Monopoly

Generated by Doxygen 1.8.6

Tue Feb 10 2015 20:20:19

Contents

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 2

Class Documentation

2.1 BoardLocation Class Reference

`#include <boardlocation.h>`

Public Member Functions

- BoardLocation (int id, std::string name)
- BoardLocation (int id, std::string name, int price, std::string group)
- BoardLocation (int id, std::string name, int price, std::string group, std::vector< int > rents, int houseCost)
- int getID ()
- std::string * getName ()
- int getPrice ()
- int getRents (int position)
- int getHouseCost ()
- std::string * getGroup ()
- int getBuildings ()
- int getVisits ()
- bool isMortgaged ()
- void setMortgaged (bool val)
- int **getUnmortgagePrice** ()
- void **mortgage** ()
- void **unmortgage** ()
- void incrementVisits ()
- void changeBuildings (int)
- float getAuctionModifier ()

Static Public Member Functions

- static int getGroupSize (std::string *groupName)

2.1.1 Detailed Description

A class meant to represent a location in a Monopoly gameboard.

2.1.2 Constructor & Destructor Documentation

2.1.2.1 BoardLocation::BoardLocation (int *id,* std::string *name*)

Basic BoardLocation constructor for non-properties

Needed for creation of some of the "non-property" spaces.

Parameters

id	Numerical id of the location, where 0 is Go and 40 is Boardwalk
name	Friendly name of the location

2.1.2.2 BoardLocation::BoardLocation (int *id,* std::string *name,* int *price,* std::string *group*)

More detailed BoardLocation constructor for railroads, etc

This is what some properties such as railroads use.

Parameters

id	Numerical id of the location, where 0 is Go and 40 is Boardwalk
name	Friendly name of the location
price	Integer price of the property
group	Color group the property belongs to, e.g. "Red"

2.1.2.3 BoardLocation::BoardLocation (int *id,* std::string *name,* int *price,* std::string *group,* std::vector< int > *rents,* int *houseCost*)

Most detailed BoardLocation constructor for properties

This is what most properties use as it fills in most information.

Parameters

id	Numerical id of the location, where 0 is Go and 40 is Boardwalk
name	Friendly name of the location
price	Integer price of the property
group	Color group the property belongs to, e.g. "Red"
rents	List of the 6 rents on property (singleton, monopoly, 1-4 houses, hotel)
houseCost	The cost to buy/build one house on the property

2.1.3 Member Function Documentation

2.1.3.1 void BoardLocation::changeBuildings (int *delta*)

Changes the number of buildings on the property by a given quantity

Increases or decreases the buildings by the count. Doesn't do any sanity checking.

Parameters

delta	the change in buildings

2.1.3.2 float BoardLocation::getAuctionModifier ( )

Gets a modifier between 0 and 1 for auction bidding

Right now these are arbitrarily set, but work is being done to come up with actual useful numbers.

Returns

a value between 0 and 1 based on color group

2.1.3.3 int BoardLocation::getBuildings ( )

Gets the number of buildings on the property

0: no buildings 1-4: that many houses 5: a hotel

Returns

the number of buildings on the property

2.1.3.4 std::string * BoardLocation::getGroup ( )

Gets the color group the property belongs to, e.g. "Red"

Returns

a pointer to the color group

2.1.3.5 int BoardLocation::getGroupSize (std::string * *groupName*) [static]

Returns the size of the given color group

Gets the size of the color group: 2, 3, or 4, otherwise returns -1 if it's not a property.

Parameters

groupName	the string group from the property

Returns

2, 3, or 4 depending on group; else -1

2.1.3.6 int BoardLocation::getHouseCost ( )

Gets the cost to buy/build a house

Returns

the cost in Monopoly dollars

2.1.3.7 int BoardLocation::getID ( )

Gets the space ID (0..40) of the location

Returns

numerical positional identifier

2.1.3.8 std::string * BoardLocation::getName ( )

Gets the friendly name of the property

Returns

pointer to a string containing the name

2.1.3.9 int BoardLocation::getPrice ( )

Gets the purchase price of the property, as displayed on the card

Returns

price of property

2.1.3.10 int BoardLocation::getRents (int *position*)

Gets the rent for a specific case

Parameters

position	the rent to get, where 0 is a singleton and 5 is a hotel

Returns

the rent in Monopoly dollars

2.1.3.11 int BoardLocation::getVisits ( )

Gets the number of times the property has been visited

Returns

the counter of times a player has landed on property

2.1.3.12 void BoardLocation::incrementVisits ( )

Increments the visit counter of the space

When a player lands on a space, it needs to be incremented by one. This is more convenient and accurate than a pure setter.

2.1.3.13 bool BoardLocation::isMortgaged ( )

Is this property currently mortgaged (i.e. out of play)?

Returns

true if mortgaged, else false

2.1.3.14 void BoardLocation::setMortgaged (bool *val*)

Set the mortgaged value of the property

Parameters

val	true if mortgaged, else false

The documentation for this class was generated from the following files:

- include/game/boardlocation.h
- src/game/boardlocation.cpp

2.2 CSV Class Reference

Public Member Functions

- **CSV** (std::string filename)
- void **writeline** (int *player, float result)

The documentation for this class was generated from the following files:

- include/util/csv.h
- src/util/csv.cpp

2.3 endReport Struct Reference

Public Attributes

- int **winner**
- int **turnCounter**
- std::unordered_set< std::string * > **player0Monopolies**
- std::unordered_set< std::string * > **player1Monopolies**
- int **player0Money**
- int **player1Money**

The documentation for this struct was generated from the following file:

