tetris-pygame

Release 0.0.1

Łukasz Łukaszewski

Dec 17, 2023

CONTENTS

1 tetris package **1**

1.1 Subpackages 1

1.2 Submodules 5

1.3 Module contents 19

2 Indices and tables **21**

Python Module Index **23**

Index **25**

CHAPTER

ONE

TETRIS PACKAGE

1.1 Subpackages

1.1.1 tetris.db package

Subpackages

tetris.db.models package

Submodules

tetris.db.models.get_base_dir module

tetris.db.models.user module

`class tetris.db.models.user.User(*args, **kwargs)`

Bases: `Model`

Model for user table

DoesNotExist

alias of `UserDoesNotExist`

games_played = <IntegerField: User.games_played>

highest_score = <IntegerField: User.highest_score>

id = <AutoField: User.id>

lvl = <IntegerField: User.lvl>

username = <CharField: User.username>

`tetris.db.models.user.on_save_handler(model_class, instance, created)` → None

Increments games_played field on save

`tetris.db.models.user.user_exists(username)` → bool

Checks if a user exists in the database

Module contents

tetris.db.scripts package

Submodules

tetris.db.scripts.get_base_dir module

tetris.db.scripts.migration module

`tetris.db.scripts.migration.`**`crete_db`**`()` → None

Migrates the database

`tetris.db.scripts.migration.`**`reset_db`**`()` → None

Resets the database

Module contents

Submodules

tetris.db.settings module

Module contents

1.1.2 tetris.tests package

Submodules

tetris.tests.test_game module

`class` `tetris.tests.test_game.`**`TestGame`**`(`*methodName='runTest'*`)`

Bases: `TestCase`

A test case class for testing the Game class.

`CHECK_LINE_FALSE_GRID = [[0, 0, 0, 0, 0, 0, 0, 0, 0, 0], ['I', 'I', 'I', 'I', 'I', 'I', 'I', 'I', 'I', 'I'], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]]`

CHECK_LINE_TRUE_GRID = [[0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], ['I', 'I', 'I', 'I', 'I', 'I', 'I', 'I', 'I', 'I'], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]]

CHECK_TETROMINO_GRID = [[0, 0, 0, 0, 0, 0, 0, 0, 0, 0], ['I', 'I', 'I', 'I', 'I', 'I', 'I', 'I', 'I', 'I'], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]]

CLEAR_GRID = [[0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]]

CLEAR_NEXT_TETROMINO_GRID = [[0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0], [0, 0, 0, 0]]

DELETE_LINE_AFTER_GRID = [[0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 'L', 0, 0, 0, 0], [0, 0, 0, 'L', 'L', 'L', 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 'O', 'O', 0, 0, 0, 0], ['I', 'I', 'I', 'I', 'O', 'O', 0, 0, 0, 0], [0, 'T', 0, 0, 0, 'Z', 0, 'J', 'J', 0], [0, 'T', 'T', 0, 'Z', 'Z', 0, 'J', 'O', 'O'], [0, 'T', 0, 0, 'Z', 0, 0, 'J', 'O', 'O'], ['Z', 'Z', 0, 'O', 'O', 0, 0, 0, 0, 'L'], [0, 'Z', 'Z', 'O', 'O', 'L', 0, 'L', 'L', 'L'], [0, 'Z', 'Z', 'L', 'L', 'L', 0, 'Z', 'Z', 0], ['O', 'O', 0, 'O', 'O', 0, 0, 0, 0, 0], [0, 0, 0, 'O', 'O', 0, 0, 0, 0, 0], [0, 0, 0, 0, 'S', 'S', 0, 0, 0, 0], [0, 0, 0, 'S', 'S', 0, 0, 'O', 'O', 0], [0, 0, 'L', 'O', 'O', 0, 0, 'O', 'O', 'L'], ['L', 'L', 'L', 'O', 'O', 0, 0, 'L', 'L', 'L']]

DELETE_LINE_BEFORE_GRID = [[0, 0, 0, 0, 0, 'L', 0, 0, 0, 0], [0, 0, 0, 'L', 'L', 'L', 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 0, 0, 0, 0, 0, 0], [0, 0, 0, 0, 'O', 'O', 0, 0, 0, 0], ['I', 'I', 'I', 'I', 'O', 'O', 0, 0, 0, 0], [0, 'T', 0, 0, 0, 'Z', 0, 'J', 'J', 0], [0, 'T', 'T', 0, 'Z', 'Z', 0, 'J', 'O', 'O'], [0, 'T', 0, 0, 'Z', 0, 0, 'J', 'O', 'O'], ['Z', 'Z', 0, 'O', 'O', 0, 0, 0, 0, 'L'], [0, 'Z', 'Z', 'O', 'O', 'L', 0, 'L', 'L', 'L'], [0, 'Z', 'Z', 'L', 'L', 'L', 0, 'Z', 'Z', 0], ['O', 'O', 'Z', 'Z', 'I', 'I', 'I', 'I', 'Z', 'Z'], ['O', 'O', 0, 'O', 'O', 0, 0, 0, 0, 0], [0, 0, 0, 'O', 'O', 0, 0, 0, 0, 0], [0, 0, 0, 0, 'S', 'S', 0, 0, 0, 0], [0, 0, 0, 'S', 'S', 0, 0, 'O', 'O', 0], [0, 0, 'L', 'O', 'O', 0, 0, 'O', 'O', 'L'], ['L', 'L', 'L', 'O', 'O', 0, 0, 'L', 'L', 'L']] `classmethod setUpClass()` → None

Set up the test class by initializing the settings and game objects.

`test_check_line()` → None

Test the check_line method of the Game class.

`test_check_tetromino_above_top()` → None

Test the check_tetromino_above_top method of the Game class.

`test_delete_line()` → None

Test the delete_line method of the Game class.

`test_init_properties()` → None

Test the init_properties method of the Game class.

`test_random_tetromino()` → None

Test the random_tetromino method of the Game class.

tetris.tests.test_menu module

`class tetris.tests.test_menu.TestMenu(`*methodName='runTest'*`)`

Bases: `TestCase`

Test case class for testing the Menu class.

`setUp()` → None

Hook method for setting up the test fixture before exercising it.

`test_main(`*mock_display_update*, *mock_draw_buttons*, *mock_check_hover*, *mock_check_events*`)` → None

tetris.tests.test_tetrominos module

`class tetris.tests.test_tetrominos.TestTetromino(`*methodName='runTest'*`)`

Bases: `TestCase`

Test case class for testing the Tetromino class.

`CHECK_DOWN_POS = [[21, 3], [21, 4], [21, 5], [21, 6]]`

`LEFT_EDGE_POS = [[1, 0], [1, 1], [1, 2], [1, 3]]`

`RIGHT_EDGE_POS = [[1, 9], [1, 8], [1, 7], [1, 6]]`

`classmethod setUpClass()` → None

Set up the test class by initializing the settings, game, and tetromino objects.

Parameters

`cls` – The class object.

