

## Spice Grinder

Generated by Doxygen 1.8.11



# Contents

|          |                                       |          |
|----------|---------------------------------------|----------|
| <b>1</b> | <b>Namespace Index</b>                | <b>1</b> |
| 1.1      | Namespace List . . . . .              | 1        |
| <b>2</b> | <b>Hierarchical Index</b>             | <b>3</b> |
| 2.1      | Class Hierarchy . . . . .             | 3        |
| <b>3</b> | <b>Class Index</b>                    | <b>5</b> |
| 3.1      | Class List . . . . .                  | 5        |
| <b>4</b> | <b>File Index</b>                     | <b>7</b> |
| 4.1      | File List . . . . .                   | 7        |
| <b>5</b> | <b>Namespace Documentation</b>        | <b>9</b> |
| 5.1      | app Namespace Reference . . . . .     | 9        |
| 5.1.1    | Function Documentation . . . . .      | 10       |
| 5.1.1.1  | application(request) . . . . .        | 10       |
| 5.1.1.2  | cleanAndExit() . . . . .              | 10       |
| 5.1.1.3  | grindSpice(motor, amount) . . . . .   | 10       |
| 5.1.1.4  | grindSpices(kwargs) . . . . .         | 10       |
| 5.1.1.5  | hello(kwargs) . . . . .               | 10       |
| 5.1.1.6  | toggle(kwargs) . . . . .              | 10       |
| 5.1.1.7  | toggle_motor(toggle, motor) . . . . . | 10       |
| 5.1.2    | Variable Documentation . . . . .      | 10       |
| 5.1.2.1  | CCW . . . . .                         | 10       |
| 5.1.2.2  | CW . . . . .                          | 10       |

|          |                                     |    |
|----------|-------------------------------------|----|
| 5.1.2.3  | Dir                                 | 10 |
| 5.1.2.4  | Ins                                 | 10 |
| 5.1.2.5  | Motor1                              | 10 |
| 5.1.2.6  | Motor2                              | 10 |
| 5.1.2.7  | Motor3                              | 10 |
| 5.1.2.8  | Motor4                              | 10 |
| 5.1.2.9  | Motor5                              | 10 |
| 5.1.2.10 | Motor6                              | 10 |
| 5.1.2.11 | motorDict                           | 10 |
| 5.1.2.12 | Outs                                | 11 |
| 5.1.2.13 | Scale1                              | 11 |
| 5.1.2.14 | Scale2                              | 11 |
| 5.1.2.15 | Scale3                              | 11 |
| 5.1.2.16 | Scale4                              | 11 |
| 5.1.2.17 | Scale5                              | 11 |
| 5.1.2.18 | Scale6                              | 11 |
| 5.1.2.19 | Sleep                               | 11 |
| 5.1.2.20 | Step                                | 11 |
| 5.2      | buttonPressed Namespace Reference   | 11 |
| 5.2.1    | Variable Documentation              | 12 |
| 5.2.1.1  | button                              | 12 |
| 5.3      | conversion_func Namespace Reference | 12 |
| 5.3.1    | Function Documentation              | 12 |
| 5.3.1.1  | CupToTsp(Cup)                       | 12 |
| 5.3.1.2  | GramToKg(gram)                      | 12 |
| 5.3.1.3  | GramToOunce(gram)                   | 12 |
| 5.3.1.4  | GramToTsp(cursor, gram, name)       | 12 |
| 5.3.1.5  | KgToGram(kilogram)                  | 12 |
| 5.3.1.6  | OunceToGram(ounce)                  | 12 |
| 5.3.1.7  | PoundGramTo(gram)                   | 12 |

