

SE 3XA3: Module Interface Specification

Random Flag Generator

Team #2, Team Jakriel
Akram Hannoufa, hannoufa
Ganghoon (James) Park, parkg10
Nathaniel Hu, hun4

April 11, 2022

Generated by Doxygen 1.9.3 (plus some manual editing)

1 Namespace Index	1
1.1 Namespace List	1
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	5
3.1 Class List	5
4 File Index	7
4.1 File List	7
5 Namespace Documentation	9
5.1 DecisionUtilities Namespace Reference	9
5.1.1 Function Documentation	9
5.1.1.1 choose_from_list()	9
5.1.1.2 diff()	10
5.1.1.3 hex2rgb()	10
5.1.1.4 map_decision()	11
5.1.1.5 pad_hashcode()	11
5.1.1.6 split_sequence()	12
5.1.2 Variable Documentation	12
5.1.2.1 ASPECT_CONTROL_LEN	12
5.1.2.2 COLOR_QUANTITY	12
5.1.2.3 DEBUG	13
5.1.2.4 HEX_BASE	13
5.1.2.5 HEX_COLOR_LEN	13
5.1.2.6 MAX_DECISION_VALUE	13
5.1.2.7 MINIMUM_HASH_LEN	13
5.2 Display Namespace Reference	13
5.3 FlagAssetsLib Namespace Reference	13
5.3.1 Variable Documentation	14
5.3.1.1 BASE_STRIPE_STYLES	14
5.3.1.2 colours2rgb	14
5.3.1.3 high_res_flag_assets	14
5.3.1.4 low_res_flag_assets	15
5.3.1.5 mid_res_flag_assets	15
5.3.1.6 OVERLAY_STRIPE_STYLES	15
5.3.1.7 STRIPE_NUMBER	16
5.3.1.8 SYMBOL_LOCATION	16
5.3.1.9 SYMBOL_NUMBER	16
5.3.1.10 SYMBOL_TYPES	16
5.4 FlagGenerator Namespace Reference	16
5.5 Gallery Namespace Reference	16

5.6 GUI Namespace Reference	16
5.6.1 Variable Documentation	17
5.6.1.1 app	17
5.7 HashGenerator Namespace Reference	17
5.7.1 Function Documentation	17
5.7.1.1 _get_hash_algo()	17
5.7.1.2 _get_hash_hex()	18
5.7.1.3 hash_generator()	18
5.8 HashToFlag Namespace Reference	18
5.8.1 Function Documentation	19
5.8.1.1 grind_hash_for_base_stripe_style()	19
5.8.1.2 grind_hash_for_colors()	19
5.8.1.3 grind_hash_for_overlay_stripe_style()	20
5.8.1.4 grind_hash_for_stripe_number()	20
5.8.1.5 grind_hash_for_symbol_locations()	20
5.8.1.6 grind_hash_for_symbol_number()	21
5.8.1.7 grind_hash_for_symbol_types()	21
5.9 Help Namespace Reference	22
5.10 JKARReader Namespace Reference	22
5.10.1 Function Documentation	22
5.10.1.1 parse_jka_file()	22
5.10.2 Variable Documentation	23
5.10.2.1 FILLED_PIXEL	23
5.10.2.2 UNFILLED_PIXEL	23
5.11 Settings Namespace Reference	23
6 Class Documentation	25
6.1 Display.FlagDisplay Class Reference	25
6.1.1 Detailed Description	26
6.1.2 Constructor & Destructor Documentation	26
6.1.2.1 __init__()	27
6.1.3 Member Function Documentation	28
6.1.3.1 delete_image()	28
6.1.3.2 display()	28
6.1.3.3 generate()	28
6.1.3.4 get_input_string()	29
6.1.3.5 init_window()	29
6.1.4 Member Data Documentation	29
6.1.4.1 button_back	29
6.1.4.2 button_clear	29
6.1.4.3 button_display	30
6.1.4.4 button_generate	30

6.1.4.5 controller	30
6.1.4.6 delete_image	30
6.1.4.7 display	30
6.1.4.8 FG	30
6.1.4.9 generate	30
6.1.4.10 image_back_button	30
6.1.4.11 image_clear_button	31
6.1.4.12 image_display	31
6.1.4.13 image_display_button	31
6.1.4.14 image_empty	31
6.1.4.15 image_generate_button	31
6.1.4.16 image_text_input	31
6.1.4.17 input_box	31
6.1.4.18 label	31
6.1.4.19 label_display	32
6.1.4.20 label_empty	32
6.1.4.21 photo_back_button	32
6.1.4.22 photo_clear_button	32
6.1.4.23 photo_display	32
6.1.4.24 photo_display_button	32
6.1.4.25 photo_empty	32
6.1.4.26 photo_generate_button	32
6.1.4.27 photo_text_input	33
6.1.4.28 resize_generate_button	33
6.1.4.29 resize_image_back_button	33
6.1.4.30 resize_image_clear_button	33
6.1.4.31 resize_image_display_button	33
6.1.4.32 resize_image_text_input	33
6.1.4.33 resized_image_display	33
6.1.4.34 resized_image_empty	33
6.1.4.35 val	34
6.2 Gallery.FlagGallery Class Reference	34
6.2.1 Detailed Description	34
6.2.2 Constructor & Destructor Documentation	35
6.2.2.1 __init__()	35
6.2.3 Member Function Documentation	35
6.2.3.1 init_window()	35
6.2.3.2 open_gallery()	35
6.2.3.3 open_gallery_win()	35
6.2.4 Member Data Documentation	36
6.2.4.1 button_back	36
6.2.4.2 button_open_gallery	36

6.2.4.3 controller	36
6.2.4.4 image_back_button	36
6.2.4.5 image_open_gallery_button	36
6.2.4.6 photo_back_button	36
6.2.4.7 photo_open_gallery_button	36
6.2.4.8 resize_image_back_button	37
6.2.4.9 resize_image_open_gallery_button	37
6.3 FlagGenerator.FlagGenerator Class Reference	37
6.3.1 Detailed Description	38
6.3.2 Constructor & Destructor Documentation	38
6.3.2.1 __init__()	38
6.3.3 Member Function Documentation	38
6.3.3.1 _save_flag_image()	38
6.3.3.2 _set_flag_base_colour()	39
6.3.3.3 _set_flag_base_stripes()	39
6.3.3.4 _set_flag_dimensions()	39
6.3.3.5 _set_flag_overlay_stripes()	39
6.3.3.6 _set_flag_symbol()	40
6.3.3.7 generate_flag()	40
6.3.3.8 generate_flag_data()	40
6.3.4 Member Data Documentation	41
6.3.4.1 colours	41
6.3.4.2 flag_image	41
6.3.4.3 flag_pixels	41
6.3.4.4 height	41
6.3.4.5 settings	41
6.3.4.6 stripe_info	41
6.3.4.7 symbol_info	41
6.3.4.8 width	42
6.4 Help.Help Class Reference	42
6.4.1 Detailed Description	42
6.4.2 Constructor & Destructor Documentation	43
6.4.2.1 __init__()	43
6.4.3 Member Function Documentation	43
6.4.3.1 init_window()	43
6.4.4 Member Data Documentation	43
6.4.4.1 button_back	43
6.4.4.2 controller	44
6.4.4.3 image	44
6.4.4.4 image_back_button	44
6.4.4.5 label	44
6.4.4.6 photo	44

6.4.4.7 photo_back_button	44
6.4.4.8 resize_image_back_button	44
6.4.4.9 resized_image	45
6.5 GUI.SampleApp Class Reference	45
6.5.1 Detailed Description	45
6.5.2 Constructor & Destructor Documentation	45
6.5.2.1 __init__()	45
6.5.3 Member Function Documentation	46
6.5.3.1 show_frame()	46
6.5.4 Member Data Documentation	46
6.5.4.1 frames	46
6.6 Settings.Settings Class Reference	46
6.6.1 Detailed Description	48
6.6.2 Constructor & Destructor Documentation	48
6.6.2.1 __init__()	48
6.6.3 Member Function Documentation	49
6.6.3.1 delete_image()	49
6.6.3.2 display()	49
6.6.3.3 generate()	49
6.6.3.4 get_input_string()	50
6.6.3.5 init_window()	50
6.6.4 Member Data Documentation	50
6.6.4.1 button_back	50
6.6.4.2 button_clear	50
6.6.4.3 button_display	51
6.6.4.4 button_generate	51
6.6.4.5 controller	51
6.6.4.6 delete_image	51
6.6.4.7 display	51
6.6.4.8 f_s	51
6.6.4.9 FG	51
6.6.4.10 flag_symbols	51
6.6.4.11 generate	52
6.6.4.12 h_t	52
6.6.4.13 hash_types	52
6.6.4.14 image_back_button	52
6.6.4.15 image_clear_button	52
6.6.4.16 image_display	52
6.6.4.17 image_display_button	52
6.6.4.18 image_empty	52
6.6.4.19 image_f_s	53
6.6.4.20 image_generate_button	53

6.6.4.21 image_h_t	53
6.6.4.22 image_p_c	53
6.6.4.23 image_r_t	53
6.6.4.24 image_s_c	53
6.6.4.25 image_text_input	53
6.6.4.26 input_box	53
6.6.4.27 label1	54
6.6.4.28 label2	54
6.6.4.29 label3	54
6.6.4.30 label4	54
6.6.4.31 label5	54
6.6.4.32 label_display	54
6.6.4.33 label_empty	54
6.6.4.34 label_t_i	54
6.6.4.35 p_c	55
6.6.4.36 photo_back_button	55
6.6.4.37 photo_clear_button	55
6.6.4.38 photo_display	55
6.6.4.39 photo_display_button	55
6.6.4.40 photo_empty	55
6.6.4.41 photo_f_s	55
6.6.4.42 photo_generate_button	55
6.6.4.43 photo_h_t	56
6.6.4.44 photo_p_c	56
6.6.4.45 photo_r_t	56
6.6.4.46 photo_s_c	56
6.6.4.47 photo_text_input	56
6.6.4.48 primary_colours	56
6.6.4.49 r_t	56
6.6.4.50 res_types	56
6.6.4.51 resize_image_back_button	57
6.6.4.52 resize_image_clear_button	57
6.6.4.53 resize_image_display_button	57
6.6.4.54 resize_image_f_s	57
6.6.4.55 resize_image_generate_button	57
6.6.4.56 resize_image_h_t	57
6.6.4.57 resize_image_p_c	57
6.6.4.58 resize_image_r_t	57
6.6.4.59 resize_image_s_c	58
6.6.4.60 resize_image_text_input	58
6.6.4.61 resized_image_display	58
6.6.4.62 resized_image_empty	58

6.6.4.63 s_c	58
6.6.4.64 secondary_colours	58
6.7 GUI.StartPage Class Reference	58
6.7.1 Detailed Description	59
6.7.2 Constructor & Destructor Documentation	59
6.7.2.1 __init__()	59
6.7.3 Member Function Documentation	60
6.7.3.1 init_window()	60
6.7.4 Member Data Documentation	60
6.7.4.1 app_logo	60
6.7.4.2 controller	60
6.7.4.3 gallery_button	60
6.7.4.4 gallery_generate	61
6.7.4.5 help_button	61
6.7.4.6 help_generate	61
6.7.4.7 image_app_logo	61
6.7.4.8 image_gallery_button	61
6.7.4.9 image_help_button	61
6.7.4.10 image_settings_button	61
6.7.4.11 image_start_button	61
6.7.4.12 photo_app_logo	62
6.7.4.13 photo_gallery_button	62
6.7.4.14 photo_help_button	62
6.7.4.15 photo_settings_button	62
6.7.4.16 photo_start_button	62
6.7.4.17 resize_image_app_logo	62
6.7.4.18 resize_image_gallery_button	62
6.7.4.19 resize_image_help_button	62
6.7.4.20 resize_image_settings_button	63
6.7.4.21 resize_image_start_button	63
6.7.4.22 settings_button	63
6.7.4.23 settings_generate	63
6.7.4.24 start_button	63
6.7.4.25 start_generate	63
7 File Documentation	65
7.1 DecisionUtilities.py File Reference	65
7.1.1 Detailed Description	66
7.2 Display.py File Reference	66
7.2.1 Detailed Description	66
7.3 FlagAssetsLib.py File Reference	66
7.3.1 Detailed Description	67