Returns

None

`test_check_down()` → None

Test the check down method of the Tetromino class.

test_check_move_left() → None

Test the check move left method of the Tetromino class.

test_check_move_right() → None

Test the check move right method of the Tetromino class.

test_check_touch() → None

Test the check touch method of the Tetromino class.

Module contents

1.2 Submodules

1.2.1 tetris.controls module

class tetris.controls.**Controls**(*settings:* Settings)

Bases: `object`

Class for the controls screen

check_events() → bool | None

Checks for events

Returns

True if go back button is pressed, None otherwise

Return type

(bool | None)

create_controls() → None

Creates the controls

create_title() → None

Creates the title of the controls

draw_controls() → None

Draws the controls

draw_title() → None

Draws the controls title

main() → None

Main method for the controls screen, draws everything and checks for events

1.2.2 tetris.game module

class tetris.game.**Game**(*settings:* Settings)

Bases: `object`

Class contains main game logic and methods to draw game elements

add_score(*lines_cleared: int*) → None

Add score for cleared lines

`check_events()` → None

Check pygane events and react to them

`check_hover()` → None

Checks if mouse is hovering over the buttons and changes the cursor accordingly

`check_line()` → bool

Check if there is a line of tetrominos and delete it

Parameters
`None` –

Returns
True if there is a line of tetrominos, False otherwise

Return type
(bool)

`check_lvl_up()` → None

`check_pressed_down_movement()` → None

Check if down movement keys are pressed and react to them

`check_pressed_rotate()` → None

Check if rotation keys are pressed and react to them

`check_pressed_side_movement()` → None

Check if side movement keys are pressed and react to them

`check_tetromino_above_top()` → bool

Check if there is a tetromino above visible top of the grid

`create_game_windows()` → None

Create game windows

`db_insert_user()` → None

Insert user into database

`delete_line(`*row*`)` → None

Delete line from grid and move all tetrominos above it down

`draw_end_of_game_btns()` → None

Draw buttons displayed at the end of the game, after loss

`draw_game_window()` → None

Draw game window on screen

`draw_grid()` → None

Draw grid with tetrominos on game window

`draw_lvl()` → None

Draw score window and score on

`draw_lvl_title()` → None

Draw score title on screen

`draw_lvl_window()` → None

Draw score window on screen

draw_next_tetromino() → None

Draw next tetromino on next tetromino window

draw_next_tetromino_title() → None

Draw next tetromino title on screen

draw_next_tetromino_window() → None

Draw next tetromino window on screen

draw_score() → None

Draw score window and score on

draw_score_title() → None

Draw score title on screen

draw_score_window() → None

Draw score window on screen

game_window: **Surface**

game_window_rect: **Rect**

get_username() → None

grid: **list[list[int]]**

init_properties() → None

Initialize game properties

lines_cleared: **int**

lvl: **int**

lvl_up() → None

lvl_window: **Surface**

lvl_window_rect: **Rect**

main() → None

Main game loop

move_down_key_pressed: **bool = False**

next_game() → bool

Check if user clicked on next game button or menu button

next_tetromino_grid: **list[list[int]]**

next_tetromino_window: **Surface**

next_tetromino_window_rect: **Rect**

print_grid() → None

Print grid in console

random_tetromino() → *Tetromino*

Return random tetromino

Returns
Random tetromino

Return type
(*Tetromino*)

score: int

score_window: Surface

score_window_rect: Rect

space_down: bool = False

username: str

1.2.3 tetris.leaderboard module

class tetris.leaderboard.**Leaderboard**(*settings:* Settings)

Bases: `object`

Class for the leaderboard screen

check_events() → bool | None

Checks for events

Returns
True if go back button is pressed, None otherwise

Return type
(bool | None)

create_draw_row(*rank: int*, *user:* User) → None

Creates and draws a row of the leaderboard

create_header() → None

Creates the header of the leaderboard

create_leaderboard() → None

Creates the leaderboard

create_title() → None

Creates the title of the leaderboard

draw_header() → None

Draws the leaderboard header

draw_title() → None

Draws the leaderboard title

main() → None

Main function of the leaderboard screen that draws everything and checks for events

1.2.4 tetris.main module

`tetris.main.main()` → None

Main function of the game, creates the settings, menu and game objects and runs main functions of them in the loop

1.2.5 tetris.menu module

class `tetris.menu.Menu`(*settings:* Settings)

Bases: `object`

Menu class for the game

`CONTROLS_CHOICE: str = 'controls'`

`GAME_CHOICE: str = 'game'`

`LEADERBOARD_CHOICE: str = 'leaderboard'`

`QUIT_CHOICE: str = 'quit'`

`RULES_CHOICE: str = 'rules'`

`check_events()` → bool | None

Checks for pygame events QUIT and MOUSEBUTTONDOWN and reacts to them

Returns

True if game, leaderboard, controls or rules button is pressed, None otherwise

Return type

(bool | None)

`check_hover()` → None

Checks if mouse is hovering over the buttons and changes the cursor accordingly

`choice:  str = ''`

`create_buttons()` → None

Creates menu buttons and its rects

`draw_buttons()` → None

Draws the buttons on the screen

`main()` → str

Main menu loop, waits for the user to press a button and returns the choice

Returns

Choice of the user

Return type

(str)

1.2.6 tetris.rules module

class `tetris.rules.`**`Rules`**(*settings:* Settings)

Bases: `object`

A class to show the rules of the game

`TEXT: list[str] = ['The goal of the game is to get the highest score', 'possible. You earn points by moving down tetrominos', '1 point for soft drop and two points for hard drop.', 'You can also earn points by clearing lines.', '1 line is 100 points, 2 lines is 300 points,', '3 lines is 500 points, and 4 lines is 800 points.', 'Each 10 lines cleared, the level increases.', 'The level increases the speed of the tetrominos.', 'The game ends when the tetrominos reach', 'the top of the screen.']`

`check_events``()` → bool | None

Checks for events

Returns
True if go back button is pressed, None otherwise

Return type
(bool | None)

`create_draw_text``()` → None

Creates and draws the text of the rules

`create_title``()` → None

Creates the title of the controls

`draw_title``()` → None

Draws the controls title

`main``()` → None

Main method for the controls screen

1.2.7 tetris.settings module

class `tetris.settings.`**`Settings`**

Bases: `object`

Settings class for the game

`BG_COLOR: tuple = (0, 0, 100)`

`CELL_BORDER_COLOR: tuple = (119, 136, 153)`

`CHECK_KEYS_PRESSED_MOVEMENT_DOWN_TIME: int = 50`

`CHECK_KEYS_PRESSED_MOVEMENT_SIDE_TIME: int = 80`

`CHECK_KEYS_PRESSED_ROTATION_TIME: int = 150`

`CONTROLS_ONE_KEY_X: float = 650.0`

`CONTROLS_TEXTS: list[str] = ['move down', 'move left', 'move right', 'hard drop', 'rotate left', 'rotate right']`

CONTROLS_TEXTS_X: float = 333.3333333333333

CONTROLS_TEXTS_Y: list[float] = [320.0, 400.0, 480.0, 560.0, 640.0, 720.0]

