
playerNumberReplyCheck(Message) boolean
MessageType
ASSISTANTCARD_RESULT
LOGIN_REQUEST
ERROR
PLAYERNUMBER_REQUEST
PING
MODE_MESSAGE
LOGIN_RESULT
MOTHERNATURE_RESULT
MOTHERNATURE_REQUEST
CLOUD_MESSAGE
GENERIC_MESSAGE
CHARACTERCARDS_DESCRIPTION_RESULT
ASSISTANTCARD_REQUEST
CHARACTERCARDS_REQUEST
CHARACTERCARDS_DESCRIPTION_REQUEST
BOARD_MESSAGE
MOVE_STUDENT
END_MESSAGE
DISCONNECTION
PLAYERNUMBER_RESULT
values() MessageType[]
valueOf(String) MessageType
CharacterCard
serialVersionUID long
characterName CharacterName
description String
game Game
price int
enabled boolean
students List<Student>
banCards boolean[]
hasStudents boolean
setPrice(int) void
setEnabled(boolean) void
isEnabled() boolean
setDescription(String) void
getDescription() String
getPrice() int
getBanCards() boolean[]
hasStudents() boolean
getCharacterName() CharacterName
getStudents() List<Student>
Board
random Random
cardActivated boolean
game Game
turnCont int
bag Bag
serialVersionUID long
professionControlledBy String[]
coin int
clouds List<Cloud>
NUM_ISLAND_INIT int
yellowCont int
greenCont int
pinkCont int
redCont int
randomInt int
studentIslandInit ArrayList<Student>
islands List<Island>
setGameInstance(Game) void
getProfessionControlledBy() String[]
getGameInstance() Game
applyEffectHerbalist(Player, CharacterCard, int) void
moveProfessorChef(Player) void
applyEffectSommelier(Player, CharacterCard, StudentColor, int) void
moveTowerFromPlankToIsland(Player, Island) void
moveProfessorNormal(Player) void
applyEffectMessenger(Player, CharacterCard, int) void
applyEffectPoisman(Player, CharacterCard) void
applyEffectLady(Player, CharacterCard, StudentColor) void
moveMotherNature(int) void
conquerIsland(Island) void
getBag() Bag
joinIslands(Island) void
applyEffectChef(Player, CharacterCard) void
isCardActivated() boolean
getCloud(s) List<Cloud>
areIslandLessThanThree() boolean
applyEffectKnight(Player, CharacterCard) void
isBagEmptyGC() boolean
getIslands() List<Island>
moveStudentsFromBagToClouds() void
calculateSupremacy(Island) void
moveTowerFromIslandToPlank(Island) void
setCardActivated(boolean) void
towerContIsland() void
moveProfessor(Player) void
initBoard() void
applyEffectCentaur(Player, CharacterCard) void
setProfessionControlledBy(String[]) void
Player
game Game
towerCont int
supremacyCont int
playerColor TowerColor
nickname String
plank Plank
chosenAssistantCard AssistantCard
deck AssistantDeck
chosenACValue int
numCoins int
serialVersionID long
getTowerCont() int
chosenAssistantCard(int) void
getPlank() Plank
getSupremacyCont() int
getChosenAssistantCard() AssistantCard
moveStudentFromEntranceToIsland(Student, Island) void
setPlayerColorTowerColor void
moveStudentsFromCloudToEntrance(Cloud) void
isDeckEmpty() boolean
getNumCoins() int
getPlayerColor() TowerColor
getNickname() String
setNumCoins(int) void
isTowerSpaceEmpty() boolean
getDeck() AssistantDeck
moveStudentFromEntranceToDiningRoom(Student) void
setSupremacyCont(int) void
setTowerCont(int) void
initPlayer() void
Game
chosenPlayersNumber int
board Board
nicknames List<String>
serialVersionID long
players List<Player>
instance Game
state GameState
SERVER_NICKNAME String
contPlayer int
getNicknames() List<String>
getChosenPlayerNumber() int
getPlayers() List<Player>
getState() GameState
initGame() void
resetInstance() void
notifyBoard() void
removePlayerByNickname(String) boolean
getBoard() Board
getNumPlayers() int
isNicknameTaken(String) boolean
getPlayerByNickname(String) Player
getInstance() Game
addPlayer(Player) void
getContPlayer() int
setInstance(Game) void
setChosenPlayersNumber(int) void
setState(GameState) void
ClientController
client Client
view View
taskQueue ExecutorService
nickname String
onUpdateCloud(int) void
update(Message) void
onUpdateMotherNatureMove(int) void
onUpdateAssistantCard(int) void
onUpdateMoveStudentToIsland(StudentColor, int) void
onUpdateQuit(String) void
onUpdateCharacterCardsDescription(String) void
onUpdateNickname(String) void
onUpdateCharacterCard(String, StudentColor, int) void
onUpdatePlayersNumber(int) void