7.4 FlagGenerator.py File Reference	67
7.4.1 Detailed Description	68
7.5 Gallery.py File Reference	68
7.5.1 Detailed Description	68
7.6 GUI.py File Reference	68
7.6.1 Detailed Description	69
7.7 HashGenerator.py File Reference	69
7.7.1 Detailed Description	70
7.8 HashToFlag.py File Reference	70
7.8.1 Detailed Description	71
7.9 Help.py File Reference	71
7.9.1 Detailed Description	71
7.10 JKARReader.py File Reference	71
7.10.1 Detailed Description	72
7.11 Settings.py File Reference	72
7.11.1 Detailed Description	72
Index	73

Table 1 Revision History

Date	Version	Notes
March 15, 2022	1.0	Initial Document
March 16, 2022	1.1	Added/updated doxygen commenting to HashGenerator.py, FlagGenerator.py and JKARReader.py modules
March 16, 2022	1.2	Added doxygen commenting to DecisionUtilites.py, FlagAssetsLib.py and HashToFlag.py modules
March 17, 2022	1.3	Added doxygen commenting to GUI.py, Settings.py, Display.py and Help.py modules
March 18, 2022	1.4	Updated doxygen commenting to various modules
April 11, 2022	2.0	Updated Module Interface Specification documentation for Revision 1 submission

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

DecisionUtilities	9
Display	13
FlagAssetsLib	13
FlagGenerator	16
Gallery	16
GUI	16
HashGenerator	17
HashToFlag	18
Help	22
JKAResader	22
Settings	23

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

FlagGenerator.FlagGenerator	37
Frame	
Display.FlagDisplay	25
GUI.StartPage	58
Gallery.FlagGallery	34
Help.Help	42
Settings.Settings	46
tk.Tk	
GUI.SampleApp	45

Chapter 3

Class Index

3.1 Class List

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

Display.FlagDisplay	
FlagDisplay	is a class that allows the user to generate and display the flag 25
Gallery.FlagGallery	
FlagGallery	is a class that opens the gallery so that the user can look at the generated flags 34
FlagGenerator.FlagGenerator	
FlagGenerator	is a class that implements and encapsulates various attributes and methods needed to generate the flag image files 37
Help.Help	
	Shows the help menu to teach the user how to use the software and how it works 42
GUI.SampleApp	
SampleApp	creates the graphical user interface and allows the window to switch between different frames 45
Settings.Settings	
	Creates settings graphical user interface app for user to use the random flag generator and set their settings 46
GUI.StartPage	
StartPage	is a start page that opens once the program starts 58

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

DecisionUtilities.py	
@title DecisionUtilities	65
Display.py	
@title Display	66
FlagAssetsLib.py	
@title FlagAssetsLib	66
FlagGenerator.py	
@title FlagGenerator	67
Gallery.py	
@title Gallery	68
GUI.py	
@title GUI	68
HashGenerator.py	
@title HashGenerator	69
HashToFlag.py	
@title HashToFlag	70
Help.py	
@title Help	71
JKARReader.py	
@title JKARReader	71
Settings.py	
@title Settings	72

Chapter 5

Namespace Documentation

5.1 DecisionUtilities Namespace Reference

Functions

- def [pad_hashcode](#) (hashcode)
Generates a padded hashcode if it is not the minimum required length.
- def [choose_from_list](#) (source_list, index)
Generates a selection from an array.
- def [map_decision](#) (max_digitsum, num_decisions, digitsum)
Maps a number to an index of an array.
- def [split_sequence](#) (seq, length)
Generates a list of shorter tokens from a given input string.
- def [hex2rgb](#) (hexvalue)
Generates a tuple of an RGB colour from a hexadecimal number.
- def [diff](#) (num1, num2)
Calculates the absolute difference of two float values.

Variables

- int [COLOR_QUANTITY](#) = 5
- int [HEX_COLOR_LEN](#) = 6
- int [HEX_BASE](#) = 16
- int [MINIMUM_HASH_LEN](#) = [COLOR_QUANTITY](#) * [HEX_COLOR_LEN](#)
- int [ASPECT_CONTROL_LEN](#) = 6
- int [MAX_DECISION_VALUE](#) = 16777215
- bool [DEBUG](#) = False

5.1.1 Function Documentation

5.1.1.1 [choose_from_list\(\)](#)

```
def DecisionUtilities.choose_from_list (
    source_list,
    index )
```

Generates a selection from an array.

Parameters

<i>source_list</i>	the list to make a selection from
<i>index</i>	the index to take the selection.

Exceptions

<i>ValueError</i>	if index value is negative
-------------------	----------------------------

Returns

choice, the option from the *source_list* that matches the index.

5.1.1.2 diff()

```
def DecisionUtilities.diff (
    num1,
    num2 )
```

Calculates the absolute difference of two float values.

Parameters

<i>num1</i>	first float value
<i>num2</i>	second float value

Returns

The integer value of the absolute difference between float values.

5.1.1.3 hex2rgb()

```
def DecisionUtilities.hex2rgb (
    hexvalue )
```

Generates a tuple of an RGB colour from a hexadecimal number.

hexvalue is length 6, with every 2 characters being part of the RGB tuple

Parameters

<i>hexvalue</i>	hexadecimal value (length=6) to convert to RGB
-----------------	--

Exceptions

<i>TypeError</i>	if hexvalue is not a string
------------------	-----------------------------

Returns

RGB tuple, representing an RGB colour in integer values

5.1.1.4 map_decision()

```
def DecisionUtilities.map_decision (
    max_digitsum,
    num_decisions,
    digitsum )
```

Maps a number to an index of an array.

Parameters

<i>max_digitsum</i>	the maximum possible option
<i>num_decisions</i>	the number of possible decisions
<i>digitsum</i>	the digit to map within possible options

Exceptions

<i>ValueError</i>	if any param is negative
-------------------	--------------------------

Returns

decision, float index of array to get decision from.

5.1.1.5 pad_hashcode()

```
def DecisionUtilities.pad_hashcode (
    hashcode )
```

Generates a padded hashcode if it is not the minimum required length.

Input string gets padded until it is long enough to generate all required flag options.

Parameters

<i>hashcode</i>	a string of the input string's corresponding hashcode value.
-----------------	--

Exceptions

<i>TypeError</i>	if hashcode is not string type
------------------	--------------------------------

Returns

modified hashcode, a padded version of the input hashcode.

5.1.1.6 split_sequence()

```
def DecisionUtilities.split_sequence (
    seq,
    length )
```

Generates a list of shorter tokens from a given input string.

Created strings are of a specified size, n

Parameters

<i>seq</i>	input string to break apart
<i>length</i>	length of generated substrings

Returns

tokens, list of shorter substrings of length n

5.1.2 Variable Documentation

5.1.2.1 ASPECT_CONTROL_LEN

```
int DecisionUtilities.ASPECT_CONTROL_LEN = 6
```

5.1.2.2 COLOR_QUANTITY

```
int DecisionUtilities.COLOR_QUANTITY = 5
```

5.1.2.3 DEBUG

```
bool DecisionUtilities.DEBUG = False
```

5.1.2.4 HEX_BASE

```
int DecisionUtilities.HEX_BASE = 16
```

5.1.2.5 HEX_COLOR_LEN

```
int DecisionUtilities.HEX_COLOR_LEN = 6
```

5.1.2.6 MAX_DECISION_VALUE

```
int DecisionUtilities.MAX_DECISION_VALUE = 16777215
```

5.1.2.7 MINIMUM_HASH_LEN

```
int DecisionUtilities.MINIMUM_HASH_LEN = COLOR_QUANTITY * HEX_COLOR_LEN
```

5.2 Display Namespace Reference

Classes

- class [FlagDisplay](#)
[FlagDisplay](#) is a class that allows the user to generate and display the flag.

5.3 FlagAssetsLib Namespace Reference

Variables

- list [BASE_STRIPE_STYLES](#)
- list [OVERLAY_STRIPE_STYLES](#)
- list [STRIPE_NUMBER](#) = ['ONE', 'TWO', 'THREE', 'SIX', 'TWELVE']
- list [SYMBOL_LOCATION](#) = ['TOP_LEFT', 'CENTER', 'TOP_RIGHT']
- list [SYMBOL_NUMBER](#) = ['ONE', 'TWO']
- list [SYMBOL_TYPES](#) = ['NONE', 'MOON', 'ROUNDEL', 'SWORD']
- dictionary [colours2rgb](#)
- dictionary [low_res_flag_assets](#)
- dictionary [mid_res_flag_assets](#)
- dictionary [high_res_flag_assets](#)

5.3.1 Variable Documentation

5.3.1.1 BASE_STRIPE_STYLES

list FlagAssetsLib.BASE_STRIPE_STYLES

Initial value:

```
1 = ['NONE', 'HORIZONTAL',
2      'VERTICAL', 'CROSS', 'SALTIRE', 'CROSS_SALTIRE']
```

5.3.1.2 colours2rgb

dictionary FlagAssetsLib.colours2rgb

Initial value:

```
1 = {'RED': (255, 0, 0), 'GREEN': (0, 255, 0), 'BLUE': (
2      0, 0, 255), 'YELLOW': (255, 255, 0), 'PURPLE': (127, 0, 255)}
```

5.3.1.3 high_res_flag_assets

dictionary FlagAssetsLib.high_res_flag_assets

Initial value:

```
1 = {"VERTICAL": {"ONE": "jka/high_res/vstripe_1.jka",
2                  "TWO": "jka/high_res/vstripe_2.jka",
3                  "THREE": "jka/high_res/vstripe_3.jka",
4                  "SIX": "jka/high_res/vstripe_6.jka",
5                  "TWELVE": "jka/high_res/vstripe_12.jka",
6                  "ONE_THIN":
7                      "jka/high_res/vstripe_1_thin.jka"},
8      "HORIZONTAL": {"ONE": "jka/high_res/hstripe_1.jka",
9                      "TWO": "jka/high_res/hstripe_2.jka",
10                     "THREE": "jka/high_res/hstripe_3.jka",
11                     "SIX": "jka/high_res/hstripe_6.jka",
12                     "TWELVE": "jka/high_res/hstripe_12.jka",
13                     "ONE_THIN":
14                         "jka/high_res/hstripe_1_thin.jka"},
15      "SALTIRE": "jka/high_res/saltire.jka",
16      "CROSS": "jka/high_res/cross.jka",
17      "CROSS_SALTIRE": "jka/high_res/cross_saltire.jka",
18      "SALTIRE_THIN": "jka/high_res/saltire_thin.jka",
19      "CROSS_THIN": "jka/high_res/cross_thin.jka",
20      "CROSS_SALTIRE_THIN":
21          "jka/high_res/cross_saltire_thin.jka",
22      "MOON": "jka/high_res/moon.jka",
23      "SWORD": "jka/high_res/sword.jka",
24      "ROUNDEL": "jka/high_res/roundel.jka"}
```


5.3.1.4 low_res_flag_assets

dictionary FlagAssetsLib.low_res_flag_assets

Initial value:

```
1 = {"VERTICAL": {"ONE": "jka/low_res/vstripe_1.jka",
2               "TWO": "jka/low_res/vstripe_2.jka",
3               "THREE": "jka/low_res/vstripe_3.jka",
4               "SIX": "jka/low_res/vstripe_6.jka",
5               "TWELVE": "jka/low_res/vstripe_12.jka",
6               "ONE_THIN":
7               "jka/low_res/vstripe_1_thin.jka"},
8       "HORIZONTAL": {"ONE": "jka/low_res/hstripe_1.jka",
9               "TWO": "jka/low_res/hstripe_2.jka",
10              "THREE": "jka/low_res/hstripe_3.jka",
11              "SIX": "jka/low_res/hstripe_6.jka",
12              "TWELVE":
13              "jka/low_res/hstripe_12.jka",
14              "ONE_THIN":
15              "jka/low_res/hstripe_1_thin.jka"},
16       "SALTIRE": "jka/low_res/saltire.jka",
17       "CROSS": "jka/low_res/cross.jka",
18       "CROSS_SALTIRE": "jka/low_res/cross_saltire.jka",
19       "SALTIRE_THIN": "jka/low_res/saltire_thin.jka",
20       "CROSS_THIN": "jka/low_res/cross_thin.jka",
21       "CROSS_SALTIRE_THIN":
22       "jka/low_res/cross_saltire_thin.jka",
23       "MOON": "jka/low_res/moon.jka",
24       "SWORD": "jka/low_res/sword.jka",
25       "ROUNDEL": "jka/low_res/roundel.jka"}
```