CONTROLS_TITLE: str = 'CONTROLS'

CONTROLS_TITLE_Y: int = 160

CONTROLS_TWO_KEYS_X: list[float] = [600.0, 700.0]

EMPTY_CELL_TAG: int = 0

END_OF_BTNS_COLOR: tuple = (218, 165, 32)

END_OF_GAME_BTNS_HEIGHT: int = 44

END_OF_GAME_BTNS_WIDTH: int = 125

END_OF_GAME_BTNS_Y: int = 400

END_OF_GAME_MENU_BTN_TEXT: str = 'Menu'

END_OF_GAME_MENU_BTN_X: int = 345

END_OF_GAME_NEXT_BTN_TEXT: str = 'Next'

END_OF_GAME_NEXT_BTN_X: int = 530

FONT_COLOR: tuple = (255, 255, 255)

FONT_NAME: str = 'Tahoma'

FONT_SIZE_CONTROLS: int = 30

FONT_SIZE_CONTROLS_TITLE: int = 40

FONT_SIZE_END_OF_GAME_BTNS: int = 20

FONT_SIZE_GET_USERNAME: int = 23

FONT_SIZE_INFO_TITLES: int = 30

FONT_SIZE_LEADERBOARD: int = 15

FONT_SIZE_LEADERBOARD_HEADER: int = 20

FONT_SIZE_LEADERBOARD_TITLE: int = 40

FONT_SIZE_RULES: int = 30

FONT_SIZE_RULES_TITLE: int = 40

FONT_SIZE_SCORE_LVL: int = 30

FONT_SIZE_TETRIS_TITLE: int = 80

FPS: int = 60

GAME_BORDER_COLOR: tuple = (255, 0, 0)

GAME_WINDOW_HEIGHT: int = 600

GAME_WINDOW_WIDTH: int = 400

GET_USERNAME_INPUT_BOX_HEIGHT: int = 40

GET_USERNAME_INPUT_BOX_WIDTH: int = 250

GET_USERNAME_INPUT_BOX_X: float = 588.2352941176471

GET_USERNAME_INPUT_BOX_Y: float = 392.0

GET_USERNAME_TEXT: str = 'ENTER YOUR USERNAME (ENTER)'

GO_BACK_BTN_X: int = 30

GO_BACK_BTN_Y: int = 40

GO_BACK_ICON_FILENAME: str =
'C:\\studia\\tetris\\tetris\\tetris\\assets\\arrow_back.png'

GRID_CELL_HEIGHT: int = 30

GRID_CELL_WIDTH: int = 40

GRID_N_OF_COL: int = 10

GRID_N_OF_ROWS: int = 22

HARD_DROP_KEY_FILENAME: str =
'C:\\studia\\tetris\\tetris\\tetris\\assets\\space_key.png'

HARD_DROP_LOOP_SLEEP_TIME: float = 0.01

I: tuple = (0, 255, 255)

INFO_WINDOW_HEIGHT: int = 150

INFO_WINDOW_WIDTH: int = 219

J: tuple = (0, 0, 255)

L: tuple = (255, 165, 0)

LEADERBOARD_BORDER_COLOR: tuple = (128, 128, 128)

LEADERBOARD_BORDER_WIDTH: int = 1

LEADERBOARD_BORDER_X: float = 62.5

LEADERBOARD_FIRST_ROW_Y: float = 352.22222222222223

LEADERBOARD_HEADERS_TEXTS: list[str] = ['RANK', 'USERNAME', 'SCORE', 'LVL', 'GAMES PLAYED']

LEADERBOARD_HEADER_BORDER_WIDTH: int = 3

LEADERBOARD_HEADER_HEIGHT: int = 100

LEADERBOARD_HEADER_TEXT_Y: float = 272.22222222222223

LEADERBOARD_HEADER_Y: float = 222.22222222222223

LEADERBOARD_ROW_HEIGHT: float = 40.0

LEADERBOARD_TEXT_WIDTH: float = 175.0

LEADERBOARD_TEXT_X: list[float] = [107.14285714285715, 282.14285714285717, 500.0, 675.0, 850.0]

LEADERBOARD_TITLE: str = 'LEADERBOARD'

LEADERBOARD_TITLE_Y: int = 160

LEADERBOARD_WIDTH: float = 875.0

LVL_TITLE: str = 'LEVEL'

LVL_TITLE_Y: float = 410.2564102564103

LVL_WINDOW_X: int = 40

LVL_WINDOW_Y: float = 444.44444444444446

MENU_BTNS_FIRST_Y: int = 200

MENU_BTNS_HEIGHT: int = 100

MENU_BTN_GAP: int = 20

MENU_CONTROLS_TEXT = 'Controls'

MENU_LEADERBOARD_TEXT = 'Leaderboard'

MENU_QUIT_TEXT = 'Quit'

MENU_RULES_TEXT = 'Rules'

MENU_START_GAME_TEXT = 'Start game'

MOVE_DOWN_ACCELERATION_PER_LVL: int = 19

MOVE_DOWN_START_TIME: int = 1000

MOVING_DOWN_KEY_1_FILENAME: str = 'C:\\studia\\tetris\\tetris\\tetris\\assets\\s_key.png'

MOVING_DOWN_KEY_2_FILENAME: str = 'C:\\studia\\tetris\\tetris\\tetris\\assets\\down_key.png'

MOVING_LEFT_KEY_1_FILENAME: str = 'C:\\studia\\tetris\\tetris\\tetris\\assets\\a_key.png'

MOVING_LEFT_KEY_2_FILENAME: str = 'C:\\studia\\tetris\\tetris\\tetris\\assets\\left_key.png'

MOVING_RIGHT_KEY_1_FILENAME: str = 'C:\\studia\\tetris\\tetris\\tetris\\assets\\d_key.png'

MOVING_RIGHT_KEY_2_FILENAME: str = 'C:\\studia\\tetris\\tetris\\tetris\\assets\\right_key.png'

NEXT_TETROMINO_CELL_HEIGHT: int = 37

NEXT_TETROMINO_CELL_WIDTH: int = 54

NEXT_TETROMINO_N_OF_COL: int = 4

NEXT_TETROMINO_N_OF_ROWS: int = 4

NEXT_TETROMINO_TITLE: str = 'NEXT'

NEXT_WINDOW_X: float = 740.7407407407406

NEXT_WINDOW_Y: int = 200

N_OF_LINES_TO_LVL_UP: int = 10

O: tuple = (255, 255, 0)

POINTS_FOR_HARD_DROP: int = 2

POINTS_FOR_SOFT_DROP: int = 1

POINTS_PER_LINES: dict = {1: 100, 2: 300, 3: 500, 4: 800}

ROTATE_LEFT_KEY_FILENAME: str =
'C:\\studia\\tetris\\tetris\\tetris\\assets\\z_key.png'

ROTATE_RIGHT_KEY_1_FILENAME: str =
'C:\\studia\\tetris\\tetris\\tetris\\assets\\w_key.png'

