

model

Table
<code>NUMCARDS : int</code>
<code>reserve : ArrayList<Dice></code>
<code>activePlayers : ArrayList<Player></code>
<code>canExtract : boolean</code>
<code>roundTrack : ArrayList<Dice></code>
<code>redExt : int</code>
<code>greenExt : int</code>
<code>purpleExt : int</code>
<code>yellowExt : int</code>
<code>blueExt : int</code>
<code>round : int</code>
<code>clockwise : boolean</code>
<code>turn : int</code>
<code>numPlayers : int</code>
<code>realTurn : int</code>
<code>numSchemes : int</code>
<code>custom : boolean</code>
<code>Table(int)</code>
<code>hasPlayer(nicknames : List<String>)</code>
<code>nextTurn()</code>
<code>nextRound()</code>
<code>pickDiceFromBag()</code> : Dice
<code>pickDiceFromReserve(int)</code> : Dice
<code>pickDiceFromRoundtrack(int)</code> : Dice
<code>putDiceInBag(int)</code> : boolean
<code>rerollReserve()</code> : boolean
<code>checkDiceFromReserve(int)</code> : Dice
<code>checkDiceFromRoundtrack(int)</code> : Dice
<code>putDiceInReserve(Dice)</code>
<code>putDiceInRoundtrack(Dice)</code>
<code>useToolCard(int, Player, IHandler) : boolean</code>
<code>setCanExtract(boolean)</code>
<code>getTurn()</code> : int
<code>getRealTurn()</code> : int
<code>getRound()</code> : int
<code>getActivePlayers() : List<Player></code>
<code>getReserve()</code> : String
<code>getRoundtrack()</code> : String
<code>setCustom()</code> : boolean
<code>getNumSchemes()</code> : int

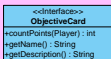
Player
<code>nickname : String</code>
<code>points : int</code>
<code>tokens : int</code>
<code>disconnected : Dice</code>
<code>ownScheme : Scheme</code>
<code>tool8 : boolean</code>
<code>IDPlayer : int</code>
<code>disconnected : boolean</code>
<code>+Player(String, int)</code>
<code>+chooseScheme(Scheme)</code>
<code>+retractDice(x : int, y : int, Table, indexDice : int) : boolean</code>
<code>+placeDice(x : int, y : int, Table, indexDice : int) : boolean</code>
<code>+countPoints()</code>
<code>+pointsInPrivObj()</code> : int
<code>+getOwnScheme() : Scheme</code>
<code>+getPlayer() : int</code>
<code>+getUsername() : String</code>
<code>+getTokens() : int</code>
<code>+getTool8() : boolean</code>
<code>+decrementTokens(int)</code>
<code>+getDiceInHand() : Dice</code>
<code>+setDisconnected(boolean)</code>
<code>+isDisconnected() : boolean</code>
<code>+setTool8(boolean)</code>
<code>+getTool8() : boolean</code>
<code>+addObserver(Observer)</code>
<code>+notifyObservers(Object)</code>

Scheme
<code>name : String</code>
<code>difficulty : int</code>
<code>MAX_COL : int</code>
<code>MAX_ROW : int</code>
<code>STD_NUM_SCHEMES : int</code>
<code>grid : Box[]</code>
<code>+Scheme(int, boolean)</code>
<code>+initialize(int, custom : boolean, numSchemes : int)</code>
<code>+checkBox(x : int, y : int) : Dice</code>
<code>firstDiceNum(x : int, y : int, Dice, boolean) : boolean</code>
<code>firstDiceNoCol(x : int, y : int, Dice, boolean) : boolean</code>
<code>firstDiceNoCol(x : int, y : int, Dice, boolean) : boolean</code>
<code>checkColorRestr(x : int, y : int, Dice, boolean) : boolean</code>
<code>isPlaceableNoDiceNear(x : int, y : int, Dice) : boolean</code>
<code>placeFromTool(x : int, y : int, int, Dice) : boolean</code>
<code>isPlaceable(x : int, y : int, Dice, boolean) : boolean</code>
<code>placeDice(x : int, y : int, Dice) : boolean</code>
<code>isGridEmpty() : boolean</code>
<code>removeDice(x : int, y : int) : Dice</code>
<code>getName() : String</code>
<code>getDifficulty() : int</code>
<code>getGrid() : Box[]</code>
<code>+addObserver(Observer)</code>
<code>+notifyObservers(Object)</code>

