

## Famished Buffs

Generated by Doxygen 1.8.6

Mon Dec 7 2015 17:23:03



# Contents

<b>1</b>	<b>Namespace Index</b>	<b>1</b>
1.1	Namespace List . . . . .	1
<b>2</b>	<b>File Index</b>	<b>3</b>
2.1	File List . . . . .	3
<b>3</b>	<b>Namespace Documentation</b>	<b>5</b>
3.1	boot Namespace Reference . . . . .	5
3.2	cgiScript Namespace Reference . . . . .	5
3.2.1	Variable Documentation . . . . .	5
3.2.1.1	handler . . . . .	5
3.2.1.2	httpd . . . . .	5
3.2.1.3	server . . . . .	5
3.2.1.4	server_address . . . . .	5
3.3	code Namespace Reference . . . . .	5
3.3.1	Function Documentation . . . . .	6
3.3.1.1	buildLocation . . . . .	6
3.3.1.2	endBody . . . . .	6
3.3.1.3	endLocation . . . . .	6
3.3.1.4	endStation . . . . .	6
3.3.1.5	GenerateSite . . . . .	6
3.3.1.6	genIDforLocation . . . . .	6
3.3.1.7	genIDforStation . . . . .	7
3.3.1.8	genIDs . . . . .	7
3.3.1.9	genMeal . . . . .	7
3.3.1.10	getHead . . . . .	7
3.3.1.11	getMealStr . . . . .	7
3.3.1.12	main . . . . .	7
3.3.1.13	makeVariableName . . . . .	7
3.3.1.14	startBody . . . . .	8
3.3.1.15	startLocation . . . . .	8
3.3.1.16	startStation . . . . .	8

3.4	rotate Namespace Reference . . . . .	8
3.5	sqlpass Namespace Reference . . . . .	8
3.5.1	Variable Documentation . . . . .	8
3.5.1.1	password . . . . .	8
3.5.1.2	username . . . . .	8
3.6	style Namespace Reference . . . . .	8
<b>4</b>	<b>File Documentation</b>	<b>9</b>
4.1	boot.py File Reference . . . . .	9
4.2	cgiScript.py File Reference . . . . .	9
4.3	code.py File Reference . . . . .	9
4.4	rotate.py File Reference . . . . .	10
4.5	sqlpass.py File Reference . . . . .	10
4.6	style.py File Reference . . . . .	10
<b>Index</b>		<b>11</b>

# Chapter 1

## Namespace Index

### 1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

<a href="#">boot</a>	5
<a href="#">cgiScript</a>	5
<a href="#">code</a>	5
<a href="#">rotate</a>	8
<a href="#">sqlpass</a>	8
<a href="#">style</a>	8



## Chapter 2

# File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

<a href="#">boot.py</a>	9
<a href="#">cgiScript.py</a>	9
<a href="#">code.py</a>	9
<a href="#">rotate.py</a>	10
<a href="#">sqlpass.py</a>	10
<a href="#">style.py</a>	10





## Chapter 3

# Namespace Documentation

### 3.1 boot Namespace Reference

### 3.2 cgiScript Namespace Reference

#### Variables

- `server` = BaseHTTPServer.HTTPServer  
*This line enables CGI error reporting.*
- `handler` = CGIHTTPServer.CGIHTTPRequestHandler
- tuple `server_address` = ("", 8000)
- tuple `httpd` = `server(server_address, handler)`

#### 3.2.1 Variable Documentation

3.2.1.1 `cgiScript.handler` = CGIHTTPServer.CGIHTTPRequestHandler

3.2.1.2 tuple `cgiScript.httpd` = `server(server_address, handler)`

3.2.1.3 `cgiScript.server` = BaseHTTPServer.HTTPServer

This line enables CGI error reporting.