- include/game/game.h

2.4 Game Class Reference

Public Member Functions

- **Game** (std::vector< Player * > players, int cutoff)
- **Game** (std::vector< Player * > players, bool auct, bool fpp, bool dog, bool nrij, bool tts, bool seb, int cutoff)
- void **communityChest** (Player *player)
- void **chance** (Player *player)
- void **addCommunityChestCardToDeck** ()
- void **addChanceCardToDeck** ()
- void **moveAhead** (Player *player, int numberOfSpaces)
- void **moveTo** (Player *player, int location)
- void **payOutOfJail** (Player *player)
- void **goToJail** (Player *player)
- void **buyProperty** (Player *player, BoardLocation *boardSpace, int customPrice=0)
- Player * **propertyOwner** (BoardLocation *property)

- void **payRent** (Player *player)
- int **unmortgagePrice** (BoardLocation *property)
- void **sellBuilding** (Player *player, BoardLocation *property, std::string building)
- void **exchangeMoney** (Player *giver, Player *receiver, int amount)
- void **exchangeMoney** (Player *giver, MoneyPool *receiver, int amount)
- void **exchangeMoney** (MoneyPool *giver, Player *receiver, int amount)
- void **exchangeMoney** (Player *giver, int amount)
- bool **evenSellingTest** (BoardLocation *property, Player *player)
- bool **evenBuildingTest** (BoardLocation *property, Player *player)
- bool **mortgageCheck** (BoardLocation *property, Player *player)
- void **developProperties** (Player *player)
- bool **monopolyStatus** (Player *player, BoardLocation *boardSpace)
- bool **monopolyStatus** (Player *player, BoardLocation *boardSpace, std::vector< BoardLocation * > *additionalProperties)
- int **findAvailableMortgageValue** (Player *player)
- void **auction** (BoardLocation *property)
- int **totalAssets** (Player *player)
- void **propertyAction** (Player *player, BoardLocation *boardSpace)
- void **boardAction** (Player *player, BoardLocation *boardSpace)
- void **takeTurn** (Player *player)
- void **updateStatus** ()
- endReport **play** ()
- int **rollDie** ()
- int **chooseRandomPlayer** ()
- int **getAvailableHouses** ()
- int **getAvailableHotels** ()
- void **changeHouses** (int delta)
- void **changeHotels** (int delta)
- bool **hasFirstBuilding** ()
- void **removeFromUnownedProperties** (BoardLocation *boardSpace)

The documentation for this class was generated from the following files:

- include/game/game.h
- src/game/game.cpp

2.5 MoneyPool Class Reference

Public Member Functions

- **MoneyPool** (int start)
- int **getMoney** ()
- void **setMoney** (int newMoney)
- void **addMoney** (int add)
- char **getType** ()

The documentation for this class was generated from the following files:

- include/game/moneypool.h
- src/game/moneypool.cpp

2.6 Player Class Reference

Public Member Functions

- **Player** (int num, std::unordered_set< std::string * > groupPreferences, int buy_thresh, int build_thresh, int jt, bool sjs, int cm, int dt, int maw)
- **Player** (const int *parameters)
- void **resetValues** ()
- void **changePosition** (int delta)
- void **passGo** ()
- void **appendToInventory** (BoardLocation *boardSpace)
- bool **isInInventory** (BoardLocation *boardSpace)
- void **appendToMonopolies** (std::string *group)
- bool **isInMonopolies** (std::string *group)
- bool **isInGroupPreferences** (std::string group)
- void **payOutOfJail** (Game *game)
- bool **buyProperty** (BoardLocation *property, int customPrice)
- void **developProperties** (Game *game)
- void **sellBuilding** (BoardLocation *property, std::string building, Game *game)
- void **makeFunds** (Game *game)
- void **setJailStrategy** (Game *game)
- bool **evenSellingTest** (BoardLocation *property)
- bool **evenBuildingTest** (BoardLocation *property)
- int **findAvailableMortgageValue** ()
- int **makeBid** (Game *game, BoardLocation *property)
- bool **unownedPropertyAction** (Game *game, BoardLocation *property)
- bool **jailDecision** (Game *game)
- bool **completesMonopoly** (BoardLocation *property)
- int **getNumber** ()
- void **setNumber** (int num)
- int **getBuyingThreshold** ()
- int **getBuildingThreshold** ()
- bool **hasPassedGo** ()
- int **getJailTime** ()
- void **setJailTime** (int jailtime)
- int **getInitJailTime** ()
- bool **hasSmartJailStrategy** ()
- int **getCompleteMonopoly** ()
- std::unordered_set < std::string * > * **getGroupPreferences** ()
- int **getDevelopmentThreshold** ()
- void **giveCommunityChestCard** ()
- void **takeCommunityChestCard** ()
- bool **hasCommunityChestCard** ()
- void **giveChanceCard** ()
- void **takeChanceCard** ()
- bool **hasChanceCard** ()
- std::unordered_set < std::string * > * **getMonopolies** ()
- std::unordered_set < BoardLocation * > * **getInventory** ()
- int **getPosition** ()
- void **setPosition** (int pos)
- bool **isInJail** ()
- void **setInJail** (bool jail)

- int **getMoney** ()
- void **setMoney** (int newMoney)
- void **addMoney** (int add)
- bool **getBidIncludesMortgages** ()
- void **setBidIncludesMortgages** (bool bid)
- bool **getCardRent** ()
- void **setCardRent** (bool rent)
- int **getJailCounter** ()
- void **setJailCounter** (int count)
- void **incrementJailCounter** ()
- void **endGame** ()
- int ∗ **getInfo** ()

The documentation for this class was generated from the following files:

- include/game/player.h
- src/game/player.cpp

Index