|          |   |    |
|----------|---|----|
| 5.3.1.8  | PoundToGram(Pound)  | 12 |
| 5.3.1.9  | TspToCup(tsp)   | 12 |
| 5.3.1.10 | TspToGram(cursor, tsp, name)  | 12 |
| 5.4      | database_func Namespace Reference   | 12 |
| 5.4.1    | Function Documentation  | 13 |
| 5.4.1.1  | deleteTuple(cursor, tableName, attribute, field)  | 13 |
| 5.4.1.2  | insertNewRecipe(cursor, name, ingredient1, ingredient2, ingredient3, ingredient4, ingredient5, ingredient6, ingredient7, amount1, amount2, amount3, amount4, amount5, amount6, amount7) | 13 |
| 5.4.1.3  | insertNewSpice(cursor, name, gpt, available)  | 13 |
| 5.4.1.4  | printSortedTable(cursor, tableName, sortBy)   | 13 |
| 5.4.1.5  | printTable(cursor, tableName)   | 13 |
| 5.4.1.6  | printTuple(cursor, tableName, attribute, field)   | 13 |
| 5.4.1.7  | retrieveTuple(cursor, tableName, attribute, field)  | 13 |
| 5.5      | database_header Namespace Reference   | 13 |
| 5.5.1    | Variable Documentation  | 13 |
| 5.5.1.1  | createRecipeTable   | 13 |
| 5.5.1.2  | createSpiceTable  | 13 |
| 5.5.1.3  | deleteRecipeTable   | 14 |
| 5.5.1.4  | deleteSpiceTable  | 14 |
| 5.6      | database_main Namespace Reference   | 14 |
| 5.6.1    | Detailed Description  | 14 |
| 5.6.2    | Variable Documentation  | 14 |
| 5.6.2.1  | c   | 14 |
| 5.6.2.2  | conn  | 14 |
| 5.6.2.3  | spices  | 14 |
| 5.7      | example Namespace Reference   | 14 |
| 5.7.1    | Function Documentation  | 15 |
| 5.7.1.1  | cleanAndExit()  | 15 |
| 5.7.1.2  | print_time(threadName, delay)   | 15 |
| 5.7.2    | Variable Documentation  | 15 |

|          |                                   |    |
|----------|-----------------------------------|----|
| 5.7.2.1  | hx                                | 15 |
| 5.7.2.2  | val                               | 15 |
| 5.8      | hx711 Namespace Reference         | 15 |
| 5.9      | motors Namespace Reference        | 15 |
| 5.9.1    | Function Documentation            | 16 |
| 5.9.1.1  | motor(toggle, motor)              | 16 |
| 5.9.1.2  | steppermotor(rotation, direction) | 16 |
| 5.9.2    | Variable Documentation            | 16 |
| 5.9.2.1  | CCW                               | 16 |
| 5.9.2.2  | CW                                | 16 |
| 5.9.2.3  | Dir                               | 16 |
| 5.9.2.4  | Motor1                            | 16 |
| 5.9.2.5  | Motor2                            | 16 |
| 5.9.2.6  | Motor3                            | 16 |
| 5.9.2.7  | Motor4                            | 16 |
| 5.9.2.8  | Motor5                            | 16 |
| 5.9.2.9  | Motor6                            | 16 |
| 5.9.2.10 | Outs                              | 16 |
| 5.9.2.11 | Sleep                             | 16 |
| 5.9.2.12 | Step                              | 16 |
| 5.10     | server Namespace Reference        | 16 |
| 5.10.1   | Function Documentation            | 17 |
| 5.10.1.1 | grind_spices()                    | 17 |
| 5.10.1.2 | grindSpice(motor, amount)         | 17 |
| 5.10.1.3 | toggle_motor(toggle, motor)       | 17 |
| 5.10.2   | Variable Documentation            | 17 |
| 5.10.2.1 | app                               | 17 |
| 5.10.2.2 | CCW                               | 17 |
| 5.10.2.3 | CW                                | 17 |
| 5.10.2.4 | debug                             | 17 |

|           |                               |    |
|-----------|-------------------------------|----|
| 5.10.2.5  | Dir                           | 17 |
| 5.10.2.6  | host                          | 17 |
| 5.10.2.7  | Ins                           | 17 |
| 5.10.2.8  | methods                       | 17 |
| 5.10.2.9  | Motor1                        | 17 |
| 5.10.2.10 | Motor2                        | 18 |
| 5.10.2.11 | Motor3                        | 18 |
| 5.10.2.12 | Motor4                        | 18 |
| 5.10.2.13 | Motor5                        | 18 |
| 5.10.2.14 | Motor6                        | 18 |
| 5.10.2.15 | motorDict                     | 18 |
| 5.10.2.16 | Outs                          | 18 |
| 5.10.2.17 | Scale1                        | 18 |
| 5.10.2.18 | Scale2                        | 18 |
| 5.10.2.19 | Scale3                        | 18 |
| 5.10.2.20 | Scale4                        | 18 |
| 5.10.2.21 | Scale5                        | 18 |
| 5.10.2.22 | Scale6                        | 18 |
| 5.10.2.23 | Sleep                         | 19 |
| 5.10.2.24 | Step                          | 19 |
| 5.11      | sqlite_ex Namespace Reference | 19 |
| 5.11.1    | Detailed Description          | 19 |
| 5.11.2    | Variable Documentation        | 19 |
| 5.11.2.1  | c                             | 19 |
| 5.11.2.2  | conn                          | 19 |
| 5.11.2.3  | t                             | 19 |