Box
<code>restrictionCol : Color</code>
<code>restrictionNum : int</code>
<code>isFull : boolean</code>
<code>disallowed : Dice</code>
<code>+Box()</code>
<code>+Box(restrictionCol : Color)</code>
<code>+Box(restrictionNum : int)</code>
<code>+isEmployableNoCol(Dice, boolean) : boolean</code>
<code>+isEmployableNoNum(Dice, boolean) : boolean</code>
<code>+isFull() : boolean</code>
<code>setDice(Dice) : boolean</code>
<code>getDice() : Dice</code>
<code>getRestrictionCol() : Color</code>
<code>getRestrictionNum() : int</code>
<code>isFull() : boolean</code>
<code>notifyObservers(Object)</code>

Dice
<code>value : int</code>
<code>color : Color</code>
<code>+Dice()</code>
<code>+Dice(col : Color)</code>
<code>+Dice(val : int)</code>
<code>+Dice(col : Color, val : int)</code>
<code>+rollDice()</code>
<code>+assignValue(int)</code>
<code>+getValue() : int</code>
<code>+getColor() : Color</code>
<code>turnDice()</code>

Color
<code>RED</code>
<code>GREEN</code>
<code>YELLOW</code>
<code>BLUE</code>
<code>PURPLE</code>
<code>+RESET : String</code>
<code>escape : String</code>
<code>Color(String)</code>
<code>rescape() : String</code>
<code>useRandomColor() : Color</code>



PrivObjHandler
<code>activeID : ArrayList<PrivateOC></code>
<code>usePrivOC(int)</code>
<code>+countPoints(Player) : int</code>
<code>+getName(Player) : String</code>
<code>+getDescription(Player) : String</code>
<code>+getColor(Player) : Color</code>
<code>+getCard(Player) : String</code>

PrivateOC
<code>name : String</code>
<code>description : String</code>
<code>color : Color</code>
<code>+PrivateOC(Color)</code>
<code>+getName() : String</code>
<code>+getDescription() : String</code>
<code>+countPoints(Player) : int</code>
<code>+getColor() : Color</code>

PublicOC
<code>name : String</code>
<code>description : String</code>
<code>cardID : int</code>
<code>+PublicOC(int)</code>
<code>+getName() : String</code>
<code>+getDescription() : String</code>
<code>+countPoints(Player) : int</code>
<code>+publicOC1(Player) : int</code>
<code>+publicOC2(Player) : int</code>
<code>+publicOC3(Player) : int</code>
<code>+publicOC4(Player) : int</code>
<code>+publicOC5(Player) : int</code>
<code>+publicOC6(Player) : int</code>
<code>+publicOC7(Player) : int</code>
<code>+publicOC8(Player) : int</code>
<code>+publicOC9(Player) : int</code>
<code>+publicOC10(Player) : int</code>

PublicObjHandler
<code>activeID : ArrayList<PublicOC></code>
<code>NUM_PUB_OC : int</code>
<code>usePUB_OC(int)</code>
<code>+countPoints(Player, int) : int</code>
<code>+getName(int) : String</code>
<code>+getDescription(int) : String</code>

ToolCard
<code>name : String</code>
<code>description : String</code>
<code>tokens : int</code>
<code>cardID : int</code>
<code>+ToolCard(int)</code>
<code>+useEffect : UseEffect()</code>
<code>+ToolCard(int)</code>
<code>+getName() : String</code>
<code>+getDescription() : String</code>
<code>+getTokens() : int</code>
<code>+toolEffect(Player, Table) : boolean</code>
<code>+getCoordinates(Player) : int[]</code>
<code>+isDiceMovement(int, Player, Color) : boolean</code>
<code>+chooseFromRoundtrack(int, Player, Table) : Dice</code>
<code>+extractFromReserve(Player, Table) : Dice</code>
<code>+extractDice(Player, int[])</code> : boolean
<code>+modifyDiceValue(int, Player, Dice) : boolean</code>
<code>+moveDiceIndex : int, Player, coordinates : int[]</code> : boolean
<code>+isPlaceable(Dice, Player) : String</code>
<code>+putDiceInBag(Player, Table) : boolean</code>
<code>+incrementTokens(Player)</code>

ToolHandler
<code>activeID : ArrayList<ToolCard></code>
<code>NUM_TOOLCS : int</code>
<code>currentID : IHandler</code>
<code>+setTool()</code>
<code>+useTool(int, Player, Table, IHandler) : boolean</code>
<code>+getCoordinates(Player) : int</code>
<code>+getFromRoundtrack(Player) : int</code>
<code>+getFromReserve(Player) : int</code>
<code>+setDiceValue(boolean, Player) : int</code>
<code>+setYesOrNo(Player) : boolean</code>
<code>+tool8()</code>
<code>+getName(int) : String</code>
<code>+getDescription(int) : String</code>
<code>+getTokens(int) : int</code>