ROTATE_RIGHT_KEY_2_FILENAME: str =
'C:\\studia\\tetris\\tetris\\tetris\\assets\\up_key.png'

RULES_LINE_HEIGHT: int = 40

RULES_TEXT_Y: float = 266.6666666666667

RULES_TITLE: str = 'RULES'

RULES_TITLE_Y: int = 160

S: tuple = (0, 255, 0)

SCORE_NEXT_TITLE_Y: int = 160

SCORE_TITLE: str = 'SCORE'

SCORE_WINDOW_X: int = 40

SCORE_WINDOW_Y: int = 200

SCREEN_HEIGHT: int = 800

SCREEN_WIDTH: int = 1000

SECOND_BG_COLOR: tuple = (0, 0, 0)

T: tuple = (128, 0, 128)

TETRIS_TITLE: str = 'TETRIS'

TETRIS_TITLE_Y: int = 50

Z: tuple = (255, 0, 0)

check_go_back_btn_hover() → None

Checks if mouse is hovering over the go_back button and changes the cursor accordingly

create_end_of_game_btns() → None

Creates buttons displayed after the game is lost

create_get_username_text() → None

Creates the input box for the username

create_go_back_btn() → None

Creates the go back button

create_titles_properties() → None

Create properties for the titles

draw_go_back_btn() → None

Draws the go back button

draw_tetris_title() → None

Draws the tetris title

1.2.8 tetris.tetrominos module

class tetris.tetrominos.**Itetromino**(*game:* Game)

Bases: *Tetromino*

NEXT_TETROMINO_GRID_POS: list[list] = [[1, 0], [1, 1], [1, 2], [1, 3]]

SPAWN_POS: list[list] = [[1, 3], [1, 4], [1, 5], [1, 6]]

TAG: str = 'I'

pos0() → None

Rotates the tetromino to position 0

pos1() → None

Rotates the tetromino to position 1

pos2() → None

Rotates the tetromino to position 2

pos3() → None

Rotates the tetromino to position 3

rotate_left() → None

Rotates the tetromino left

rotate_right() → None

Rotates the tetromino right

class tetris.tetrominos.**Jtetromino**(*game:* Game)

Bases: *Tetromino*

NEXT_TETROMINO_GRID_POS: list[list] = [[1, 0], [2, 0], [2, 1], [2, 2]]

SPAWN_POS: list[list] = [[0, 3], [1, 4], [1, 3], [1, 5]]

TAG: str = 'J'

pos0() → None

Set the tetromino to position 0

pos1() → None

Set the tetromino to position 1

pos2() → None

Set the tetromino to position 2

pos3() → None

Set the tetromino to position 3

class tetris.tetrominos.**Ltetromino**(*game:* Game)

Bases: *Tetromino*

NEXT_TETROMINO_GRID_POS: list[list] = [[2, 0], [2, 1], [2, 2], [1, 2]]

SPAWN_POS: list[list] = [[1, 3], [1, 4], [1, 5], [0, 5]]

TAG: str = 'L'

pos0() → None

Set the tetromino to position 0

pos1() → None

Set the tetromino to position 1

pos2() → None

Set the tetromino to position 2

pos3() → None

Set the tetromino to position 3

class tetris.tetrominos.**Otetromino**(*game:* Game)

Bases: *Tetromino*

NEXT_TETROMINO_GRID_POS: list[list] = [[1, 1], [2, 1], [1, 2], [2, 2]]

SPAWN_POS: list[list] = [[1, 4], [1, 5], [0, 4], [0, 5]]

TAG: str = 'O'

rotate_left() → None

Do nothing because the tetromino is a square

rotate_right() → None

Do nothing because the tetromino is a square

class tetris.tetrominos.**Stetromino**(*game:* Game)

Bases: *Tetromino*

NEXT_TETROMINO_GRID_POS: list[list] = [[2, 0], [2, 1], [1, 1], [1, 2]]

SPAWN_POS: list[list] = [[1, 3], [1, 4], [0, 4], [0, 5]]

TAG: str = 'S'

pos0() → None

Set the tetromino to position 0

pos1() → None

Set the tetromino to position 1

pos2() → None

Set the tetromino to position 2

pos3() → None

Set the tetromino to position 3

class tetris.tetrominos.**Tetromino**(*game:* Game)

Bases: `object`

Tetromino class for the game

LAST_COL_IDX: int = 9

NEXT_TETROMINO_GRID_POS: list[list]

SPAWN_POS: list[list]

TAG: str

check_cell_available_for_rotation(*cell: list[int]*) → bool

Check if the cell is available for rotation

Parameters

cell (*list[int]*) – Cell to check

Returns

True if the cell is available for rotation, False otherwise

Return type

(bool)

check_down() → bool

Checks if the tetromino is at the bottom of the grid

Returns

True if the tetromino is at the bottom of the grid, False otherwise

Return type

(bool)

check_move_left() → bool

Check if the tetromino can move left by checking if it is at the left edge of the grid or if it is touching another tetromino on the left

Returns

True if the tetromino can move left, False otherwise

Return type

(bool)

`check_move_right()` → bool

Check if the tetromino can move right by checking if it is at the right edge of the grid or if it is touching another tetromino on the right

Returns
True if the tetromino can move right, False otherwise

Return type
(bool)

`check_touch()` → bool

Check if the bottom of the tetromino is touching another tetromino

Returns
True if the bottom of the tetromino is touching another tetromino, False otherwise

Return type
(bool)

`clear()` → None

Clears the grid from the tetromino

`clear_next_tetromino_window()` → None

Clears the next tetromino window

`hard_drop()` → None

Hard drops the tetromino

`move_down()` → bool | None

Moves the tetromino down

`move_left()` → None

Moves the tetromino left

`move_right()` → None

Moves the tetromino right

`pos: list[list]`

`pos0()` → None

Rotates the tetromino to position 0

`pos1()` → None

Rotates the tetromino to position 1

`pos2()` → None

Rotates the tetromino to position 2

`pos3()` → None

Rotates the tetromino to position 3

`put_on_next_tetromino_window()` → None

Puts the next tetromino on the next tetromino window

`rotate_left()` → None

Rotates the tetromino left

`rotate_right()` → None

Rotates the tetromino right

spawn() → None

Set the spawn position

update_on_grid() → None

Updates the grid with the new position of the tetromino

class tetris.tetrominos.**Ttetromino**(*game:* Game)

Bases: *Tetromino*

NEXT_TETROMINO_GRID_POS: list[list] = [[2, 0], [2, 1], [2, 2], [1, 1]]

SPAWN_POS: list[list] = [[1, 2], [1, 3], [1, 4], [0, 3]]

TAG: str = 'T'

pos0() → None

Set the tetromino to position 0

pos1() → None

Set the tetromino to position 1

pos2() → None

Set the tetromino to position 2

pos3() → None

Set the tetromino to position 3

class tetris.tetrominos.**Ztetromino**(*game:* Game)

Bases: *Tetromino*

NEXT_TETROMINO_GRID_POS: list[list] = [[1, 0], [1, 1], [2, 1], [2, 2]]