|  |           |
|--|-----------|
| <b>6 Class Documentation</b>                             | <b>21</b> |
| 6.1 app.grindThread Class Reference . . . . .            | 21        |
| 6.2 server.grindThread Class Reference . . . . .         | 21        |
| 6.2.1 Constructor & Destructor Documentation . . . . .   | 21        |
| 6.2.1.1 __init__(self, motor, amount) . . . . .          | 21        |
| 6.2.2 Member Function Documentation . . . . .            | 21        |
| 6.2.2.1 run(self) . . . . .                              | 21        |
| 6.2.3 Member Data Documentation . . . . .                | 21        |
| 6.2.3.1 amount . . . . .                                 | 21        |
| 6.2.3.2 motor . . . . .                                  | 21        |
| 6.3 hx711.HX711 Class Reference . . . . .                | 22        |
| 6.3.1 Constructor & Destructor Documentation . . . . .   | 23        |
| 6.3.1.1 __init__(self, dout, pd_sck, gain=128) . . . . . | 23        |
| 6.3.1.2 __init__(self, dout, pd_sck, gain=128) . . . . . | 23        |
| 6.3.2 Member Function Documentation . . . . .            | 23        |
| 6.3.2.1 get_binary_string(self) . . . . .                | 23        |
| 6.3.2.2 get_binary_string(self) . . . . .                | 23        |
| 6.3.2.3 get_gain(self) . . . . .                         | 23        |
| 6.3.2.4 get_gain(self) . . . . .                         | 23        |
| 6.3.2.5 get_np_arr8_string(self) . . . . .               | 24        |
| 6.3.2.6 get_np_arr8_string(self) . . . . .               | 24        |
| 6.3.2.7 get_value(self, times=3) . . . . .               | 24        |
| 6.3.2.8 get_value(self, times=3) . . . . .               | 24        |
| 6.3.2.9 get_value_A(self, times=3) . . . . .             | 24        |
| 6.3.2.10 get_value_A(self, times=3) . . . . .            | 24        |
| 6.3.2.11 get_value_B(self, times=3) . . . . .            | 24        |
| 6.3.2.12 get_value_B(self, times=3) . . . . .            | 24        |
| 6.3.2.13 get_weight(self, times=3) . . . . .             | 24        |
| 6.3.2.14 get_weight(self, times=3) . . . . .             | 24        |
| 6.3.2.15 get_weight_A(self, times=3) . . . . .           | 24        |



|          |  |    |
|----------|--|----|
| 6.3.2.16 | <code>get_weight_A(self, times=3)</code>                                       | 24 |
| 6.3.2.17 | <code>get_weight_B(self, times=3)</code>                                       | 24 |
| 6.3.2.18 | <code>get_weight_B(self, times=3)</code>                                       | 24 |
| 6.3.2.19 | <code>is_ready(self)</code>  | 24 |
| 6.3.2.20 | <code>is_ready(self)</code>  | 24 |
| 6.3.2.21 | <code>power_down(self)</code>  | 24 |
| 6.3.2.22 | <code>power_down(self)</code>  | 24 |
| 6.3.2.23 | <code>power_up(self)</code>  | 24 |
| 6.3.2.24 | <code>power_up(self)</code>  | 24 |
| 6.3.2.25 | <code>read(self)</code>  | 24 |
| 6.3.2.26 | <code>read(self)</code>  | 24 |
| 6.3.2.27 | <code>read_average(self, times=3)</code>                                       | 24 |
| 6.3.2.28 | <code>read_average(self, times=3)</code>                                       | 25 |
| 6.3.2.29 | <code>read_long(self)</code>   | 25 |
| 6.3.2.30 | <code>read_long(self)</code>   | 25 |
| 6.3.2.31 | <code>read_median(self, times=3)</code>  | 25 |
| 6.3.2.32 | <code>read_median(self, times=3)</code>  | 25 |
| 6.3.2.33 | <code>read_np_arr8(self)</code>  | 25 |
| 6.3.2.34 | <code>read_np_arr8(self)</code>  | 25 |
| 6.3.2.35 | <code>reset(self)</code>   | 25 |
| 6.3.2.36 | <code>reset(self)</code>   | 25 |
| 6.3.2.37 | <code>set_gain(self, gain)</code>  | 25 |
| 6.3.2.38 | <code>set_gain(self, gain)</code>  | 25 |
| 6.3.2.39 | <code>set_offset(self, offset)</code>  | 25 |
| 6.3.2.40 | <code>set_offset(self, offset)</code>  | 25 |
| 6.3.2.41 | <code>set_offset_A(self, offset)</code>  | 25 |
| 6.3.2.42 | <code>set_offset_A(self, offset)</code>  | 25 |
| 6.3.2.43 | <code>set_offset_B(self, offset)</code>  | 25 |
| 6.3.2.44 | <code>set_offset_B(self, offset)</code>  | 25 |
| 6.3.2.45 | <code>set_reading_format(self, byte_format=""LSB"", bit_format=""MSB"")</code> | 25 |