5.3.1.5 mid_res_flag_assets

dictionary FlagAssetsLib.mid_res_flag_assets

Initial value:

```
1 = {"VERTICAL": {"ONE": "jka/mid_res/vstripe_1.jka",
2               "TWO": "jka/mid_res/vstripe_2.jka",
3               "THREE": "jka/mid_res/vstripe_3.jka",
4               "SIX": "jka/mid_res/vstripe_6.jka",
5               "TWELVE": "jka/mid_res/vstripe_12.jka",
6               "ONE_THIN":
7               "jka/mid_res/vstripe_1_thin.jka"},
8       "HORIZONTAL": {"ONE": "jka/mid_res/hstripe_1.jka",
9               "TWO": "jka/mid_res/hstripe_2.jka",
10              "THREE": "jka/mid_res/hstripe_3.jka",
11              "SIX": "jka/mid_res/hstripe_6.jka",
12              "TWELVE": "jka/mid_res/hstripe_12.jka",
13              "ONE_THIN":
14              "jka/mid_res/hstripe_1_thin.jka"},
15       "SALTIRE": "jka/mid_res/saltire.jka",
16       "CROSS": "jka/mid_res/cross.jka",
17       "CROSS_SALTIRE": "jka/mid_res/cross_saltire.jka",
18       "SALTIRE_THIN": "jka/mid_res/saltire_thin.jka",
19       "CROSS_THIN": "jka/mid_res/cross_thin.jka",
20       "CROSS_SALTIRE_THIN":
21       "jka/mid_res/cross_saltire_thin.jka",
22       "MOON": "jka/mid_res/moon.jka",
23       "SWORD": "jka/mid_res/sword.jka",
24       "ROUNDEL": "jka/mid_res/roundel.jka"}
```

5.3.1.6 OVERLAY_STRIPE_STYLES

list FlagAssetsLib.OVERLAY_STRIPE_STYLES

Initial value:

```
1 = ['NONE', 'HORIZONTAL_THIN', 'VERTICAL_THIN',
2     'CROSS_THIN', 'SALTIRE_THIN', 'CROSS_SALTIRE_THIN']
```

5.3.1.7 STRIPE_NUMBER

```
list FlagAssetsLib.STRIPE_NUMBER = ['ONE', 'TWO', 'THREE', 'SIX', 'TWELVE']
```

5.3.1.8 SYMBOL_LOCATION

```
list FlagAssetsLib.SYMBOL_LOCATION = ['TOP_LEFT', 'CENTER', 'TOP_RIGHT']
```

5.3.1.9 SYMBOL_NUMBER

```
list FlagAssetsLib.SYMBOL_NUMBER = ['ONE', 'TWO']
```

5.3.1.10 SYMBOL_TYPES

```
list FlagAssetsLib.SYMBOL_TYPES = ['NONE', 'MOON', 'ROUNDEL', 'SWORD']
```

5.4 FlagGenerator Namespace Reference

Classes

- class [FlagGenerator](#)
FlagGenerator is a class that implements and encapsulates various attributes and methods needed to generate the flag image files.

5.5 Gallery Namespace Reference

Classes

- class [FlagGallery](#)
FlagGallery is a class that opens the gallery so that the user can look at the generated flags.

5.6 GUI Namespace Reference

Classes

- class [SampleApp](#)
SampleApp creates the graphical user interface and allows the window to switch between different frames.
- class [StartPage](#)
StartPage is a start page that opens once the program starts.

Variables

- `app` = `SampleApp()`

5.6.1 Variable Documentation

5.6.1.1 `app`

```
GUI.app = SampleApp()
```

5.7 HashGenerator Namespace Reference

Functions

- `def _get_hash_algo` (`hash_type`)
gets the hashing algorithm from the dictionary of available hashing algorithms using the given input hash type string
- `def _get_hash_hex` (`hash_input`, `hash_algo`)
gets the hexadecimal representation of the hashing digest using the given input string and hashing algorithm
- `def hash_generator` (`hash_input`, `hash_type='sha256'`)
generates a hashing digest using the given input string and hashing algorithm

5.7.1 Function Documentation

5.7.1.1 `_get_hash_algo()`

```
def HashGenerator._get_hash_algo (
    hash_type ) [private]
```

gets the hashing algorithm from the dictionary of available hashing algorithms using the given input hash type string

the default hashing algorithm, SHA-256, is used if the input hash type is not in the dictionary of available hashing algorithms

Parameters

<i>hash_type</i>	a string representing the selected hashing algorithm
------------------	--

Returns

selected hashing algorithm if found; otherwise SHA-256 hashing algorithm returned

5.7.1.2 `_get_hash_hex()`

```
def HashGenerator._get_hash_hex (
    hash_input,
    hash_algo ) [private]
```

gets the hexadecimal representation of the hashing digest using the given input string and hashing algorithm

the byte encoding will be specified per the Python version used

Parameters

<i>hash_input</i>	a string that will be run through the given hashing algorithm to get a hexadecimal hashing digest
<i>hash_algo</i>	a hashing algorithm that will be used to turn the input string into a hexadecimal hashing digest

Returns

hexadecimal hashing digest obtained from the given input string using the given hashing algorithm

5.7.1.3 `hash_generator()`

```
def HashGenerator.hash_generator (
    hash_input,
    hash_type = 'sha256' )
```

generates a hashing digest using the given input string and hashing algorithm

the hashing algorithm SHA-256 will be used by default if none is specified

Parameters

<i>hash_input</i>	a string that will be run through the given hashing algorithm to get a hexadecimal hashing digest
<i>hash_type</i>	a string representing the selected hashing algorithm

Returns

hexadecimal hashing digest obtained from the given input string using the given hashing algorithm

5.8 HashToFlag Namespace Reference

Functions

- def [grind_hash_for_colors](#) (hashcode)
Generates the array of colours to be used in the flag generation.
- def [grind_hash_for_base_stripe_style](#) (hashcode)
Generates the base stripe style to be used in flag generation.

- def [grind_hash_for_overlay_stripe_style](#) (hashcode)
Generates the overlay stripe style to be used in flag generation.
- def [grind_hash_for_stripe_number](#) (hashcode)
Generates the number of stripes to be used in flag generation.
- def [grind_hash_for_symbol_locations](#) (hashcode)
Generates the symbol location to be used in flag generation.
- def [grind_hash_for_symbol_number](#) (hashcode)
Generates the number of symbols to be used in flag generation.
- def [grind_hash_for_symbol_types](#) (hashcode)
Generates the symbol type to be used in flag generation.

5.8.1 Function Documentation

5.8.1.1 [grind_hash_for_base_stripe_style\(\)](#)

```
def HashToFlag.grind_hash_for_base_stripe_style (  
    hashcode )
```

Generates the base stripe style to be used in flag generation.

Uses the first 6 characters of a hashcode to map to an array index ,ie. the option to use for the desired aspect.

Parameters

<i>hashcode</i>	a string of the input string's corresponding hashcode value.
-----------------	--

Returns

A base stripe style option.

5.8.1.2 [grind_hash_for_colors\(\)](#)

```
def HashToFlag.grind_hash_for_colors (  
    hashcode )
```

Generates the array of colours to be used in the flag generation.

Hex values of the hashcode are converted to an RGB value for colour.

Parameters

<i>hashcode</i>	a string of the input string's corresponding hashcode value.
-----------------	--

Returns

colors, an array of RGB values to be used by [FlagGenerator](#).

5.8.1.3 grind_hash_for_overlay_stripe_style()

```
def HashToFlag.grind_hash_for_overlay_stripe_style (  
    hashcode )
```

Generates the overlay stripe style to be used in flag generation.

Uses the first 6 characters of a hashcode to map to an array index ,ie.the option to use for the desired aspect.

Parameters

<i>hashcode</i>	a string of the input string's corresponding hashcode value.
-----------------	--

Returns

An overlay stripe style option.

5.8.1.4 grind_hash_for_stripe_number()

```
def HashToFlag.grind_hash_for_stripe_number (  
    hashcode )
```

Generates the number of stripes to be used in flag generation.

Uses the second 6 characters of a hashcode to map to an array index ,ie.the option to use for the desired aspect.

Parameters

<i>hashcode</i>	a string of the input string's corresponding hashcode value.
-----------------	--

Returns

A stripe number option.

5.8.1.5 grind_hash_for_symbol_locations()

```
def HashToFlag.grind_hash_for_symbol_locations (  
    hashcode )
```

Generates the symbol location to be used in flag generation.

Uses the third 6 characters of a hashcode to map to an array index ,ie.the option to use for the desired aspect.

Parameters

<i>hashcode</i>	a string of the input string's corresponding hashcode value.
-----------------	--

Returns

A symbol location option.

5.8.1.6 grind_hash_for_symbol_number()

```
def HashToFlag.grind_hash_for_symbol_number (
    hashcode )
```

Generates the number of symbols to be used in flag generation.

Uses the fourth 6 characters of a hashcode to map to an array index ,ie.the option to use for the desired aspect.

Parameters

<i>hashcode</i>	a string of the input string's corresponding hashcode value.
-----------------	--

Returns

A symbol number option.

5.8.1.7 grind_hash_for_symbol_types()

```
def HashToFlag.grind_hash_for_symbol_types (
    hashcode )
```

Generates the symbol type to be used in flag generation.

Uses the fifth 6 characters of a hashcode to map to an array index, ie.the option to use for the desired aspect.

Parameters

<i>hashcode</i>	a string of the input string's corresponding hashcode value.
-----------------	--

Returns

A symbol type option.

5.9 Help Namespace Reference

Classes

- class [Help](#)

Shows the help menu to teach the user how to use the software and how it works.

5.10 JKAResource Namespace Reference

Functions

- def [parse_jka_file](#) (filename)
parses the input flag asset (.jka) file data into a pixel map

Variables

- string [FILLED_PIXEL](#) = "#"
- string [UNFILLED_PIXEL](#) = "."

5.10.1 Function Documentation

5.10.1.1 [parse_jka_file\(\)](#)

```
def JKAResource.parse_jka_file (
    filename )
```

parses the input flag asset (.jka) file data into a pixel map

parses the file data by pixel and adds filled pixels to the pixel map

Parameters

<i>filename</i>	a string representing the name of the flag asset (.jka) file that contains the flag asset pixel map data
-----------------	--

Returns

a list containing the (x, y) coordinates of all filled pixels for the given flag asset

5.10.2 Variable Documentation

5.10.2.1 FILLED_PIXEL

```
string JKARader.FILLED_PIXEL = "#"
```

5.10.2.2 UNFILLED_PIXEL

```
string JKARader.UNFILLED_PIXEL = "."
```

5.11 Settings Namespace Reference

Classes

- class [Settings](#)

Creates settings graphical user interface app for user to use the random flag generator and set their settings.

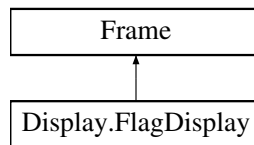
Chapter 6

Class Documentation

6.1 Display.FlagDisplay Class Reference

[FlagDisplay](#) is a class that allows the user to generate and display the flag.

Inheritance diagram for Display.FlagDisplay:



Public Member Functions

- `def __init__ (self, parent, controller)`
Creates the next page after clicking the start button from the [GUI](#) main page.
- `def init_window (self)`
Places the different widgets onto the second page after clicking on the start button on the [GUI](#) main page.
- `def get_input_string (self)`
Gets the input string from the input text box.
- `def generate (self, event)`
Generates the flag from the input string.
- `def display (self, event)`
Displays the generated flag.
- `def delete_image (self, event)`
Clears the displayed flag.