SPAWN_POS: list[list] = [[0, 3], [1, 4], [0, 4], [1, 5]]

TAG: str = 'Z'

pos0() → None

Set the tetromino to position 0

pos1() → None

Set the tetromino to position 1

pos2() → None

Set the tetromino to position 2

pos3() → None

Set the tetromino to position 3

1.3 Module contents

CHAPTER

TWO

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

t

tetris, 19
tetris.controls, 5
tetris.db, 2
tetris.db.models, 2
tetris.db.models.get_base_dir, 1
tetris.db.models.user, 1
tetris.db.scripts, 2
tetris.db.scripts.get_base_dir, 2
tetris.db.scripts.migration, 2
tetris.db.settings, 2
tetris.game, 5
tetris.leaderboard, 8
tetris.main, 9
tetris.menu, 9
tetris.rules, 10
tetris.settings, 10
tetris.tests, 5
tetris.tests.test_game, 2
tetris.tests.test_menu, 4
tetris.tests.test_tetrominos, 4
tetris.tetrominos, 15

INDEX

A

add_score() (*tetris.game.Game method*), 5

B

BG_COLOR (*tetris.settings.Settings attribute*), 10

C

CELL_BORDER_COLOR (*tetris.settings.Settings attribute*), 10

check_cell_available_for_rotation() (*tetris.tetrominos.Tetromino method*), 17

check_down() (*tetris.tetrominos.Tetromino method*), 17

CHECK_DOWN_POS (*tetris.tests.test_tetrominos.TestTetromino attribute*), 4

check_events() (*tetris.controls.Controls method*), 5

check_events() (*tetris.game.Game method*), 5

check_events() (*tetris.leaderboard.Leaderboard method*), 8

check_events() (*tetris.menu.Menu method*), 9

check_events() (*tetris.rules.Rules method*), 10

check_go_back_btn_hover() (*tetris.settings.Settings method*), 15

check_hover() (*tetris.game.Game method*), 6

check_hover() (*tetris.menu.Menu method*), 9

CHECK_KEYS_PRESSED_MOVEMENT_DOWN_TIME (*tetris.settings.Settings attribute*), 10

CHECK_KEYS_PRESSED_MOVEMENT_SIDE_TIME (*tetris.settings.Settings attribute*), 10

CHECK_KEYS_PRESSED_ROTATION_TIME (*tetris.settings.Settings attribute*), 10

check_line() (*tetris.game.Game method*), 6

CHECK_LINE_FALSE_GRID (*tetris.tests.test_game.TestGame attribute*), 2

CHECK_LINE_TRUE_GRID (*tetris.tests.test_game.TestGame attribute*), 2

check_lvl_up() (*tetris.game.Game method*), 6

check_move_left() (*tetris.tetrominos.Tetromino method*), 17

check_move_right() (*tetris.tetrominos.Tetromino method*), 17

check_pressed_down_movement() (*tetris.game.Game method*), 6

check_pressed_rotate() (*tetris.game.Game method*), 6

check_pressed_side_movement() (*tetris.game.Game method*), 6

check_tetromino_above_top() (*tetris.game.Game method*), 6

CHECK_TETROMINO_GRID (*tetris.tests.test_game.TestGame attribute*), 3

check_touch() (*tetris.tetrominos.Tetromino method*), 18

choice (*tetris.menu.Menu attribute*), 9

clear() (*tetris.tetrominos.Tetromino method*), 18

CLEAR_GRID (*tetris.tests.test_game.TestGame attribute*), 3

CLEAR_NEXT_TETROMINO_GRID (*tetris.tests.test_game.TestGame attribute*), 3

clear_next_tetromino_window() (*tetris.tetrominos.Tetromino method*), 18

Controls (*class in tetris.controls*), 5

CONTROLS_CHOICE (*tetris.menu.Menu attribute*), 9

CONTROLS_ONE_KEY_X (*tetris.settings.Settings attribute*), 10

CONTROLS_TEXTS (*tetris.settings.Settings attribute*), 10

CONTROLS_TEXTS_X (*tetris.settings.Settings attribute*), 10

CONTROLS_TEXTS_Y (*tetris.settings.Settings attribute*), 11

CONTROLS_TITLE (*tetris.settings.Settings attribute*), 11

CONTROLS_TITLE_Y (*tetris.settings.Settings attribute*), 11

CONTROLS_TWO_KEYS_X (*tetris.settings.Settings attribute*), 11

create_buttons() (*tetris.menu.Menu method*), 9

create_controls() (*tetris.controls.Controls method*), 5

create_draw_row() (*tetris.leaderboard.Leaderboard method*), 8

create_draw_text() (*tetris.rules.Rules method*), 10

create_end_of_game_btns() (*tetris.settings.Settings method*), 15

create_game_windows() (*tetris.game.Game method*), 6 create_get_username_text() (*tetris.settings.Settings method*), 15

