Tic-Tac-Toe

Generated by Doxygen 1.14.0

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Chapter 2

Class Index

2.1 Class List

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

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

Chapter 4

Namespace Documentation

4.1 CLASSES Namespace Reference

Classes

- class Game

  Tic-Tac-Toe game class.

4.1.1 Detailed Description

```
@file CLASSES.py
@brief Tic-Tac-Toe game realisation
@author Yevik A. 421702
```

4.2 main Namespace Reference

Functions

- start_game ()

Variables

- greeting = str(input("Wanna start a new game? y/n: "))

4.2.1 Detailed Description

```
@file main.py
@brief Program file that is executed
```

4.2.2 Function Documentation

4.2.2.1 start_game()

```
main.start_game ()
```

```
@brief Start game
@details A function that runs the gaming process
@see Game
@see Game.turn()
@see Game.show_field()
@see Game.get_symbol()
@see Game.game_winner()
@see Game.next_move()
```

4.2.3 Variable Documentation

4.2.3.1 greeting

```
main.greeting = str(input("Wanna start a new game?  y/n:  "))
```

4.3 test_classes Namespace Reference

Functions

- test_field ()
- test_next_move ()
- test_turn ()
- test_occupied ()
- test_invalid ()
- test_winner_row ()
- test_winner_column ()
- test_winner_diagonals ()
- test_draw ()

4.3.1 Detailed Description

```
@file test_classes.py
@brief Tests for Game class
```

4.3.2 Function Documentation

4.3.2.1 test_draw()

```
test_classes.test_draw ()
```

4.3.2.2 test_field()

```
test_classes.test_field ()
```

4.3.2.3 test_invalid()

```
test_classes.test_invalid ()
```

4.3.2.4 test_next_move()

```
test_classes.test_next_move ()
```

4.3.2.5 test_occupied()

```
test_classes.test_occupied ()
```

4.3.2.6 test_turn()

```
test_classes.test_turn ()
```

4.3.2.7 test_winner_column()

```
test_classes.test_winner_column ()
```

4.3.2.8 test_winner_diagonals()

```
test_classes.test_winner_diagonals ()
```

4.3.2.9 test_winner_row()

```
test_classes.test_winner_row ()
```

4.4 tests Namespace Reference

Chapter 5

Class Documentation

5.1 CLASSES.Game Class Reference

Tic-Tac-Toe game class.

Public Member Functions

- __init__ (self, int field_size)
- get_symbol (self)
- get_field (self)
- get_field_size (self)
- set_field (self, row, column, symbol)
- show_field (self)
- next_move (self)
- turn (self, int row, int column)
- game_winner (self)

5.1.1 Detailed Description

Tic-Tac-Toe game class.

Controls the field, players turns and defines the winner

5.1.2 Constructor & Destructor Documentation

5.1.2.1 __init__()

```
CLASSES.Game.__init__ (
              self,
             int field_size)
```

```
@brief Constructor
:param field_size: The size of a game field
```

5.1.3 Member Function Documentation

5.1.3.1 game_winner()

```
CLASSES.Game.game_winner (
              self)
```

```
@brief Game winner
@details A method to implement the mechanics of game winning or a draw
:return: bool game is won by one of the players
@see turn
```

5.1.3.2 get_field()

```
CLASSES.Game.get_field (
              self)
```

```
@brief Field getter
@details A method to get the field
:return: List[list] game field
```

5.1.3.3 get_field_size()

```
CLASSES.Game.get_field_size (
              self)
```

```
@brief Field size getter
@details A method to get the size
:return: int field size
```

5.1.3.4 get_symbol()

```
CLASSES.Game.get_symbol (
              self)
```

```
@brief Symbol getter
@details A method to get the current game symbol
:return: str game symbol
```

5.1.3.5 next_move()

```
CLASSES.Game.next_move (
              self)
```

```
@brief Game next move
@details Implements players turn change mechanics
```

5.1.3.6 set_field()

```
CLASSES.Game.set_field (
              self,
              row,
              column,
              symbol)


@brief Field cell setter
:param row: row
:param column: column
:param symbol: symbol
```

5.1.3.7 show_field()

```
CLASSES.Game.show_field (
              self)


@brief Show field
@details A method to show the game field
```

5.1.3.8 turn()

```
CLASSES.Game.turn (
              self,
             int row,
             int column)


@brief Game turn
@details A method to implement the mechanics of cell occupation
:param row: row
:param column: column
@see next_move
@see game_winner
```

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

- CLASSES.py

Chapter 6

File Documentation

6.1 CLASSES.py File Reference

Classes

- class CLASSES.Game

  Tic-Tac-Toe game class.

Namespaces

- namespace CLASSES

6.2 main.py File Reference

Namespaces

- namespace main

Functions

- main.start_game ()

Variables

- main.greeting = str(input("Wanna start a new game? y/n: "))

6.3 test_classes.py File Reference

Namespaces

- namespace test_classes

Functions

- test_classes.test_field ()
- test_classes.test_next_move ()
- test_classes.test_turn ()
- test_classes.test_occupied ()
- test_classes.test_invalid ()
- test_classes.test_winner_row ()
- test_classes.test_winner_column ()
- test_classes.test_winner_diagonals ()
- test_classes.test_draw ()

6.4 tests.py File Reference

Namespaces

- namespace tests

Index