Public Attributes

- [controller](#)
- [FG](#)
- [image_text_input](#)
- [image_generate_button](#)
- [image_display_button](#)
- [image_back_button](#)
- [image_clear_button](#)
- [resize_image_text_input](#)
- [resize_generate_button](#)
- [resize_image_display_button](#)
- [resize_image_back_button](#)
- [resize_image_clear_button](#)
- [photo_text_input](#)
- [photo_generate_button](#)
- [photo_display_button](#)
- [photo_back_button](#)
- [photo_clear_button](#)
- [label](#)
- [button_generate](#)
- [generate](#)
- [val](#)
- [button_display](#)
- [display](#)
- [button_back](#)
- [button_clear](#)
- [delete_image](#)
- [input_box](#)
- [image_display](#)
- [resized_image_display](#)
- [photo_display](#)
- [label_display](#)
- [image_empty](#)
- [resized_image_empty](#)
- [photo_empty](#)
- [label_empty](#)

6.1.1 Detailed Description

[FlagDisplay](#) is a class that allows the user to generate and display the flag.

the [FlagDisplay](#) contains the text input box, generate button, back button. display button, and a clear button for the user to use.

6.1.2 Constructor & Destructor Documentation

6.1.2.1 `__init__()`

```
def Display.FlagDisplay.__init__ (
    self,
    parent,
    controller )
```

Creates the next page after clicking the start button from the [GUI](#) main page.

The [GUI](#) second page will have the text input box, generate button, display button, and a back button.

Parameters

<i>self</i>	Current object, common first parameter for any method of a class.
<i>parent</i>	A widget that acts as the parent of self, current object. All widgets in tkinter except the root window require a parent
<i>controller</i>	Other objects that are designed to act as a shared point, allowing several pages of widgets to interact. It decouples the different pages, making them independent. The controller decides what page will be visible.

6.1.3 Member Function Documentation**6.1.3.1 delete_image()**

```
def Display.FlagDisplay.delete_image (
    self,
    event )
```

Clears the displayed flag.

The generated flag is cleared on the screen with this function

Parameters

<i>event</i>	When clear button is clicked, it will call this function to clear the flag being displayed.
--------------	---

6.1.3.2 display()

```
def Display.FlagDisplay.display (
    self,
    event )
```

Displays the generated flag.

The generated flag is displayed on the screen with this function

Parameters

<i>event</i>	When display button is clicked, it will call this function to display the flag.
--------------	---

6.1.3.3 generate()

```
def Display.FlagDisplay.generate (
```

```
self,
event )
```

Generates the flag from the input string.

The flag is generated with this function

Parameters

<i>event</i>	When generate button is clicked, it will call this function to generate the flag.
--------------	---

6.1.3.4 get_input_string()

```
def Display.FlagDisplay.get_input_string (
    self )
```

Gets the input string from the input text box.

The input string is retrieved with this function

6.1.3.5 init_window()

```
def Display.FlagDisplay.init_window (
    self )
```

Places the different widgets onto the second page after clicking on the start button on the [GUI](#) main page.

The second page will include the text input input box, generate button, display button, clear button, and a back button.

6.1.4 Member Data Documentation

6.1.4.1 button_back

```
Display.FlagDisplay.button_back
```

6.1.4.2 button_clear

```
Display.FlagDisplay.button_clear
```

6.1.4.3 button_display

`Display.FlagDisplay.button_display`

6.1.4.4 button_generate

`Display.FlagDisplay.button_generate`

6.1.4.5 controller

`Display.FlagDisplay.controller`

6.1.4.6 delete_image

`Display.FlagDisplay.delete_image`

6.1.4.7 display

`Display.FlagDisplay.display`

6.1.4.8 FG

`Display.FlagDisplay.FG`

6.1.4.9 generate

`Display.FlagDisplay.generate`

6.1.4.10 image_back_button

`Display.FlagDisplay.image_back_button`

6.1.4.11 image_clear_button

Display.FlagDisplay.image_clear_button

6.1.4.12 image_display

Display.FlagDisplay.image_display

6.1.4.13 image_display_button

Display.FlagDisplay.image_display_button

6.1.4.14 image_empty

Display.FlagDisplay.image_empty

6.1.4.15 image_generate_button

Display.FlagDisplay.image_generate_button

6.1.4.16 image_text_input

Display.FlagDisplay.image_text_input

6.1.4.17 input_box

Display.FlagDisplay.input_box

6.1.4.18 label

Display.FlagDisplay.label

6.1.4.19 label_display

`Display.FlagDisplay.label_display`

6.1.4.20 label_empty

`Display.FlagDisplay.label_empty`

6.1.4.21 photo_back_button

`Display.FlagDisplay.photo_back_button`

6.1.4.22 photo_clear_button

`Display.FlagDisplay.photo_clear_button`

6.1.4.23 photo_display

`Display.FlagDisplay.photo_display`

6.1.4.24 photo_display_button

`Display.FlagDisplay.photo_display_button`

6.1.4.25 photo_empty

`Display.FlagDisplay.photo_empty`

6.1.4.26 photo_generate_button

`Display.FlagDisplay.photo_generate_button`

6.1.4.27 photo_text_input

`Display.FlagDisplay.photo_text_input`

6.1.4.28 resize_generate_button

`Display.FlagDisplay.resize_generate_button`

6.1.4.29 resize_image_back_button

`Display.FlagDisplay.resize_image_back_button`

6.1.4.30 resize_image_clear_button

`Display.FlagDisplay.resize_image_clear_button`

6.1.4.31 resize_image_display_button

`Display.FlagDisplay.resize_image_display_button`

6.1.4.32 resize_image_text_input

`Display.FlagDisplay.resize_image_text_input`

6.1.4.33 resized_image_display

`Display.FlagDisplay.resized_image_display`

6.1.4.34 resized_image_empty

`Display.FlagDisplay.resized_image_empty`

6.1.4.35 val

`Display.FlagDisplay.val`

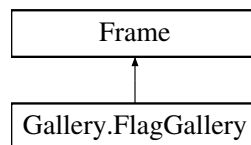
The documentation for this class was generated from the following file:

- [Display.py](#)

6.2 Gallery.FlagGallery Class Reference

[FlagGallery](#) is a class that opens the gallery so that the user can look at the generated flags.

Inheritance diagram for `Gallery.FlagGallery`:



Public Member Functions

- `def __init__ (self, parent, controller)`
Constructor for [FlagGallery](#) GUI object.
- `def init_window (self)`
method for placing buttons in [FlagGallery](#).
- `def open_gallery (self)`
method for open the gallery after clicking the open gallery button.
- `def open_gallery_win (self)`
method for constructing the gallery.

Public Attributes

- [controller](#)
- [image_back_button](#)
- [resize_image_back_button](#)
- [photo_back_button](#)
- [button_back](#)
- [image_open_gallery_button](#)
- [resize_image_open_gallery_button](#)
- [photo_open_gallery_button](#)
- [button_open_gallery](#)

6.2.1 Detailed Description

[FlagGallery](#) is a class that opens the gallery so that the user can look at the generated flags.

the [FlagGallery](#) contains the open gallery button and a back button for the user to control.

6.2.2 Constructor & Destructor Documentation

6.2.2.1 `__init__()`

```
def Gallery.FlagGallery.__init__ (
    self,
    parent,
    controller )
```

Constructor for [FlagGallery GUI](#) object.

Creates object for [GUI](#)

Parameters

<i>*args</i>	allows the function to accept an arbitrary number of arguments
<i>**kwargs</i>	allows the function to accept an arbitrary number of keyword arguments.

6.2.3 Member Function Documentation

6.2.3.1 `init_window()`

```
def Gallery.FlagGallery.init_window (
    self )
```

method for placing buttons in [FlagGallery](#).

the buttons will be placed onto the [FlagGallery GUI](#).

6.2.3.2 `open_gallery()`

```
def Gallery.FlagGallery.open_gallery (
    self )
```

method for open the gallery after clicking the open gallery button.

it will close the [FlagGallery](#) frame and open the gallery window.

6.2.3.3 `open_gallery_win()`

```
def Gallery.FlagGallery.open_gallery_win (
    self )
```

method for constructing the gallery.

it will showcase all the generated flags by clicking buttons for next and previous flag.

6.2.4 Member Data Documentation

6.2.4.1 button_back

`Gallery.FlagGallery.button_back`

6.2.4.2 button_open_gallery

`Gallery.FlagGallery.button_open_gallery`

6.2.4.3 controller

`Gallery.FlagGallery.controller`

6.2.4.4 image_back_button

`Gallery.FlagGallery.image_back_button`

6.2.4.5 image_open_gallery_button

`Gallery.FlagGallery.image_open_gallery_button`

6.2.4.6 photo_back_button

`Gallery.FlagGallery.photo_back_button`

6.2.4.7 photo_open_gallery_button

`Gallery.FlagGallery.photo_open_gallery_button`

6.2.4.8 `resize_image_back_button`

`Gallery.FlagGallery.resize_image_back_button`

6.2.4.9 `resize_image_open_gallery_button`

`Gallery.FlagGallery.resize_image_open_gallery_button`

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

- [Gallery.py](#)

6.3 FlagGenerator.FlagGenerator Class Reference

[FlagGenerator](#) is a class that implements and encapsulates various attributes and methods needed to generate the flag image files.

Public Member Functions

- `def __init__ (self)`
constructor method for [FlagGenerator](#)
- `def generate_flag (self, hash_input, hash_type="SHA256", settings={"RESOLUTION":"HIGH", "BASE_COLOUR":"RANDOM", "SYMBOL_COLOUR":"RANDOM", "SYMBOL_TYPE":"RANDOM"})`
generates the flag image file using the given hash input string and input hash type string
- `def generate_flag_data (self, hash_input, hash_type)`
generates the flag data using the given hash input string and input hash type string

Public Attributes

- [settings](#)
- [colours](#)
- [stripe_info](#)
- [symbol_info](#)
- [width](#)
- [height](#)
- [flag_image](#)
- [flag_pixels](#)

Private Member Functions

- `def _set_flag_dimensions (self)`
sets dimensions for the generated flag image
- `def _set_flag_base_colour (self)`
sets base colour for the generated flag image
- `def _set_flag_base_stripes (self)`
sets base stripes colour and style for the generated flag image
- `def _set_flag_overlay_stripes (self)`
sets overlay stripes colour and style for the generated flag image
- `def _set_flag_symbol (self)`
sets symbol colour and type for the generated flag image
- `def _save_flag_image (self, hash_input)`
saves generated flag image

6.3.1 Detailed Description

`FlagGenerator` is a class that implements and encapsulates various attributes and methods needed to generate the flag image files.

the `FlagGenerator` contains the user settings and flag colours, stripe info, symbol info, dimensions, image and pixel data

6.3.2 Constructor & Destructor Documentation

6.3.2.1 `__init__()`

```
def FlagGenerator.FlagGenerator.__init__ (
    self )
```

constructor method for `FlagGenerator`

the attributes will be used to temporarily store the data needed to generate each flag image; overwritten with each flag subsequently generated

6.3.3 Member Function Documentation

6.3.3.1 `_save_flag_image()`

```
def FlagGenerator.FlagGenerator._save_flag_image (
    self,
    hash_input ) [private]
```

saves generated flag image

saves generated flag image as png image file to flag gallery directory, using given `hash_input` to name the image file

Parameters

<i>hash_input</i>	a string that will be used to name the generated flag image file
-------------------	--

6.3.3.2 `_set_flag_base_colour()`

```
def FlagGenerator.FlagGenerator._set_flag_base_colour (
    self ) [private]
```

sets base colour for the generated flag image

sets flag image base colour based on set colour choice (or if set to RANDOM, then sets to randomly generated colour)

6.3.3.3 `_set_flag_base_stripes()`

```
def FlagGenerator.FlagGenerator._set_flag_base_stripes (
    self ) [private]
```

sets base stripes colour and style for the generated flag image

sets flag image base stripes colour to randomly generated colour and base stripes style to randomly chosen stripes style