create_go_back_btn() (*tetris.settings.Settings method*), 15

create_header() (*tetris.leaderboard.Leaderboard method*), 8

create_leaderboard() (*tetris.leaderboard.Leaderboard method*), 8

create_title() (*tetris.controls.Controls method*), 5

create_title() (*tetris.leaderboard.Leaderboard method*), 8

create_title() (*tetris.rules.Rules method*), 10

create_titles_properties() (*tetris.settings.Settings method*), 15

crete_db() (*in module tetris.db.scripts.migration*), 2

D

db_insert_user() (*tetris.game.Game method*), 6

delete_line() (*tetris.game.Game method*), 6

DELETE_LINE_AFTER_GRID (*tetris.tests.test_game.TestGame attribute*), 3

DELETE_LINE_BEFORE_GRID (*tetris.tests.test_game.TestGame attribute*), 3

DoesNotExist (*tetris.db.models.user.User attribute*), 1

draw_buttons() (*tetris.menu.Menu method*), 9

draw_controls() (*tetris.controls.Controls method*), 5

draw_end_of_game_btns() (*tetris.game.Game method*), 6

draw_game_window() (*tetris.game.Game method*), 6

draw_go_back_btn() (*tetris.settings.Settings method*), 15

draw_grid() (*tetris.game.Game method*), 6

draw_header() (*tetris.leaderboard.Leaderboard method*), 8

draw_lvl() (*tetris.game.Game method*), 6

draw_lvl_title() (*tetris.game.Game method*), 6

draw_lvl_window() (*tetris.game.Game method*), 6

draw_next_tetromino() (*tetris.game.Game method*), 6

draw_next_tetromino_title() (*tetris.game.Game method*), 7

draw_next_tetromino_window() (*tetris.game.Game method*), 7

draw_score() (*tetris.game.Game method*), 7

draw_score_title() (*tetris.game.Game method*), 7

draw_score_window() (*tetris.game.Game method*), 7

draw_tetris_title() (*tetris.settings.Settings method*), 15

draw_title() (*tetris.controls.Controls method*), 5

draw_title() (*tetris.leaderboard.Leaderboard method*), 8

draw_title() (*tetris.rules.Rules method*), 10

E

EMPTY_CELL_TAG (*tetris.settings.Settings attribute*), 11

END_OF_BTNS_COLOR (*tetris.settings.Settings attribute*), 11

END_OF_GAME_BTNS_HEIGHT (*tetris.settings.Settings attribute*), 11

END_OF_GAME_BTNS_WIDTH (*tetris.settings.Settings attribute*), 11

END_OF_GAME_BTNS_Y (*tetris.settings.Settings attribute*), 11

END_OF_GAME_MENU_BTN_TEXT (*tetris.settings.Settings attribute*), 11

END_OF_GAME_MENU_BTN_X (*tetris.settings.Settings attribute*), 11

END_OF_GAME_NEXT_BTN_TEXT (*tetris.settings.Settings attribute*), 11

END_OF_GAME_NEXT_BTN_X (*tetris.settings.Settings attribute*), 11

F

FONT_COLOR (*tetris.settings.Settings attribute*), 11

FONT_NAME (*tetris.settings.Settings attribute*), 11

FONT_SIZE_CONTROLS (*tetris.settings.Settings attribute*), 11

FONT_SIZE_CONTROLS_TITLE (*tetris.settings.Settings attribute*), 11

FONT_SIZE_END_OF_GAME_BTNS (*tetris.settings.Settings attribute*), 11

FONT_SIZE_GET_USERNAME (*tetris.settings.Settings attribute*), 11

FONT_SIZE_INFO_TITLES (*tetris.settings.Settings attribute*), 11

FONT_SIZE_LEADERBOARD (*tetris.settings.Settings attribute*), 11

FONT_SIZE_LEADERBOARD_HEADER (*tetris.settings.Settings attribute*), 11

FONT_SIZE_LEADERBOARD_TITLE (*tetris.settings.Settings attribute*), 11

FONT_SIZE_RULES (*tetris.settings.Settings attribute*), 11

FONT_SIZE_RULES_TITLE (*tetris.settings.Settings attribute*), 11

FONT_SIZE_SCORE_LVL (*tetris.settings.Settings attribute*), 11

FONT_SIZE_TETRIS_TITLE (*tetris.settings.Settings attribute*), 11

FPS (*tetris.settings.Settings attribute*), 11

G

Game (*class in tetris.game*), 5

GAME_BORDER_COLOR (*tetris.settings.Settings attribute*), 11

GAME_CHOICE (*tetris.menu.Menu attribute*), 9

game_window (*tetris.game.Game attribute*), 7

GAME_WINDOW_HEIGHT (*tetris.settings.Settings attribute*), 11
game_window_rect (*tetris.game.Game attribute*), 7
GAME_WINDOW_WIDTH (*tetris.settings.Settings attribute*), 11
games_played (*tetris.db.models.user.User attribute*), 1
get_username() (*tetris.game.Game method*), 7
GET_USERNAME_INPUT_BOX_HEIGHT (*tetris.settings.Settings attribute*), 12
GET_USERNAME_INPUT_BOX_WIDTH (*tetris.settings.Settings attribute*), 12
GET_USERNAME_INPUT_BOX_X (*tetris.settings.Settings attribute*), 12
GET_USERNAME_INPUT_BOX_Y (*tetris.settings.Settings attribute*), 12
GET_USERNAME_TEXT (*tetris.settings.Settings attribute*), 12
GO_BACK_BTN_X (*tetris.settings.Settings attribute*), 12
GO_BACK_BTN_Y (*tetris.settings.Settings attribute*), 12
GO_BACK_ICON_FILENAME (*tetris.settings.Settings attribute*), 12
grid (*tetris.game.Game attribute*), 7
GRID_CELL_HEIGHT (*tetris.settings.Settings attribute*), 12
GRID_CELL_WIDTH (*tetris.settings.Settings attribute*), 12
GRID_N_OF_COL (*tetris.settings.Settings attribute*), 12
GRID_N_OF_ROWS (*tetris.settings.Settings attribute*), 12

H

hard_drop() (*tetris.tetrominos.Tetromino method*), 18
HARD_DROP_KEY_FILENAME (*tetris.settings.Settings attribute*), 12
HARD_DROP_LOOP_SLEEP_TIME (*tetris.settings.Settings attribute*), 12
highest_score (*tetris.db.models.user.User attribute*), 1

I

I (*tetris.settings.Settings attribute*), 12
id (*tetris.db.models.user.User attribute*), 1
INFO_WINDOW_HEIGHT (*tetris.settings.Settings attribute*), 12
INFO_WINDOW_WIDTH (*tetris.settings.Settings attribute*), 12
init_properties() (*tetris.game.Game method*), 7
Itetromino (*class in tetris.tetrominos*), 15

J

J (*tetris.settings.Settings attribute*), 12
Jtetromino (*class in tetris.tetrominos*), 15

L

L (*tetris.settings.Settings attribute*), 12
LAST_COL_IDX (*tetris.tetrominos.Tetromino attribute*), 17
Leaderboard (*class in tetris.leaderboard*), 8
LEADERBOARD_BORDER_COLOR (*tetris.settings.Settings attribute*), 12
LEADERBOARD_BORDER_WIDTH (*tetris.settings.Settings attribute*), 12
LEADERBOARD_BORDER_X (*tetris.settings.Settings attribute*), 12
LEADERBOARD_CHOICE (*tetris.menu.Menu attribute*), 9
LEADERBOARD_FIRST_ROW_Y (*tetris.settings.Settings attribute*), 12
LEADERBOARD_HEADER_BORDER_WIDTH (*tetris.settings.Settings attribute*), 12
LEADERBOARD_HEADER_HEIGHT (*tetris.settings.Settings attribute*), 12
LEADERBOARD_HEADER_TEXT_Y (*tetris.settings.Settings attribute*), 12
LEADERBOARD_HEADER_Y (*tetris.settings.Settings attribute*), 12
LEADERBOARD_HEADERS_TEXTS (*tetris.settings.Settings attribute*), 12
LEADERBOARD_ROW_HEIGHT (*tetris.settings.Settings attribute*), 12
LEADERBOARD_TEXT_WIDTH (*tetris.settings.Settings attribute*), 13
LEADERBOARD_TEXT_X (*tetris.settings.Settings attribute*), 13
LEADERBOARD_TITLE (*tetris.settings.Settings attribute*), 13
LEADERBOARD_TITLE_Y (*tetris.settings.Settings attribute*), 13
LEADERBOARD_WIDTH (*tetris.settings.Settings attribute*), 13
LEFT_EDGE_POS (*tetris.tests.test_tetrominos.TestTetromino attribute*), 4
lines_cleared (*tetris.game.Game attribute*), 7
Ltetromino (*class in tetris.tetrominos*), 16
lvl (*tetris.db.models.user.User attribute*), 1
lvl (*tetris.game.Game attribute*), 7
LVL_TITLE (*tetris.settings.Settings attribute*), 13
LVL_TITLE_Y (*tetris.settings.Settings attribute*), 13
lvl_up() (*tetris.game.Game method*), 7
lvl_window (*tetris.game.Game attribute*), 7
lvl_window_rect (*tetris.game.Game attribute*), 7
LVL_WINDOW_X (*tetris.settings.Settings attribute*), 13
LVL_WINDOW_Y (*tetris.settings.Settings attribute*), 13