onUpdateMode(String) void
onUpdateServerInfo(Map<String, String>) void
onUpdateMoveStudentToDiningRoom(StudentColor) void
VirtualView
clientHandler ClientHandler
showGenericMessage(String) void
onDemandAssistantCard(List<AssistantCard>) void
showLoginResult(boolean, boolean, String) void
onDemandNickname() void
drawBoard(Game) void
showErrorAndQuit(String) void
onDemandMotherNatureMove(int) void
showDisconnectionMessage(String, String) void
onDemandPlayersNumber() void
onDemandCharacterCard(String[]) void
update(Message) void
EriantysCLI
out PrintStream
assistantCardValue int
s int
console BufferedReader
CANCEL_INPUT String
IPV4_PATTERN Pattern
numMoves int
IPV4_REGEX String
playerNumber int
numCloud int
ccFlag boolean
showGenericMessage(String) void
showLoginResult(boolean, boolean, String) void
onDemandQuit() void
clearConsole() void
onDemandMoveStudent() void
onDemandNickname() void
welcome() void
onDemandMode() void
showDisconnectionMessage(String, String) void
onDemandCharacterCard(String[]) void
chooseStudentS(String[], StudentColor, String) StudentColor
onDemandServerInfo() void
getXY(int, int) void
nextLine() String
drawClouds(Game) void
convertANS(StudentColor) String?
drawPlayer(Game) void
drawIslands(Game) void
onDemandCloud() void
onDemandAssistantCard(List<AssistantCard>) void
showErrorAndQuit(String) void
drawPlanks(Game) void
isValidIneteAddress(String) boolean
drawCharacterCards(GameExpert) void
drawBoard(Game) void
onDemandPlayersNumber() void
convertANS(TowerColor) String?
onDemandMotherNatureMove(int) void
Client
LOGGER Logger
enablePinger(boolean) void
sendMessage(Message) void
readMessage() void
disconnect() void
ServerHandler
readExecutionQueue ExecutorService
outputStm ObjectOutputStream
inputStm ObjectInputStream
pinger ScheduledExecutorService
SOCKET_TIMEOUT int
socket Socket
enablePinger(boolean) void
readMessage() void
sendMessage(Message) void
disconnect() void
GameExpert
tempRandomint1 int
tempRandomint2 int
serialVersionID long
randomint int
random Random
threeChosenCards List<CharacterCard>
getThreeChosenCards() List<CharacterCard>
initGameExpert() void
fillCCDescription() void
Lobby
LOGGER Logger
gameController GameController
realTimeNumPlayer int
mode String
full boolean
numPlayers int
players List<Player>
setFull(boolean) void
isNicknameTaken(String) boolean
deleteFirstPlayer() void
addPlayer(String) void
isFull() boolean
getNumPlayers() int
checkStart() boolean
getPlayers() List<Player>
setNumPlayers(int) void
checkFull() void
getMode() String
setMode(String) void
Message
serialVersionUID long
nickname String
messageType MessageType
getNickname() String
getMessageType() MessageType
toString() String
ModeMessage
mode String
serialVersionUID long
getMode() String
toString() String
PlayerNumberRequest
serialVersionUID long
toString() String
BoardMessage
serialVersionUID long
game Game
getGame() Game
LoginResult
serialVersionUID long
connectionSuccessful boolean
nicknameAccepted boolean
isNicknameAccepted() boolean
isConnectionSuccessful() boolean
toString() String
GenericMessage
serialVersionUID long
message String
getMessage() String
toString() String
MotherNatureResult
serialVersionUID long
numMoves int
toString() String
getNumMoves() int
PlayerNumberResult
serialVersionUID long
playerNumber int
getPlayerNumber() int
toString() String
AssistantCardResult
serialVersionUID long
card int
toString() String
getCard() int
LoginRequest
serialVersionUID long
toString() String
EndMessage
serialVersionUID long
out String
toString() String
DisconnectionMessage
messageInt String
nicknameDisconnected String
serialVersionUID long
toString() String
getNicknameDisconnected() String
getMessageInt() String
PingMessage
serialVersionUID long
MotherNatureRequest
maxMoves int
serialVersionUID long
getMaxMoves() int
toString() String
CharacterCardsDescriptionRequest
askInterrupted String
serialVersionUID long
toString() String
getAskInterrupted() String
CloudMessage
serialVersionUID long
numCloud int
toString() String
getNumCloud() int
ErrorMessage
error String