|          |  |    |
|----------|--|----|
| 6.3.2.46 | <code>set_reading_format(self, byte_format=""LSB"", bit_format=""MSB"")</code> | 25 |
| 6.3.2.47 | <code>set_reference_unit(self, reference_unit)</code>                          | 25 |
| 6.3.2.48 | <code>set_reference_unit(self, reference_unit)</code>                          | 25 |
| 6.3.2.49 | <code>set_reference_unit_A(self, reference_unit)</code>                        | 25 |
| 6.3.2.50 | <code>set_reference_unit_A(self, reference_unit)</code>                        | 25 |
| 6.3.2.51 | <code>set_reference_unit_B(self, reference_unit)</code>                        | 26 |
| 6.3.2.52 | <code>set_reference_unit_B(self, reference_unit)</code>                        | 26 |
| 6.3.2.53 | <code>tare(self, times=15)</code>  | 26 |
| 6.3.2.54 | <code>tare(self, times=15)</code>  | 26 |
| 6.3.2.55 | <code>tare_A(self, times=15)</code>  | 26 |
| 6.3.2.56 | <code>tare_A(self, times=15)</code>  | 26 |
| 6.3.2.57 | <code>tare_B(self, times=15)</code>  | 26 |
| 6.3.2.58 | <code>tare_B(self, times=15)</code>  | 26 |
| 6.3.3    | Member Data Documentation  | 26 |
| 6.3.3.1  | <code>bit_format</code>  | 26 |
| 6.3.3.2  | <code>bit_range_values</code>  | 26 |
| 6.3.3.3  | <code>byte_format</code>   | 26 |
| 6.3.3.4  | <code>byte_range_values</code>   | 26 |
| 6.3.3.5  | <code>DOUT</code>  | 26 |
| 6.3.3.6  | <code>GAIN</code>  | 26 |
| 6.3.3.7  | <code>isNegative</code>  | 26 |
| 6.3.3.8  | <code>lastVal</code>   | 26 |
| 6.3.3.9  | <code>LSBit</code>   | 26 |
| 6.3.3.10 | <code>LSByte</code>  | 26 |
| 6.3.3.11 | <code>MSBIndex24Bit</code>   | 26 |
| 6.3.3.12 | <code>MSBIndex32Bit</code>   | 26 |
| 6.3.3.13 | <code>MSBit</code>   | 26 |
| 6.3.3.14 | <code>MSByte</code>  | 26 |
| 6.3.3.15 | <code>OFFSET</code>  | 26 |
| 6.3.3.16 | <code>OFFSET_B</code>  | 26 |
| 6.3.3.17 | <code>PD_SCK</code>  | 26 |
| 6.3.3.18 | <code>REFERENCE_UNIT</code>  | 26 |
| 6.3.3.19 | <code>REFERENCE_UNIT_B</code>  | 26 |

|   |           |
|---|-----------|
| <b>7 File Documentation</b>   | <b>27</b> |
| 7.1 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/conversion_func.py File Reference . . . . .            | 27        |
| 7.2 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/database_func.py File Reference . . . . .              | 27        |
| 7.3 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/database_header.py File Reference . . . . .            | 28        |
| 7.4 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/database_main.py File Reference . . . . .              | 28        |
| 7.5 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/sqlite_ex.py File Reference . . . . .                  | 28        |
| 7.6 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/app.py File Reference . . . . .            | 29        |
| 7.7 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/button↔Pressed.py File Reference . . . . . | 29        |
| 7.8 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/motors.py File Reference . . . . .         | 30        |
| 7.9 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/server.py File Reference . . . . .         | 30        |
| 7.10 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Scale_Code/example.py File Reference . . . . .           | 31        |
| 7.11 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/example.py File Reference . . . . .       | 32        |
| 7.12 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Scale_Code/hx711.py File Reference . . . . .             | 32        |
| 7.13 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/hx711.py File Reference . . . . .         | 32        |
| <b>Index</b>  | <b>33</b> |