6.3.3.4 `_set_flag_dimensions()`

```
def FlagGenerator.FlagGenerator._set_flag_dimensions (
    self ) [private]
```

sets dimensions for the generated flag image

sets flag image width and height (in pixels) based upon the set resolution

6.3.3.5 `_set_flag_overlay_stripes()`

```
def FlagGenerator.FlagGenerator._set_flag_overlay_stripes (
    self ) [private]
```

sets overlay stripes colour and style for the generated flag image

sets flag image overlay stripes colour to randomly generated colour and overlay stripes style to randomly chosen stripes style

6.3.3.6 `_set_flag_symbol()`

```
def FlagGenerator.FlagGenerator._set_flag_symbol (
    self ) [private]
```

sets symbol colour and type for the generated flag image

sets flag image symbol type based on set symbol type choice (or if set to RANDOM, then sets to randomly selected symbol type); sets flag image symbol colour based on set symbol colour choice (or if set to RANDOM, then sets to randomly generated colour)

6.3.3.7 `generate_flag()`

```
def FlagGenerator.FlagGenerator.generate_flag (
    self,
    hash_input,
    hash_type = "SHA256",
    settings = {"RESOLUTION": "HIGH", "BASE_COLOUR": "RANDOM",
    _COLOUR": "RANDOM", "SYMBOL_TYPE": "RANDOM"} )
```

"SYME

generates the flag image file using the given hash input string and input hash type string

uses functions to get the flag data from the hash input string and input hash type string and stacks the flag asset layers to generate the final flag image saved to the gallery

Parameters

<i>hash_input</i>	a string that will be run through the given hashing algorithm to get a hexadecimal hashing digest
<i>hash_type</i>	a string representing the selected hashing algorithm; default hashing algorithm is sha256
<i>settings</i>	a dictionary representing the user selected settings; default resolution is high default colours and symbol type are randomly generated, i.e. taken from generated flag data

6.3.3.8 `generate_flag_data()`

```
def FlagGenerator.FlagGenerator.generate_flag_data (
    self,
    hash_input,
    hash_type )
```

generates the flag data using the given hash input string and input hash type string

uses external functions to generate the flag data from the hash input string and input hash type string

Parameters

<i>hash_input</i>	a string that will be run through the given hashing algorithm to get a hexadecimal hashing digest
<i>hash_type</i>	a string representing the selected hashing algorithm

6.3.4 Member Data Documentation

6.3.4.1 colours

FlagGenerator.FlagGenerator.colours

6.3.4.2 flag_image

FlagGenerator.FlagGenerator.flag_image

6.3.4.3 flag_pixels

FlagGenerator.FlagGenerator.flag_pixels

6.3.4.4 height

FlagGenerator.FlagGenerator.height

6.3.4.5 settings

FlagGenerator.FlagGenerator.settings

6.3.4.6 stripe_info

FlagGenerator.FlagGenerator.stripe_info

6.3.4.7 symbol_info

FlagGenerator.FlagGenerator.symbol_info

6.3.4.8 width

FlagGenerator.FlagGenerator.width

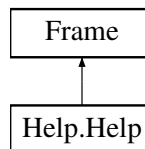
The documentation for this class was generated from the following file:

- [FlagGenerator.py](#)

6.4 Help.Help Class Reference

Shows the help menu to teach the user how to use the software and how it works.

Inheritance diagram for Help.Help:



Public Member Functions

- `def __init__(self, parent, controller)`
Creates the help page after clicking the help button from the [GUI](#) main page.
- `def init_window(self)`
Places the different widgets onto the help page after clicking on the help button on the [GUI](#) main page.

Public Attributes

- [controller](#)
- [image_back_button](#)
- [resize_image_back_button](#)
- [photo_back_button](#)
- [button_back](#)
- [image](#)
- [resized_image](#)
- [photo](#)
- [label](#)

6.4.1 Detailed Description

Shows the help menu to teach the user how to use the software and how it works.

The help menu will have instructions and a button to go back to the main [GUI](#) screen.

6.4.2 Constructor & Destructor Documentation

6.4.2.1 `__init__()`

```
def Help.Help.__init__ (
    self,
    parent,
    controller )
```

Creates the help page after clicking the help button from the [GUI](#) main page.

The help page will have the instructions and a back button.

Parameters

<i>self</i>	Current object, common first parameter for any method of a class.
<i>parent</i>	A widget that acts as the parent of self, current object. All widgets in tkinter except the root window require a parent
<i>controller</i>	Other objects that are designed to act as a shared point, allowing several pages of widgets to interact. It decouples the different pages, making them independent. The controller decides what page will be visible.

6.4.3 Member Function Documentation

6.4.3.1 `init_window()`

```
def Help.Help.init_window (
    self )
```

Places the different widgets onto the help page after clicking on the help button on the [GUI](#) main page.

The help page will include the instructions and a back button.

6.4.4 Member Data Documentation

6.4.4.1 `button_back`

```
Help.Help.button_back
```

6.4.4.2 controller

`Help.Help.controller`

6.4.4.3 image

`Help.Help.image`

6.4.4.4 image_back_button

`Help.Help.image_back_button`

6.4.4.5 label

`Help.Help.label`

6.4.4.6 photo

`Help.Help.photo`

6.4.4.7 photo_back_button

`Help.Help.photo_back_button`

6.4.4.8 resize_image_back_button

`Help.Help.resize_image_back_button`

6.4.4.9 resized_image

Help.Help.resized_image

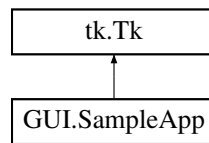
The documentation for this class was generated from the following file:

- [Help.py](#)

6.5 GUI.SampleApp Class Reference

[SampleApp](#) creates the graphical user interface and allows the window to switch between different frames.

Inheritance diagram for GUI.SampleApp:



Public Member Functions

- `def __init__ (self, *args, **kwargs)`
Constructor for new app [GUI](#) object.
- `def show_frame (self, page_name)`
Shows the frame and switches between frames.

Public Attributes

- [frames](#)

6.5.1 Detailed Description

[SampleApp](#) creates the graphical user interface and allows the window to switch between different frames.

the [SampleApp](#) contains the the start page, display, gallery, settings, and help frames.

6.5.2 Constructor & Destructor Documentation

6.5.2.1 __init__()

```
def GUI.SampleApp.__init__ (
    self,
    * args,
    ** kwargs )
```

Constructor for new app [GUI](#) object.

Creates object for [GUI](#)

Parameters

<i>*args</i>	allows the function to accept an arbitrary number of arguments
<i>**kwargs</i>	allows the function to accept an arbitrary number of keyword arguments.

6.5.3 Member Function Documentation

6.5.3.1 show_frame()

```
def GUI.SampleApp.show_frame (
    self,
    page_name )
```

Shows the frame and switches between frames.

The frame is generated and switched.

Parameters

<i>page_name</i>	Used to switch between frames in tkinter. The frame it will show on the window.
------------------	---

6.5.4 Member Data Documentation

6.5.4.1 frames

```
GUI.SampleApp.frames
```

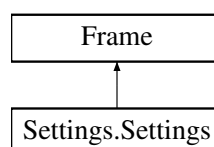
The documentation for this class was generated from the following file:

- [GUI.py](#)

6.6 Settings.Settings Class Reference

Creates settings graphical user interface app for user to use the random flag generator and set their settings.

Inheritance diagram for Settings.Settings:



Public Member Functions

- def `__init__` (self, parent, [controller](#))
Creates the settings page after clicking the settings button from the [GUI](#) main page.
- def `init_window` (self)
Places the different widgets onto the settings page after clicking on the settings button on the [GUI](#) main page.
- def `get_input_string` (self)
Gets the input string from the input text box.
- def `generate` (self, event)
Generates the flag from the input string.
- def `display` (self, event)
Displays the generated flag.
- def `delete_image` (self, event)
Clears the displayed flag.

Public Attributes

- [controller](#)
- [FG](#)
- [image_text_input](#)
- [resize_image_text_input](#)
- [photo_text_input](#)
- [label_t_i](#)
- [input_box](#)
- [image_back_button](#)
- [resize_image_back_button](#)
- [photo_back_button](#)
- [button_back](#)
- [image_generate_button](#)
- [resize_image_generate_button](#)
- [photo_generate_button](#)
- [button_generate](#)
- [generate](#)
- [image_display_button](#)
- [resize_image_display_button](#)
- [photo_display_button](#)
- [button_display](#)
- [display](#)
- [image_clear_button](#)
- [resize_image_clear_button](#)
- [photo_clear_button](#)
- [button_clear](#)
- [delete_image](#)
- [image_p_c](#)
- [resize_image_p_c](#)
- [photo_p_c](#)
- [label1](#)
- [image_s_c](#)
- [resize_image_s_c](#)
- [photo_s_c](#)
- [label2](#)
- [image_f_s](#)
- [resize_image_f_s](#)

- [photo_f_s](#)
- [label3](#)
- [image_h_t](#)
- [resize_image_h_t](#)
- [photo_h_t](#)
- [label4](#)
- [image_r_t](#)
- [resize_image_r_t](#)
- [photo_r_t](#)
- [label5](#)
- [primary_colours](#)
- [secondary_colours](#)
- [flag_symbols](#)
- [hash_types](#)
- [res_types](#)
- [p_c](#)
- [s_c](#)
- [f_s](#)
- [h_t](#)
- [r_t](#)
- [image_display](#)
- [resized_image_display](#)
- [photo_display](#)
- [label_display](#)
- [image_empty](#)
- [resized_image_empty](#)
- [photo_empty](#)
- [label_empty](#)

6.6.1 Detailed Description

Creates settings graphical user interface app for user to use the random flag generator and set their settings.

[Settings](#) graphical user interface for user to change the settings.

6.6.2 Constructor & Destructor Documentation

6.6.2.1 `__init__()`

```
def Settings.Settings.__init__ (
    self,
    parent,
    controller )
```

Creates the settings page after clicking the settings button from the [GUI](#) main page.

The settings page will have options for the user to change the flag setting and a back button.

Parameters

<i>self</i>	Current object, common first parameter for any method of a class.
<i>parent</i>	A widget that acts as the parent of self, current object. All widgets in tkinter except the root window require a parent
<i>controller</i>	Other objects that are designed to act as a shared point, allowing several pages of widgets to interact. It decouples the different pages, making them independent. The controller decides what page will be visible.

6.6.3 Member Function Documentation

6.6.3.1 delete_image()

```
def Settings.Settings.delete_image (
    self,
    event )
```

Clears the displayed flag.

The generated flag is cleared on the screen with this function

Parameters

<i>event</i>	When clear button is clicked, it will call this function to clear the flag being displayed.
--------------	---

6.6.3.2 display()

```
def Settings.Settings.display (
    self,
    event )
```

Displays the generated flag.

The generated flag is displayed on the screen with this function

Parameters

<i>event</i>	When display button is clicked, it will call this function to display the flag.
--------------	---

6.6.3.3 generate()

```
def Settings.Settings.generate (
```

```
self,  
event )
```

Generates the flag from the input string.

The flag is generated with this function

Parameters

<i>event</i>	When generate button is clicked, it will call this function to generate the flag.
--------------	---

6.6.3.4 get_input_string()

```
def Settings.Settings.get_input_string (  
    self )
```

Gets the input string from the input text box.

The input string is retrieved with this function

6.6.3.5 init_window()

```
def Settings.Settings.init_window (  
    self )
```

Places the different widgets onto the settings page after clicking on the settings button on the [GUI](#) main page.

The settings page will include the options to change the flag settings, such as colours, symbols, resolutions, and a back button.