M

main() (*in module tetris.main*), 9
main() (*tetris.controls.Controls method*), 5
main() (*tetris.game.Game method*), 7
main() (*tetris.leaderboard.Leaderboard method*), 8
main() (*tetris.menu.Menu method*), 9
main() (*tetris.rules.Rules method*), 10
Menu (*class in tetris.menu*), 9
MENU_BTN_GAP (*tetris.settings.Settings attribute*), 13

MENU_BTNS_FIRST_Y (*tetris.settings.Settings attribute*), 13

MENU_BTNS_HEIGHT (*tetris.settings.Settings attribute*), 13

MENU_CONTROLS_TEXT (*tetris.settings.Settings attribute*), 13

MENU_LEADERBOARD_TEXT (*tetris.settings.Settings attribute*), 13

MENU_QUIT_TEXT (*tetris.settings.Settings attribute*), 13

MENU_RULES_TEXT (*tetris.settings.Settings attribute*), 13

MENU_START_GAME_TEXT (*tetris.settings.Settings attribute*), 13

module
- tetris, 19
- tetris.controls, 5
- tetris.db, 2
- tetris.db.models, 2
- tetris.db.models.get_base_dir, 1
- tetris.db.models.user, 1
- tetris.db.scripts, 2
- tetris.db.scripts.get_base_dir, 2
- tetris.db.scripts.migration, 2
- tetris.db.settings, 2
- tetris.game, 5
- tetris.leaderboard, 8
- tetris.main, 9
- tetris.menu, 9
- tetris.rules, 10
- tetris.settings, 10
- tetris.tests, 5
- tetris.tests.test_game, 2
- tetris.tests.test_menu, 4
- tetris.tests.test_tetrominos, 4
- tetris.tetrominos, 15

move_down() (*tetris.tetrominos.Tetromino method*), 18

MOVE_DOWN_ACCELERATION_PER_LVL (*tetris.settings.Settings attribute*), 13

move_down_key_pressed (*tetris.game.Game attribute*), 7

MOVE_DOWN_START_TIME (*tetris.settings.Settings attribute*), 13

move_left() (*tetris.tetrominos.Tetromino method*), 18

move_right() (*tetris.tetrominos.Tetromino method*), 18

MOVING_DOWN_KEY_1_FILENAME (*tetris.settings.Settings attribute*), 13

MOVING_DOWN_KEY_2_FILENAME (*tetris.settings.Settings attribute*), 13

MOVING_LEFT_KEY_1_FILENAME (*tetris.settings.Settings attribute*), 13

MOVING_LEFT_KEY_2_FILENAME (*tetris.settings.Settings attribute*), 13

MOVING_RIGHT_KEY_1_FILENAME (*tetris.settings.Settings attribute*), 13

MOVING_RIGHT_KEY_2_FILENAME (*tetris.settings.Settings attribute*), 13

N

N_OF_LINES_TO_LVL_UP (*tetris.settings.Settings attribute*), 14

next_game() (*tetris.game.Game method*), 7

NEXT_TETROMINO_CELL_HEIGHT (*tetris.settings.Settings attribute*), 13

NEXT_TETROMINO_CELL_WIDTH (*tetris.settings.Settings attribute*), 14

next_tetromino_grid (*tetris.game.Game attribute*), 7

NEXT_TETROMINO_GRID_POS (*tetris.tetrominos.Itetromino attribute*), 15

NEXT_TETROMINO_GRID_POS (*tetris.tetrominos.Jtetromino attribute*), 15

NEXT_TETROMINO_GRID_POS (*tetris.tetrominos.Ltetromino attribute*), 16

NEXT_TETROMINO_GRID_POS (*tetris.tetrominos.Otetromino attribute*), 16

NEXT_TETROMINO_GRID_POS (*tetris.tetrominos.Stetromino attribute*), 16

NEXT_TETROMINO_GRID_POS (*tetris.tetrominos.Tetromino attribute*), 17

NEXT_TETROMINO_GRID_POS (*tetris.tetrominos.Ttetromino attribute*), 19

NEXT_TETROMINO_GRID_POS (*tetris.tetrominos.Ztetromino attribute*), 19

NEXT_TETROMINO_N_OF_COL (*tetris.settings.Settings attribute*), 14

NEXT_TETROMINO_N_OF_ROWS (*tetris.settings.Settings attribute*), 14

NEXT_TETROMINO_TITLE (*tetris.settings.Settings attribute*), 14

next_tetromino_window (*tetris.game.Game attribute*), 7

next_tetromino_window_rect (*tetris.game.Game attribute*), 7

NEXT_WINDOW_X (*tetris.settings.Settings attribute*), 14

NEXT_WINDOW_Y (*tetris.settings.Settings attribute*), 14

O

O (*tetris.settings.Settings attribute*), 14

on_save_handler() (*in module tetris.db.models.user*), 1

Otetromino (*class in tetris.tetrominos*), 16

P

POINTS_FOR_HARD_DROP (*tetris.settings.Settings attribute*), 14

POINTS_FOR_SOFT_DROP (*tetris.settings.Settings attribute*), 14

POINTS_PER_LINES (*tetris.settings.Settings attribute*), 14

pos (*tetris.tetrominos.Tetromino attribute*), 18

pos0() (*tetris.tetrominos.Itetromino method*), 15

pos0() (*tetris.tetrominos.Jtetromino method*), 16

pos0() (*tetris.tetrominos.Ltetromino method*), 16
pos0() (*tetris.tetrominos.Stetromino method*), 17
pos0() (*tetris.tetrominos.Tetromino method*), 18
pos0() (*tetris.tetrominos.Ttetromino method*), 19
pos0() (*tetris.tetrominos.Ztetromino method*), 19
pos1() (*tetris.tetrominos.Itetromino method*), 15
pos1() (*tetris.tetrominos.Jtetromino method*), 16
pos1() (*tetris.tetrominos.Ltetromino method*), 16
pos1() (*tetris.tetrominos.Stetromino method*), 17
pos1() (*tetris.tetrominos.Tetromino method*), 18
pos1() (*tetris.tetrominos.Ttetromino method*), 19
pos1() (*tetris.tetrominos.Ztetromino method*), 19
pos2() (*tetris.tetrominos.Itetromino method*), 15
pos2() (*tetris.tetrominos.Jtetromino method*), 16
pos2() (*tetris.tetrominos.Ltetromino method*), 16
pos2() (*tetris.tetrominos.Stetromino method*), 17
pos2() (*tetris.tetrominos.Tetromino method*), 18
pos2() (*tetris.tetrominos.Ttetromino method*), 19
pos2() (*tetris.tetrominos.Ztetromino method*), 19
pos3() (*tetris.tetrominos.Itetromino method*), 15
pos3() (*tetris.tetrominos.Jtetromino method*), 16
pos3() (*tetris.tetrominos.Ltetromino method*), 16
pos3() (*tetris.tetrominos.Stetromino method*), 17
pos3() (*tetris.tetrominos.Tetromino method*), 18
pos3() (*tetris.tetrominos.Ttetromino method*), 19
pos3() (*tetris.tetrominos.Ztetromino method*), 19
print_grid() (*tetris.game.Game method*), 7
put_on_next_tetromino_window() (*tetris.tetrominos.Tetromino method*), 18