3.2.1.4 tuple `cgiScript.server_address` = ("", 8000)

### 3.3 code Namespace Reference

#### Functions

- def `genIDs`  
*Creates the ID strings for the divs in the javascript.*
- def `genIDforLocation`  
*Generates the slightly longer javascript ids for locations.*
- def `genIDforStation`  
*Generates the ID for stations that dont' need as much as locations.*
- def `getHead`  
*Generates head of html file.*

- def [startBody](#)  
*Returns overhead for the body of the html.*
- def [buildLocation](#)  
*Builds html for a specific location.*
- def [startLocation](#)  
*Overhead html for a location.*
- def [startStation](#)  
*Starts the html for a specific station.*
- def [endStation](#)  
*Balances the divs that were opened for each station.*
- def [genMeal](#)  
*Generates a meal card.*
- def [endLocation](#)
- def [endBody](#)
- def [makeVariableName](#)  
*Removes all non alpha numeric characters.*
- def [GenerateSite](#)  
*Pulls the whole thing together.*
- def [main](#)  
*Really just calls GenerateSite.*
- def [getMealStr](#)  
*Parses sql database into a nice string for meals.*

### 3.3.1 Function Documentation

#### 3.3.1.1 `def code.buildLocation ( TITLE, vals, hasStation = False )`

Builds html for a specific location.

Parameters

<i>TITLE</i>	Title of Location (e.g. C4C)
<i>vals</i>	if hasStation dict of Stations otherwise dict of meals and their items
<i>hasStation</i>	only true for the C4C, where there are stations

#### 3.3.1.2 `def code.endBody ( )`

#### 3.3.1.3 `def code.endLocation ( )`

#### 3.3.1.4 `def code.endStation ( )`

Balances the divs that were opened for each station.

#### 3.3.1.5 `def code.GenerateSite ( )`

Pulls the whole thing together.

This is where the real work is done.

#### 3.3.1.6 `def code.genIDforLocation ( s )`

Generates the slightly longer javascript ids for locations.

## Parameters

<i>s</i>	variable name like string to identify the station
----------	---

3.3.1.7 def code.genIDforStation ( *s* )

Generates the ID for stations that dont' need as much as locations.

## Parameters

<i>s</i>	string of id for station
----------	--------------------------

3.3.1.8 def code.genIDs ( *IDs* )

Creates the ID strings for the divs in the javascript.

## Parameters

<i>IDs</i>	a dictionary with keys of an ID for each location and a value of the stations below
------------	---

3.3.1.9 def code.genMeal ( *Title*, *Items* )

Generates a meal card.

## Parameters

<i>Title</i>	Title of meal (e.g. Breakfast)
<i>Items</i>	list of items available at that meal

3.3.1.10 def code.getHead ( *IDs* )

Generates head of html file.

## Parameters

<i>IDs</i>	a dictionary with keys of an ID for each location and a value of the stations below
------------	---

3.3.1.11 def code.getMealStr ( *x* )

Parses sql database into a nice string for meals.

## 3.3.1.12 def code.main ( )

Really just calls GenerateSite.

3.3.1.13 def code.makeVariableName ( *s* )

Removes all non alpha numeric characters.

## Parameters

s	string identifier
---	-------------------

#### 3.3.1.14 `def code.startBody ( )`

Returns overhead for the body of the html.

#### 3.3.1.15 `def code.startLocation ( ID, TITLE )`

Overhead html for a location.

Parameters

<i>ID</i>	a variable name string for location for javascript
<i>TITLE</i>	What the page displays as the title for the location

#### 3.3.1.16 `def code.startStation ( ID, TITLE )`

Starts the html for a specific station.