6.6.4 Member Data Documentation

6.6.4.1 button_back

```
Settings.Settings.button_back
```

6.6.4.2 button_clear

```
Settings.Settings.button_clear
```

6.6.4.3 button_display

Settings.Settings.button_display

6.6.4.4 button_generate

Settings.Settings.button_generate

6.6.4.5 controller

Settings.Settings.controller

6.6.4.6 delete_image

Settings.Settings.delete_image

6.6.4.7 display

Settings.Settings.display

6.6.4.8 f_s

Settings.Settings.f_s

6.6.4.9 FG

Settings.Settings.FG

6.6.4.10 flag_symbols

Settings.Settings.flag_symbols

6.6.4.11 generate

`Settings.Settings.generate`

6.6.4.12 h_t

`Settings.Settings.h_t`

6.6.4.13 hash_types

`Settings.Settings.hash_types`

6.6.4.14 image_back_button

`Settings.Settings.image_back_button`

6.6.4.15 image_clear_button

`Settings.Settings.image_clear_button`

6.6.4.16 image_display

`Settings.Settings.image_display`

6.6.4.17 image_display_button

`Settings.Settings.image_display_button`

6.6.4.18 image_empty

`Settings.Settings.image_empty`

6.6.4.19 image_f_s

Settings.Settings.image_f_s

6.6.4.20 image_generate_button

Settings.Settings.image_generate_button

6.6.4.21 image_h_t

Settings.Settings.image_h_t

6.6.4.22 image_p_c

Settings.Settings.image_p_c

6.6.4.23 image_r_t

Settings.Settings.image_r_t

6.6.4.24 image_s_c

Settings.Settings.image_s_c

6.6.4.25 image_text_input

Settings.Settings.image_text_input

6.6.4.26 input_box

Settings.Settings.input_box

6.6.4.27 label1

`Settings.Settings.label1`

6.6.4.28 label2

`Settings.Settings.label2`

6.6.4.29 label3

`Settings.Settings.label3`

6.6.4.30 label4

`Settings.Settings.label4`

6.6.4.31 label5

`Settings.Settings.label5`

6.6.4.32 label_display

`Settings.Settings.label_display`

6.6.4.33 label_empty

`Settings.Settings.label_empty`

6.6.4.34 label_t_i

`Settings.Settings.label_t_i`

6.6.4.35 p_c

`Settings.Settings.p_c`

6.6.4.36 photo_back_button

`Settings.Settings.photo_back_button`

6.6.4.37 photo_clear_button

`Settings.Settings.photo_clear_button`

6.6.4.38 photo_display

`Settings.Settings.photo_display`

6.6.4.39 photo_display_button

`Settings.Settings.photo_display_button`

6.6.4.40 photo_empty

`Settings.Settings.photo_empty`

6.6.4.41 photo_f_s

`Settings.Settings.photo_f_s`

6.6.4.42 photo_generate_button

`Settings.Settings.photo_generate_button`

6.6.4.43 photo_h_t

`Settings.Settings.photo_h_t`

6.6.4.44 photo_p_c

`Settings.Settings.photo_p_c`

6.6.4.45 photo_r_t

`Settings.Settings.photo_r_t`

6.6.4.46 photo_s_c

`Settings.Settings.photo_s_c`

6.6.4.47 photo_text_input

`Settings.Settings.photo_text_input`

6.6.4.48 primary_colours

`Settings.Settings.primary_colours`

6.6.4.49 r_t

`Settings.Settings.r_t`

6.6.4.50 res_types

`Settings.Settings.res_types`

6.6.4.51 `resize_image_back_button`

`Settings.Settings.resize_image_back_button`

6.6.4.52 `resize_image_clear_button`

`Settings.Settings.resize_image_clear_button`

6.6.4.53 `resize_image_display_button`

`Settings.Settings.resize_image_display_button`

6.6.4.54 `resize_image_f_s`

`Settings.Settings.resize_image_f_s`

6.6.4.55 `resize_image_generate_button`

`Settings.Settings.resize_image_generate_button`

6.6.4.56 `resize_image_h_t`

`Settings.Settings.resize_image_h_t`

6.6.4.57 `resize_image_p_c`

`Settings.Settings.resize_image_p_c`

6.6.4.58 `resize_image_r_t`

`Settings.Settings.resize_image_r_t`

6.6.4.59 `resize_image_s_c`

`Settings.Settings.resize_image_s_c`

6.6.4.60 `resize_image_text_input`

`Settings.Settings.resize_image_text_input`

6.6.4.61 `resized_image_display`

`Settings.Settings.resized_image_display`

6.6.4.62 `resized_image_empty`

`Settings.Settings.resized_image_empty`

6.6.4.63 `s_c`

`Settings.Settings.s_c`

6.6.4.64 `secondary_colours`

`Settings.Settings.secondary_colours`

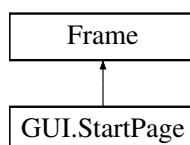
The documentation for this class was generated from the following file:

- [Settings.py](#)

6.7 GUI.StartPage Class Reference

[StartPage](#) is a start page that opens once the program starts.

Inheritance diagram for GUI.StartPage:



Public Member Functions

- `def __init__ (self, parent, controller)`
Creates the start page for the [GUI](#) main page.
- `def init_window (self)`
Places the different widgets onto the start page for the [GUI](#) main page.

Public Attributes

- [controller](#)
- [image_app_logo](#)
- [image_start_button](#)
- [image_gallery_button](#)
- [image_settings_button](#)
- [image_help_button](#)
- [resize_image_app_logo](#)
- [resize_image_start_button](#)
- [resize_image_gallery_button](#)
- [resize_image_settings_button](#)
- [resize_image_help_button](#)
- [photo_app_logo](#)
- [photo_start_button](#)
- [photo_gallery_button](#)
- [photo_settings_button](#)
- [photo_help_button](#)
- [app_logo](#)
- [start_button](#)
- [start_generate](#)
- [gallery_button](#)
- [gallery_generate](#)
- [settings_button](#)
- [settings_generate](#)
- [help_button](#)
- [help_generate](#)

6.7.1 Detailed Description

[StartPage](#) is a start page that opens once the program starts.

It displays the start page frame.

the [StartPage](#) contains buttons to start the flag generation, change the settings, open the gallery, and read the instructions.

6.7.2 Constructor & Destructor Documentation

6.7.2.1 `__init__()`

```
def GUI.StartPage.__init__ (
    self,
    parent,
    controller )
```

Creates the start page for the [GUI](#) main page.

The start page will have the logo, start button, settings button, gallery button, and a help button.

Parameters

<i>self</i>	Current object, common first parameter for any method of a class.
<i>parent</i>	A widget that acts as the parent of self, current object. All widgets in tkinter except the root window require a parent
<i>controller</i>	Other objects that are designed to act as a shared point, allowing several pages of widgets to interact. It decouples the different pages, making them independent. The controller decides what page will be visible.

6.7.3 Member Function Documentation

6.7.3.1 `init_window()`

```
def GUI.StartPage.init_window (  
    self )
```

Places the different widgets onto the start page for the [GUI](#) main page.

The start page will include the logo, start button, settings button, gallery, and a help button.

6.7.4 Member Data Documentation

6.7.4.1 `app_logo`

```
GUI.StartPage.app_logo
```

6.7.4.2 `controller`

```
GUI.StartPage.controller
```

6.7.4.3 `gallery_button`

```
GUI.StartPage.gallery_button
```

6.7.4.4 gallery_generate

GUI.StartPage.gallery_generate

6.7.4.5 help_button

GUI.StartPage.help_button

6.7.4.6 help_generate

GUI.StartPage.help_generate

6.7.4.7 image_app_logo

GUI.StartPage.image_app_logo

6.7.4.8 image_gallery_button

GUI.StartPage.image_gallery_button

6.7.4.9 image_help_button

GUI.StartPage.image_help_button

6.7.4.10 image_settings_button

GUI.StartPage.image_settings_button

6.7.4.11 image_start_button

GUI.StartPage.image_start_button

6.7.4.12 photo_app_logo

`GUI.StartPage.photo_app_logo`

6.7.4.13 photo_gallery_button

`GUI.StartPage.photo_gallery_button`

6.7.4.14 photo_help_button

`GUI.StartPage.photo_help_button`

6.7.4.15 photo_settings_button

`GUI.StartPage.photo_settings_button`

6.7.4.16 photo_start_button

`GUI.StartPage.photo_start_button`

6.7.4.17 resize_image_app_logo

`GUI.StartPage.resize_image_app_logo`

6.7.4.18 resize_image_gallery_button

`GUI.StartPage.resize_image_gallery_button`

6.7.4.19 resize_image_help_button

`GUI.StartPage.resize_image_help_button`

6.7.4.20 `resize_image_settings_button`

`GUI.StartPage.resize_image_settings_button`

6.7.4.21 `resize_image_start_button`

`GUI.StartPage.resize_image_start_button`

6.7.4.22 `settings_button`

`GUI.StartPage.settings_button`

6.7.4.23 `settings_generate`

`GUI.StartPage.settings_generate`

6.7.4.24 `start_button`

`GUI.StartPage.start_button`

6.7.4.25 `start_generate`

`GUI.StartPage.start_generate`

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

- [GUI.py](#)

Chapter 7

File Documentation

7.1 DecisionUtilities.py File Reference

@title [DecisionUtilities](#)

Namespaces

- namespace [DecisionUtilities](#)

Functions

- def [DecisionUtilities.pad_hashcode](#) (hashcode)
Generates a padded hashcode if it is not the minimum required length.
- def [DecisionUtilities.choose_from_list](#) (source_list, index)
Generates a selection from an array.
- def [DecisionUtilities.map_decision](#) (max_digitsum, num_decisions, digitsum)
Maps a number to an index of an array.
- def [DecisionUtilities.split_sequence](#) (seq, length)
Generates a list of shorter tokens from a given input string.
- def [DecisionUtilities.hex2rgb](#) (hexvalue)
Generates a tuple of an RGB colour from a hexadecimal number.
- def [DecisionUtilities.diff](#) (num1, num2)
Calculates the absolute difference of two float values.

Variables

- int [DecisionUtilities.COLOR_QUANTITY](#) = 5
- int [DecisionUtilities.HEX_COLOR_LEN](#) = 6
- int [DecisionUtilities.HEX_BASE](#) = 16
- int [DecisionUtilities.MINIMUM_HASH_LEN](#) = COLOR_QUANTITY * HEX_COLOR_LEN
- int [DecisionUtilities.ASPECT_CONTROL_LEN](#) = 6
- int [DecisionUtilities.MAX_DECISION_VALUE](#) = 16777215
- bool [DecisionUtilities.DEBUG](#) = False

7.1.1 Detailed Description

@title [DecisionUtilities](#)

A collection of modules used by [HashToFlag](#) to grind the hashcode and map decisions to arrays.

Author

Akram Hannoufa

Date

2022-04-11

7.2 Display.py File Reference

@title [Display](#)

Classes

- class [Display.FlagDisplay](#)
[FlagDisplay](#) is a class that allows the user to generate and display the flag.

Namespaces

- namespace [Display](#)

7.2.1 Detailed Description

@title [Display](#)

A display option to display the generated flag.

Author

Ganghoon Park

Date

2022-04-11

7.3 FlagAssetsLib.py File Reference

@title [FlagAssetsLib](#)

Namespaces

- namespace [FlagAssetsLib](#)

Variables

- list [FlagAssetsLib.BASE_STRIPE_STYLES](#)
- list [FlagAssetsLib.OVERLAY_STRIPE_STYLES](#)
- list [FlagAssetsLib.STRIPE_NUMBER](#) = ['ONE', 'TWO', 'THREE', 'SIX', 'TWELVE']
- list [FlagAssetsLib.SYMBOL_LOCATION](#) = ['TOP_LEFT', 'CENTER', 'TOP_RIGHT']
- list [FlagAssetsLib.SYMBOL_NUMBER](#) = ['ONE', 'TWO']
- list [FlagAssetsLib.SYMBOL_TYPES](#) = ['NONE', 'MOON', 'ROUNDEL', 'SWORD']
- dictionary [FlagAssetsLib.colours2rgb](#)
- dictionary [FlagAssetsLib.low_res_flag_assets](#)
- dictionary [FlagAssetsLib.mid_res_flag_assets](#)
- dictionary [FlagAssetsLib.high_res_flag_assets](#)

7.3.1 Detailed Description

@title [FlagAssetsLib](#)

A library of constants and symbol/design options

Author

Akram Hannoufa, Nathaniel Hu

Date

2022-03-31

7.4 FlagGenerator.py File Reference

@title [FlagGenerator](#)

Classes

- class [FlagGenerator.FlagGenerator](#)
[FlagGenerator](#) is a class that implements and encapsulates various attributes and methods needed to generate the flag image files.