serialVersionUID long
getError() String
toString() String
MoveMessage
studentColor StudentColor
numIsland int
serialVersionUID long
toString() String
getNumIsland() int
getStudentColor() StudentColor
AssistantCardRequest
deck List<AssistantCard>
getValueList List<String>
serialVersionUID long
getDeck() List<AssistantCard>
toString() String
CharacterCardsDescriptionResult
serialVersionUID long
text String[]
getText() String[]
toString() String
CharacterCardMessage
card String
studentColor StudentColor
serialVersionUID long
numIsland int
toString() String
getNumIsland() int
getCard() String
getStudentColor() StudentColor
Island
banCard boolean
motherNature boolean
serialVersionUID long
students List<Student>
towers List<Tower>
removeStudent() void
isMotherNature() boolean
getTowers() List<Tower>
getFirstTower() Tower
removeTower() void
getStudents() List<Student>
isBanCard() boolean
setBanCard(boolean) void
addStudent(Student) void
addTower(Tower) void
setMotherNature(boolean) void
getFirstStudent() Student
ANSIColor
GREEN_BOLD_BRIGHT String
RESET String
YELLOW_BOLD_BRIGHT String
WHITE_BACKGROUND String
BLACK String
RED String
RED_BACKGROUND String
UNDERLINE String
BLUE String
WHITE String
PURPLE_BOLD_BRIGHT String
BLACK_BOLD_BRIGHT String
PINK String
ORANGE_BACKGROUND String
CYAN_BOLD String
CharacterName
Sommelier
Chef
Lady
Knight
Herbalist
Joker
Sinister
Postman
Merchant
Centaur
Messenger
Musician
valueOf(String) CharacterName
values() CharacterName[]
Server
lock Object
instance Server
clientHandlerMap Map<String, ClientHandler>
LOGGER Logger
gameController GameController
removeClient(String) void
getClientHandlerMap() Map<String, ClientHandler>
addClient(String, ClientHandler) void
onMessageReceived(Message) void
getNicknameFromClientHandler(ClientHandler) String
onDisconnect(ClientHandler) void
getInstance() Server
StudentColor
YELLOW
PINK
code int
RED
GREEN
BLUE
getStudentColor(int) StudentColor
valueOf(String) StudentColor
getCode() int
values() StudentColor[]
toString(StudentColor) String?
ClientHandler
socketServer SocketServer
input ObjectInputStream
outputLock Object
client Socket
connected boolean
output ObjectOutputStream
inputLock Object
run() void
disconnect() void
sendMessage(Message) void
messageManager() void
Bag
serialVersionUID long
students List<Student>
bagEmpty boolean
addStudent(Student) void
removeStudent() void
setBagEmpty(boolean) void
getStudents() List<Student>
getFirstStudent() Student
isBagEmpty() boolean
Plank
entrance Entrance
towerSpace TowerSpace
serialVersionUID long
diningRoom StudentRow[]
getTowerSpace() TowerSpace
isTowerSpaceEmpty() boolean
getEntrance() Entrance
getDiningRoom() StudentRow[]
Cloud
serialVersionUID long
students List<Student>
getStudents() List<Student>
removeStudent() void
getStudentsSize() int
isEmpty() boolean
getFirstStudent() Student
addStudent(Student) void
SocketServer
server Server
port int
serverSocket ServerSocket
onMessageReceived(Message) void
addClient(String, ClientHandler) void
onDisconnect(ClientHandler) void
run() void
Entrance
game Game
students ArrayList<Student>
moving_student Student
serialVersionUID long
removeStudent(Student) void
getStudents() ArrayList<Student>
addStudent(Student) void
TowerSpace
game Game
towers List<Tower>
serialVersionUID long
getTowersList() List<Tower>
getFirstTower() Tower
addTower(Tower) void
removeTower() void
GameState
PLAN
PREGAME
ENDGAME
ACTION
values() GameState[]
valueOf(String) GameState
StudentRow
serialVersionUID long
students List<Student>
setStudents(List<Student>) void
addStudent(Student) void
getStudents() List<Student>
TowerColor
WHITE
GREY
BLACK
valueOf(String) TowerColor
values() TowerColor[]
AssistantCard
serialVersionUID long
maxMoves int
value int
getValue() int
getMaxMoves() int
AssistantDeck
deck List<AssistantCard>
serialVersionUID long
getDeck() List<AssistantCard>
Student
serialVersionUID long
color StudentColor
getColor() StudentColor
Tower
serialVersionUID long
color TowerColor
getColor() TowerColor
ClientMain
main(String[]) void
ServerMain
main(String[]) void