Q

QUIT_CHOICE (*tetris.menu.Menu attribute*), 9

R

random_tetromino() (*tetris.game.Game method*), 7
reset_db() (*in module tetris.db.scripts.migration*), 2
RIGHT_EDGE_POS (*tetris.tests.test_tetrominos.TestTetromino attribute*), 4
rotate_left() (*tetris.tetrominos.Itetromino method*), 15
rotate_left() (*tetris.tetrominos.Otetromino method*), 16
rotate_left() (*tetris.tetrominos.Tetromino method*), 18
ROTATE_LEFT_KEY_FILENAME (*tetris.settings.Settings attribute*), 14
rotate_right() (*tetris.tetrominos.Itetromino method*), 15
rotate_right() (*tetris.tetrominos.Otetromino method*), 16
rotate_right() (*tetris.tetrominos.Tetromino method*), 18
ROTATE_RIGHT_KEY_1_FILENAME (*tetris.settings.Settings attribute*), 14
ROTATE_RIGHT_KEY_2_FILENAME (*tetris.settings.Settings attribute*), 14
Rules (*class in tetris.rules*), 10
RULES_CHOICE (*tetris.menu.Menu attribute*), 9
RULES_LINE_HEIGHT (*tetris.settings.Settings attribute*), 14
RULES_TEXT_Y (*tetris.settings.Settings attribute*), 14
RULES_TITLE (*tetris.settings.Settings attribute*), 14
RULES_TITLE_Y (*tetris.settings.Settings attribute*), 14

S

S (*tetris.settings.Settings attribute*), 14
score (*tetris.game.Game attribute*), 8
SCORE_NEXT_TITLE_Y (*tetris.settings.Settings attribute*), 14
SCORE_TITLE (*tetris.settings.Settings attribute*), 14
score_window (*tetris.game.Game attribute*), 8
score_window_rect (*tetris.game.Game attribute*), 8
SCORE_WINDOW_X (*tetris.settings.Settings attribute*), 14
SCORE_WINDOW_Y (*tetris.settings.Settings attribute*), 14
SCREEN_HEIGHT (*tetris.settings.Settings attribute*), 14
SCREEN_WIDTH (*tetris.settings.Settings attribute*), 14
SECOND_BG_COLOR (*tetris.settings.Settings attribute*), 14
Settings (*class in tetris.settings*), 10
setUp() (*tetris.tests.test_menu.TestMenu method*), 4
setUpClass() (*tetris.tests.test_game.TestGame class method*), 4
setUpClass() (*tetris.tests.test_tetrominos.TestTetromino class method*), 4
space_down (*tetris.game.Game attribute*), 8
spawn() (*tetris.tetrominos.Tetromino method*), 18
SPAWN_POS (*tetris.tetrominos.Itetromino attribute*), 15
SPAWN_POS (*tetris.tetrominos.Jtetromino attribute*), 16
SPAWN_POS (*tetris.tetrominos.Ltetromino attribute*), 16
SPAWN_POS (*tetris.tetrominos.Otetromino attribute*), 16
SPAWN_POS (*tetris.tetrominos.Stetromino attribute*), 16
SPAWN_POS (*tetris.tetrominos.Tetromino attribute*), 17
SPAWN_POS (*tetris.tetrominos.Ttetromino attribute*), 19
SPAWN_POS (*tetris.tetrominos.Ztetromino attribute*), 19
Stetromino (*class in tetris.tetrominos*), 16

T

T (*tetris.settings.Settings attribute*), 14
TAG (*tetris.tetrominos.Itetromino attribute*), 15
TAG (*tetris.tetrominos.Jtetromino attribute*), 16
TAG (*tetris.tetrominos.Ltetromino attribute*), 16
TAG (*tetris.tetrominos.Otetromino attribute*), 16
TAG (*tetris.tetrominos.Stetromino attribute*), 17
TAG (*tetris.tetrominos.Tetromino attribute*), 17
TAG (*tetris.tetrominos.Ttetromino attribute*), 19
TAG (*tetris.tetrominos.Ztetromino attribute*), 19
test_check_down() (*tetris.tests.test_tetrominos.TestTetromino method*), 4

test_check_line() (*tetris.tests.test_game.TestGame method*), 4
test_check_move_left() (*tetris.tests.test_tetrominos.TestTetromino method*), 4
test_check_move_right() (*tetris.tests.test_tetrominos.TestTetromino method*), 5
test_check_tetromino_above_top() (*tetris.tests.test_game.TestGame method*), 4
test_check_touch() (*tetris.tests.test_tetrominos.TestTetromino method*), 5
test_delete_line() (*tetris.tests.test_game.TestGame method*), 4
test_init_properties() (*tetris.tests.test_game.TestGame method*), 4
test_main() (*tetris.tests.test_menu.TestMenu method*), 4
test_random_tetromino() (*tetris.tests.test_game.TestGame method*), 4
TestGame (*class in tetris.tests.test_game*), 2
TestMenu (*class in tetris.tests.test_menu*), 4
TestTetromino (*class in tetris.tests.test_tetrominos*), 4
tetris
  module, 19
tetris.controls
  module, 5
tetris.db
  module, 2
tetris.db.models
  module, 2
tetris.db.models.get_base_dir
  module, 1
tetris.db.models.user
  module, 1
tetris.db.scripts
  module, 2
tetris.db.scripts.get_base_dir
  module, 2
tetris.db.scripts.migration
  module, 2
tetris.db.settings
  module, 2
tetris.game
  module, 5
tetris.leaderboard
  module, 8
tetris.main
  module, 9
tetris.menu
  module, 9
tetris.rules
  module, 10
tetris.settings
  module, 10
tetris.tests
  module, 5
tetris.tests.test_game
  module, 2
tetris.tests.test_menu
  module, 4
tetris.tests.test_tetrominos
  module, 4
tetris.tetrominos
  module, 15
TETRIS_TITLE (*tetris.settings.Settings attribute*), 14
TETRIS_TITLE_Y (*tetris.settings.Settings attribute*), 14
Tetromino (*class in tetris.tetrominos*), 17
TEXT (*tetris.rules.Rules attribute*), 10
Ttetromino (*class in tetris.tetrominos*), 19

U

update_on_grid() (*tetris.tetrominos.Tetromino method*), 19
User (*class in tetris.db.models.user*), 1
user_exists() (*in module tetris.db.models.user*), 1
username (*tetris.db.models.user.User attribute*), 1
username (*tetris.game.Game attribute*), 8

Z

Z (*tetris.settings.Settings attribute*), 15
Ztetromino (*class in tetris.tetrominos*), 19