Namespaces

- namespace [FlagGenerator](#)

7.4.1 Detailed Description

@title [FlagGenerator](#)

A module for generating the flag image using a given input string and hashing algorithm

[FlagGenerator](#) module, uses [HashGenerator](#), [HashToFlag](#) and [JKARReader](#) modules; no exported constants or types, no state or environment variables, no state invariant or assumptions

Author

Nathaniel Hu, Akram Hannoufa

Date

2022-04-07

7.5 Gallery.py File Reference

@title [Gallery](#)

Classes

- class [Gallery.FlagGallery](#)
[FlagGallery](#) is a class that opens the gallery so that the user can look at the generated flags.

Namespaces

- namespace [Gallery](#)

7.5.1 Detailed Description

@title [Gallery](#)

A gallery to showcase all the generated flags.

Author

Ganghoon Park

Date

2022-04-11

7.6 GUI.py File Reference

@title [GUI](#)

Classes

- class [GUI.SampleApp](#)
SampleApp creates the graphical user interface and allows the window to switch between different frames.
- class [GUI.StartPage](#)
StartPage is a start page that opens once the program starts.

Namespaces

- namespace [GUI](#)

Variables

- [GUI.app](#) = SampleApp()

7.6.1 Detailed Description

@title [GUI](#)

A graphical user interface module uses all other modules to allow the user to communicate with the Random Flag Generator software

Author

Ganghoon Park

Date

2022-04-11

7.7 HashGenerator.py File Reference

@title [HashGenerator](#)

Namespaces

- namespace [HashGenerator](#)

Functions

- def [HashGenerator._get_hash_algo](#) (hash_type)
gets the hashing algorithm from the dictionary of available hashing algorithms using the given input hash type string
- def [HashGenerator._get_hash_hex](#) (hash_input, hash_algo)
gets the hexadecimal representation of the hashing digest using the given input string and hashing algorithm
- def [HashGenerator.hash_generator](#) (hash_input, hash_type='sha256')
generates a hashing digest using the given input string and hashing algorithm

7.7.1 Detailed Description

@title [HashGenerator](#)

A library module for getting the hexadecimal hash of a given string

[HashGenerator](#) module, uses no other modules; no exported constants or types, no state or environment variables, no state invariant or assumptions

Author

Nathaniel Hu

Date

2022-03-31

7.8 HashToFlag.py File Reference

@title [HashToFlag](#)

Namespaces

- namespace [HashToFlag](#)

Functions

- def [HashToFlag.grind_hash_for_colors](#) (hashcode)
Generates the array of colours to be used in the flag generation.
- def [HashToFlag.grind_hash_for_base_stripe_style](#) (hashcode)
Generates the base stripe style to be used in flag generation.
- def [HashToFlag.grind_hash_for_overlay_stripe_style](#) (hashcode)
Generates the overlay stripe style to be used in flag generation.
- def [HashToFlag.grind_hash_for_stripe_number](#) (hashcode)
Generates the number of stripes to be used in flag generation.
- def [HashToFlag.grind_hash_for_symbol_locations](#) (hashcode)
Generates the symbol location to be used in flag generation.
- def [HashToFlag.grind_hash_for_symbol_number](#) (hashcode)
Generates the number of symbols to be used in flag generation.
- def [HashToFlag.grind_hash_for_symbol_types](#) (hashcode)
Generates the symbol type to be used in flag generation.

7.8.1 Detailed Description

@title [HashToFlag](#)

A module with functions for taking a given hashcode input and generating the options for the flag to be generated.

Uses DecisionsUtilities and [FlagAssetsLib](#)

Author

Akram Hannoufa

Date

2022-03-15

7.9 Help.py File Reference

@title [Help](#)

Classes

- class [Help.Help](#)

Shows the help menu to teach the user how to use the software and how it works.

Namespaces

- namespace [Help](#)

7.9.1 Detailed Description

@title [Help](#)

A help option teach the user how to use the software and giving more information of it.

Author

Ganghoon Park

Date

2022-04-11

7.10 JKARReader.py File Reference

@title [JKARReader](#)

Namespaces

- namespace [JKAResult](#)

Functions

- def [JKAResult.parse_jka_file](#) (filename)
parses the input flag asset (.jka) file data into a pixel map

Variables

- string [JKAResult.FILLED_PIXEL](#) = "#"
- string [JKAResult.UNFILLED_PIXEL](#) = "."

7.10.1 Detailed Description

@title [JKAResult](#)

A library module for parsing .jka files for use in generating flags

[JKAResult](#) module, uses no other modules; exported constants FILLED_PIXEL and UNFILLED_PIXEL, no exported types, no state or environment variables, no state invariant, assumption that input .jka file exists in the flag assets directory

Author

Nathaniel Hu

Date

2022-04-07

7.11 Settings.py File Reference

@title [Settings](#)

Classes

- class [Settings.Settings](#)
Creates settings graphical user interface app for user to use the random flag generator and set their settings.

Namespaces

- namespace [Settings](#)

7.11.1 Detailed Description

@title [Settings](#)

A settings option to select differnt flag size, set certain features, such as colour, symbols, stripes, and select different hash type

Author

Ganghoon Park, Nathaniel Hu

Date

2022-04-11

Index

- `__init__`
 - `Display.FlagDisplay`, 26
 - `FlagGenerator.FlagGenerator`, 38
 - `Gallery.FlagGallery`, 35
 - `GUI.SampleApp`, 45
 - `GUI.StartPage`, 59
 - `Help.Help`, 43
 - `Settings.Settings`, 48
 - `_get_hash_algo`
 - `HashGenerator`, 17
 - `_get_hash_hex`
 - `HashGenerator`, 17
 - `_save_flag_image`
 - `FlagGenerator.FlagGenerator`, 38
 - `_set_flag_base_colour`
 - `FlagGenerator.FlagGenerator`, 39
 - `_set_flag_base_stripes`
 - `FlagGenerator.FlagGenerator`, 39
 - `_set_flag_dimensions`
 - `FlagGenerator.FlagGenerator`, 39
 - `_set_flag_overlay_stripes`
 - `FlagGenerator.FlagGenerator`, 39
 - `_set_flag_symbol`
 - `FlagGenerator.FlagGenerator`, 39
- `app`
 - `GUI`, 17
- `app_logo`
 - `GUI.StartPage`, 60
- `ASPECT_CONTROL_LEN`
 - `DecisionUtilities`, 12
- `BASE_STRIPE_STYLES`
 - `FlagAssetsLib`, 14
- `button_back`
 - `Display.FlagDisplay`, 29
 - `Gallery.FlagGallery`, 36
 - `Help.Help`, 43
 - `Settings.Settings`, 50
- `button_clear`
 - `Display.FlagDisplay`, 29
 - `Settings.Settings`, 50
- `button_display`
 - `Display.FlagDisplay`, 29
 - `Settings.Settings`, 50
- `button_generate`
 - `Display.FlagDisplay`, 30
 - `Settings.Settings`, 51
- `button_open_gallery`
 - `Gallery.FlagGallery`, 36
- `choose_from_list`
 - `DecisionUtilities`, 9
- `COLOR_QUANTITY`
 - `DecisionUtilities`, 12
- `colours`
 - `FlagGenerator.FlagGenerator`, 41
- `colours2rgb`
 - `FlagAssetsLib`, 14
- `controller`
 - `Display.FlagDisplay`, 30
 - `Gallery.FlagGallery`, 36
 - `GUI.StartPage`, 60
 - `Help.Help`, 43
 - `Settings.Settings`, 51
- `DEBUG`
 - `DecisionUtilities`, 12
- `DecisionUtilities`, 9
 - `ASPECT_CONTROL_LEN`, 12
 - `choose_from_list`, 9
 - `COLOR_QUANTITY`, 12
 - `DEBUG`, 12
 - `diff`, 10
 - `hex2rgb`, 10
 - `HEX_BASE`, 13
 - `HEX_COLOR_LEN`, 13
 - `map_decision`, 11
 - `MAX_DECISION_VALUE`, 13
 - `MINIMUM_HASH_LEN`, 13
 - `pad_hashcode`, 11
 - `split_sequence`, 12
- `DecisionUtilities.py`, 65
- `delete_image`
 - `Display.FlagDisplay`, 28, 30
 - `Settings.Settings`, 49, 51
- `diff`
 - `DecisionUtilities`, 10
- `Display`, 13
- `display`
 - `Display.FlagDisplay`, 28, 30
 - `Settings.Settings`, 49, 51
- `Display.FlagDisplay`, 25
 - `__init__`, 26
 - `button_back`, 29
 - `button_clear`, 29
 - `button_display`, 29
 - `button_generate`, 30
 - `controller`, 30
 - `delete_image`, 28, 30
 - `display`, 28, 30

- FG, 30
- generate, 28, 30
- get_input_string, 29
- image_back_button, 30
- image_clear_button, 30
- image_display, 31
- image_display_button, 31
- image_empty, 31
- image_generate_button, 31
- image_text_input, 31
- init_window, 29
- input_box, 31
- label, 31
- label_display, 31
- label_empty, 32
- photo_back_button, 32
- photo_clear_button, 32
- photo_display, 32
- photo_display_button, 32
- photo_empty, 32
- photo_generate_button, 32
- photo_text_input, 32
- resize_generate_button, 33
- resize_image_back_button, 33
- resize_image_clear_button, 33
- resize_image_display_button, 33
- resize_image_text_input, 33
- resized_image_display, 33
- resized_image_empty, 33
- val, 33
- Display.py, 66
- f_s
 - Settings.Settings, 51
- FG
 - Display.FlagDisplay, 30
 - Settings.Settings, 51
- FILLED_PIXEL
 - JKARReader, 23
- flag_image
 - FlagGenerator.FlagGenerator, 41
- flag_pixels
 - FlagGenerator.FlagGenerator, 41
- flag_symbols
 - Settings.Settings, 51
- FlagAssetsLib, 13
 - BASE_STRIPE_STYLES, 14
 - colours2rgb, 14
 - high_res_flag_assets, 14
 - low_res_flag_assets, 14
 - mid_res_flag_assets, 15
 - OVERLAY_STRIPE_STYLES, 15
 - STRIPE_NUMBER, 15
 - SYMBOL_LOCATION, 16
 - SYMBOL_NUMBER, 16
 - SYMBOL_TYPES, 16
- FlagAssetsLib.py, 66
- FlagGenerator, 16
- FlagGenerator.FlagGenerator, 37
 - __init__, 38
 - _save_flag_image, 38
 - _set_flag_base_colour, 39
 - _set_flag_base_stripes, 39
 - _set_flag_dimensions, 39
 - _set_flag_overlay_stripes, 39
 - _set_flag_symbol, 39
 - colours, 41
 - flag_image, 41
 - flag_pixels, 41
 - generate_flag, 40
 - generate_flag_data, 40
 - height, 41
 - settings, 41
 - stripe_info, 41
 - symbol_info, 41
 - width, 41
- FlagGenerator.py, 67
- frames
 - GUI.SampleApp, 46
- Gallery, 16
- Gallery.FlagGallery, 34
 - __init__, 35
 - button_back, 36
 - button_open_gallery, 36
 - controller, 36
 - image_back_button, 36
 - image_open_gallery_button, 36
 - init_window, 35
 - open_gallery, 35
 - open_gallery_win, 35
 - photo_back_button, 36
 - photo_open_gallery_button, 36
 - resize_image_back_button, 36
 - resize_image_open_gallery_button, 37
- Gallery.py, 68
- gallery_button
 - GUI.StartPage, 60
- gallery_generate
 - GUI.StartPage, 60
- generate
 - Display.FlagDisplay, 28, 30
 - Settings.Settings, 49, 51
- generate_flag
 - FlagGenerator.FlagGenerator, 40
- generate_flag_data
 - FlagGenerator.FlagGenerator, 40
- get_input_string
 - Display.FlagDisplay, 29
 - Settings.Settings, 50
- grind_hash_for_base_stripe_style
 - HashToFlag, 19
- grind_hash_for_colors
 - HashToFlag, 19
- grind_hash_for_overlay_stripe_style
 - HashToFlag, 20
- grind_hash_for_stripe_number
 - HashToFlag, 20