### 3.4 rotate Namespace Reference

### 3.5 sqlpass Namespace Reference

Variables

- string `password` = ""
- string `username` = "root"

#### 3.5.1 Variable Documentation

3.5.1.1 string `sqlpass.password` = ""

3.5.1.2 string `sqlpass.username` = "root"

### 3.6 style Namespace Reference

## Chapter 4

# File Documentation

### 4.1 boot.py File Reference

#### Namespaces

- [boot](#)

### 4.2 cgiScript.py File Reference

#### Namespaces

- [cgiScript](#)

#### Variables

- [cgiScript.server](#) = BaseHTTPServer.HTTPServer  
*This line enables CGI error reporting.*
- [cgiScript.handler](#) = CGIHTTPServer.CGIHTTPRequestHandler
- tuple [cgiScript.server\\_address](#) = ("", 8000)
- tuple [cgiScript.httpd](#) = server(server\_address, handler)

### 4.3 code.py File Reference

#### Namespaces

- [code](#)

#### Functions

- def [code.genIDs](#)  
*Creates the ID strings for the divs in the javascript.*
- def [code.genIDforLocation](#)  
*Generates the slightly longer javascript ids for locations.*
- def [code.genIDforStation](#)  
*Generates the ID for stations that dont' need as much as locations.*
- def [code.getHead](#)

- Generates head of html file.*
- def [code.startBody](#)  
*Returns overhead for the body of the html.*
- def [code.buildLocation](#)  
*Builds html for a specific location.*
- def [code.startLocation](#)  
*Overhead html for a location.*
- def [code.startStation](#)  
*Starts the html for a specific station.*
- def [code.endStation](#)  
*Balances the divs that were opened for each station.*
- def [code.genMeal](#)  
*Generates a meal card.*
- def [code.endLocation](#)
- def [code.endBody](#)
- def [code.makeVariableName](#)  
*Removes all non alpha numeric characters.*
- def [code.GenerateSite](#)  
*Pulls the whole thing together.*
- def [code.main](#)  
*Really just calls GenerateSite.*
- def [code.getMealStr](#)  
*Parses sql database into a nice string for meals.*

## 4.4 rotate.py File Reference

### Namespaces

- [rotate](#)

## 4.5 sqlpass.py File Reference

### Namespaces

- [sqlpass](#)

### Variables

- string [sqlpass.password](#) = ""
- string [sqlpass.username](#) = "root"

## 4.6 style.py File Reference

### Namespaces

- [style](#)

# Index

- boot, [5](#)
- boot.py, [9](#)
- buildLocation
  - code, [6](#)
- cgiScript, [5](#)
  - handler, [5](#)
  - httpd, [5](#)
  - server, [5](#)
  - server\_address, [5](#)
- cgiScript.py, [9](#)
- code, [5](#)
  - buildLocation, [6](#)
  - endBody, [6](#)
  - endLocation, [6](#)
  - endStation, [6](#)
  - genIDforLocation, [6](#)
  - genIDforStation, [7](#)
  - genIDs, [7](#)
  - genMeal, [7](#)
  - GenerateSite, [6](#)
  - getHead, [7](#)
  - getMealStr, [7](#)
  - main, [7](#)
  - makeVariableName, [7](#)
  - startBody, [8](#)
  - startLocation, [8](#)
  - startStation, [8](#)
- code.py, [9](#)
- endBody
  - code, [6](#)
- endLocation
  - code, [6](#)
- endStation
  - code, [6](#)
- genIDforLocation
  - code, [6](#)
- genIDforStation
  - code, [7](#)
- genIDs
  - code, [7](#)
- genMeal
  - code, [7](#)
- GenerateSite
  - code, [6](#)
- getHead
  - code, [7](#)
- getMealStr
  - code, [7](#)
- code, [7](#)
- handler
  - cgiScript, [5](#)
- httpd
  - cgiScript, [5](#)
- main
  - code, [7](#)
- makeVariableName
  - code, [7](#)
- password
  - sqlpass, [8](#)
- rotate, [8](#)
- rotate.py, [10](#)
- server
  - cgiScript, [5](#)
- server\_address
  - cgiScript, [5](#)
- sqlpass, [8](#)
  - password, [8](#)
  - username, [8](#)
- sqlpass.py, [10](#)
- startBody
  - code, [8](#)
- startLocation
  - code, [8](#)
- startStation
  - code, [8](#)
- style, [8](#)
- style.py, [10](#)
- username
  - sqlpass, [8](#)