- grind_hash_for_symbol_locations
 - HashToFlag, 20
- grind_hash_for_symbol_number
 - HashToFlag, 21
- grind_hash_for_symbol_types
 - HashToFlag, 21
- GUI, 16
 - app, 17
- GUI.py, 68
- GUI.SampleApp, 45
 - __init__, 45
 - frames, 46
 - show_frame, 46
- GUI.StartPage, 58
 - __init__, 59
 - app_logo, 60
 - controller, 60
 - gallery_button, 60
 - gallery_generate, 60
 - help_button, 61
 - help_generate, 61
 - image_app_logo, 61
 - image_gallery_button, 61
 - image_help_button, 61
 - image_settings_button, 61
 - image_start_button, 61
 - init_window, 60
 - photo_app_logo, 61
 - photo_gallery_button, 62
 - photo_help_button, 62
 - photo_settings_button, 62
 - photo_start_button, 62
 - resize_image_app_logo, 62
 - resize_image_gallery_button, 62
 - resize_image_help_button, 62
 - resize_image_settings_button, 63
 - resize_image_start_button, 63
 - settings_button, 63
 - settings_generate, 63
 - start_button, 63
 - start_generate, 63
- h_t
 - Settings.Settings, 52
- hash_generator
 - HashGenerator, 18
- hash_types
 - Settings.Settings, 52
- HashGenerator, 17
 - _get_hash_algo, 17
 - _get_hash_hex, 17
 - hash_generator, 18
- HashGenerator.py, 69
- HashToFlag, 18
 - grind_hash_for_base_stripe_style, 19
 - grind_hash_for_colors, 19
 - grind_hash_for_overlay_stripe_style, 20
 - grind_hash_for_stripe_number, 20
 - grind_hash_for_symbol_locations, 20
 - grind_hash_for_symbol_number, 21
 - grind_hash_for_symbol_types, 21
- HashToFlag.py, 70
- height
 - FlagGenerator.FlagGenerator, 41
- Help, 22
- Help.Help, 42
 - __init__, 43
 - button_back, 43
 - controller, 43
 - image, 44
 - image_back_button, 44
 - init_window, 43
 - label, 44
 - photo, 44
 - photo_back_button, 44
 - resize_image_back_button, 44
 - resized_image, 44
- Help.py, 71
- help_button
 - GUI.StartPage, 61
- help_generate
 - GUI.StartPage, 61
- hex2rgb
 - DecisionUtilities, 10
- HEX_BASE
 - DecisionUtilities, 13
- HEX_COLOR_LEN
 - DecisionUtilities, 13
- high_res_flag_assets
 - FlagAssetsLib, 14
- image
 - Help.Help, 44
- image_app_logo
 - GUI.StartPage, 61
- image_back_button
 - Display.FlagDisplay, 30
 - Gallery.FlagGallery, 36
 - Help.Help, 44
 - Settings.Settings, 52
- image_clear_button
 - Display.FlagDisplay, 30
 - Settings.Settings, 52
- image_display
 - Display.FlagDisplay, 31
 - Settings.Settings, 52
- image_display_button
 - Display.FlagDisplay, 31
 - Settings.Settings, 52
- image_empty
 - Display.FlagDisplay, 31
 - Settings.Settings, 52
- image_f_s
 - Settings.Settings, 52
- image_gallery_button
 - GUI.StartPage, 61
- image_generate_button
 - Display.FlagDisplay, 31

- Settings.Settings, 53
- image_h_t
 - Settings.Settings, 53
- image_help_button
 - GUI.StartPage, 61
- image_open_gallery_button
 - Gallery.FlagGallery, 36
- image_p_c
 - Settings.Settings, 53
- image_r_t
 - Settings.Settings, 53
- image_s_c
 - Settings.Settings, 53
- image_settings_button
 - GUI.StartPage, 61
- image_start_button
 - GUI.StartPage, 61
- image_text_input
 - Display.FlagDisplay, 31
 - Settings.Settings, 53
- init_window
 - Display.FlagDisplay, 29
 - Gallery.FlagGallery, 35
 - GUI.StartPage, 60
 - Help.Help, 43
 - Settings.Settings, 50
- input_box
 - Display.FlagDisplay, 31
 - Settings.Settings, 53
- JKARader, 22
 - FILLED_PIXEL, 23
 - parse_jka_file, 22
 - UNFILLED_PIXEL, 23
- JKARader.py, 71
- label
 - Display.FlagDisplay, 31
 - Help.Help, 44
- label1
 - Settings.Settings, 53
- label2
 - Settings.Settings, 54
- label3
 - Settings.Settings, 54
- label4
 - Settings.Settings, 54
- label5
 - Settings.Settings, 54
- label_display
 - Display.FlagDisplay, 31
 - Settings.Settings, 54
- label_empty
 - Display.FlagDisplay, 32
 - Settings.Settings, 54
- label_t_i
 - Settings.Settings, 54
- low_res_flag_assets
 - FlagAssetsLib, 14
- map_decision
 - DecisionUtilities, 11
- MAX_DECISION_VALUE
 - DecisionUtilities, 13
- mid_res_flag_assets
 - FlagAssetsLib, 15
- MINIMUM_HASH_LEN
 - DecisionUtilities, 13
- open_gallery
 - Gallery.FlagGallery, 35
- open_gallery_win
 - Gallery.FlagGallery, 35
- OVERLAY_STRIPE_STYLES
 - FlagAssetsLib, 15
- p_c
 - Settings.Settings, 54
- pad_hashcode
 - DecisionUtilities, 11
- parse_jka_file
 - JKARader, 22
- photo
 - Help.Help, 44
- photo_app_logo
 - GUI.StartPage, 61
- photo_back_button
 - Display.FlagDisplay, 32
 - Gallery.FlagGallery, 36
 - Help.Help, 44
 - Settings.Settings, 55
- photo_clear_button
 - Display.FlagDisplay, 32
 - Settings.Settings, 55
- photo_display
 - Display.FlagDisplay, 32
 - Settings.Settings, 55
- photo_display_button
 - Display.FlagDisplay, 32
 - Settings.Settings, 55
- photo_empty
 - Display.FlagDisplay, 32
 - Settings.Settings, 55
- photo_f_s
 - Settings.Settings, 55
- photo_gallery_button
 - GUI.StartPage, 62
- photo_generate_button
 - Display.FlagDisplay, 32
 - Settings.Settings, 55
- photo_h_t
 - Settings.Settings, 55
- photo_help_button
 - GUI.StartPage, 62
- photo_open_gallery_button
 - Gallery.FlagGallery, 36
- photo_p_c
 - Settings.Settings, 56
- photo_r_t

- Settings.Settings, 56
- photo_s_c
 - Settings.Settings, 56
- photo_settings_button
 - GUI.StartPage, 62
- photo_start_button
 - GUI.StartPage, 62
- photo_text_input
 - Display.FlagDisplay, 32
 - Settings.Settings, 56
- primary_colours
 - Settings.Settings, 56
- r_t
 - Settings.Settings, 56
- res_types
 - Settings.Settings, 56
- resize_generate_button
 - Display.FlagDisplay, 33
- resize_image_app_logo
 - GUI.StartPage, 62
- resize_image_back_button
 - Display.FlagDisplay, 33
 - Gallery.FlagGallery, 36
 - Help.Help, 44
 - Settings.Settings, 56
- resize_image_clear_button
 - Display.FlagDisplay, 33
 - Settings.Settings, 57
- resize_image_display_button
 - Display.FlagDisplay, 33
 - Settings.Settings, 57
- resize_image_f_s
 - Settings.Settings, 57
- resize_image_gallery_button
 - GUI.StartPage, 62
- resize_image_generate_button
 - Settings.Settings, 57
- resize_image_h_t
 - Settings.Settings, 57
- resize_image_help_button
 - GUI.StartPage, 62
- resize_image_open_gallery_button
 - Gallery.FlagGallery, 37
- resize_image_p_c
 - Settings.Settings, 57
- resize_image_r_t
 - Settings.Settings, 57
- resize_image_s_c
 - Settings.Settings, 57
- resize_image_settings_button
 - GUI.StartPage, 62
- resize_image_start_button
 - GUI.StartPage, 63
- resize_image_text_input
 - Display.FlagDisplay, 33
 - Settings.Settings, 58
- resized_image
 - Help.Help, 44
- resized_image_display
 - Display.FlagDisplay, 33
 - Settings.Settings, 58
- resized_image_empty
 - Display.FlagDisplay, 33
 - Settings.Settings, 58
- s_c
 - Settings.Settings, 58
- secondary_colours
 - Settings.Settings, 58
- Settings, 23
- settings
 - FlagGenerator.FlagGenerator, 41
- Settings.py, 72
- Settings.Settings, 46
 - __init__, 48
 - button_back, 50
 - button_clear, 50
 - button_display, 50
 - button_generate, 51
 - controller, 51
 - delete_image, 49, 51
 - display, 49, 51
 - f_s, 51
 - FG, 51
 - flag_symbols, 51
 - generate, 49, 51
 - get_input_string, 50
 - h_t, 52
 - hash_types, 52
 - image_back_button, 52
 - image_clear_button, 52
 - image_display, 52
 - image_display_button, 52
 - image_empty, 52
 - image_f_s, 52
 - image_generate_button, 53
 - image_h_t, 53
 - image_p_c, 53
 - image_r_t, 53
 - image_s_c, 53
 - image_text_input, 53
 - init_window, 50
 - input_box, 53
 - label1, 53
 - label2, 54
 - label3, 54
 - label4, 54
 - label5, 54
 - label_display, 54
 - label_empty, 54
 - label_t_i, 54
 - p_c, 54
 - photo_back_button, 55
 - photo_clear_button, 55
 - photo_display, 55
 - photo_display_button, 55
 - photo_empty, 55

- photo_f_s, [55](#)
- photo_generate_button, [55](#)
- photo_h_t, [55](#)
- photo_p_c, [56](#)
- photo_r_t, [56](#)
- photo_s_c, [56](#)
- photo_text_input, [56](#)
- primary_colours, [56](#)
- r_t, [56](#)
- res_types, [56](#)
- resize_image_back_button, [56](#)
- resize_image_clear_button, [57](#)
- resize_image_display_button, [57](#)
- resize_image_f_s, [57](#)
- resize_image_generate_button, [57](#)
- resize_image_h_t, [57](#)
- resize_image_p_c, [57](#)
- resize_image_r_t, [57](#)
- resize_image_s_c, [57](#)
- resize_image_text_input, [58](#)
- resized_image_display, [58](#)
- resized_image_empty, [58](#)
- s_c, [58](#)
- secondary_colours, [58](#)
- settings_button
 - GUI.StartPage, [63](#)
- settings_generate
 - GUI.StartPage, [63](#)
- show_frame
 - GUI.SampleApp, [46](#)
- split_sequence
 - DecisionUtilities, [12](#)
- start_button
 - GUI.StartPage, [63](#)
- start_generate
 - GUI.StartPage, [63](#)
- stripe_info
 - FlagGenerator.FlagGenerator, [41](#)
- STRIPE_NUMBER
 - FlagAssetsLib, [15](#)
- symbol_info
 - FlagGenerator.FlagGenerator, [41](#)
- SYMBOL_LOCATION
 - FlagAssetsLib, [16](#)
- SYMBOL_NUMBER
 - FlagAssetsLib, [16](#)
- SYMBOL_TYPES
 - FlagAssetsLib, [16](#)
- UNFILLED_PIXEL
 - JKARReader, [23](#)
- val
 - Display.FlagDisplay, [33](#)
- width
 - FlagGenerator.FlagGenerator, [41](#)