# Chapter 1

## Namespace Index

### 1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

|                                 |    |
|---------------------------------|----|
| <a href="#">app</a>             | 9  |
| <a href="#">buttonPressed</a>   | 11 |
| <a href="#">conversion_func</a> | 12 |
| <a href="#">database_func</a>   | 12 |
| <a href="#">database_header</a> | 13 |
| <a href="#">database_main</a>   | 14 |
| <a href="#">example</a>         | 14 |
| <a href="#">hx711</a>           | 15 |
| <a href="#">motors</a>          | 15 |
| <a href="#">server</a>          | 16 |
| <a href="#">sqlite_ex</a>       | 19 |



## Chapter 2

# Hierarchical Index

### 2.1 Class Hierarchy

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

|                              |    |
|------------------------------|----|
| hx711.HX711 . . . . .        | 22 |
| Thread                       |    |
| app.grindThread . . . . .    | 21 |
| server.grindThread . . . . . | 21 |





## Chapter 3

# Class Index

### 3.1 Class List

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

|                                    |    |
|------------------------------------|----|
| <a href="#">app.grindThread</a>    | 21 |
| <a href="#">server.grindThread</a> | 21 |
| <a href="#">hx711.HX711</a>        | 22 |



## Chapter 4

# File Index

### 4.1 File List

Here is a list of all files with brief descriptions:

|  |    |
|--|----|
| /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/ <a href="#">conversion_func.py</a> . . . .                 | 27 |
| /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/ <a href="#">database_func.py</a> . . . .                   | 27 |
| /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/ <a href="#">database_header.py</a> . . . .                 | 28 |
| /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/ <a href="#">database_main.py</a> . . . .                   | 28 |
| /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/ <a href="#">sqlite_ex.py</a> . . . .                       | 28 |
| /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/ <a href="#">app.py</a> . . . .                 | 29 |
| /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/ <a href="#">button↵<br/>Pressed.py</a> . . . . | 29 |
| /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/ <a href="#">example.py</a> . .                 | 32 |
| /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/ <a href="#">hx711.py</a> . .                   | 32 |
| /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/ <a href="#">motors.py</a> . .                  | 30 |
| /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/ <a href="#">server.py</a> . .                  | 30 |
| /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Scale_Code/ <a href="#">example.py</a> . . .                   | 31 |
| /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Scale_Code/ <a href="#">hx711.py</a> . . . .                   | 32 |



## Chapter 5

# Namespace Documentation

### 5.1 app Namespace Reference

#### Classes

- class `grindThread`

#### Functions

- def `toggle_motor` (`toggle`, `motor`)
- def `grindSpice` (`motor`, `amount`)
- def `hello` (`kwargs`)
- def `toggle` (`kwargs`)
- def `grindSpices` (`kwargs`)
- def `application` (`request`)
- def `cleanAndExit` ()

#### Variables

- int `Motor1` = 2  
*THESE LINE WILL NEED TO BE ADDED TO THE MAIN CODE TO SETUP GPIO PINS#####.*
- list `Scale1` = [15,17]
- int `Motor2` = 3
- list `Scale2` = [18, 27]
- int `Motor3` = 4
- list `Scale3` = [22, 23]
- int `Motor4` = 5
- list `Scale4` = [24, 25]
- int `Motor5` = 6
- list `Scale5` = [19, 16]
- int `Motor6` = 7
- list `Scale6` = [26, 20]
- int `Sleep` = 12
- int `Step` = 13
- int `Dir` = 14
- int `CW` = 1
- int `CCW` = 0
- tuple `Outs` = (`Motor1`,`Motor2`,`Motor3`,`Motor4`,`Motor5`,`Motor6`,`Sleep`,`Step`,`Dir`,`Scale1`[1],`Scale2`[1],`Scale3`[1],`Scale4`[1],`Scale5`[1],`Scale6`[1])
- tuple `Ins` = (`Scale1`[0],`Scale2`[0],`Scale3`[0],`Scale4`[0],`Scale5`[0],`Scale6`[0])
- dictionary `motorDict`

### 5.1.1 Function Documentation

5.1.1.1 `def app.application ( request )`

5.1.1.2 `def app.cleanAndExit ( )`

5.1.1.3 `def app.grindSpice ( motor, amount )`

5.1.1.4 `def app.grindSpices ( kwargs )`

5.1.1.5 `def app.hello ( kwargs )`

5.1.1.6 `def app.toggle ( kwargs )`

5.1.1.7 `def app.toggle_motor ( toggle, motor )`

### 5.1.2 Variable Documentation

5.1.2.1 `int app.CCW = 0`

5.1.2.2 `int app.CW = 1`

5.1.2.3 `int app.Dir = 14`

5.1.2.4 `tuple app.Ins = (Scale1[0],Scale2[0],Scale3[0],Scale4[0],Scale5[0],Scale6[0])`

5.1.2.5 `int app.Motor1 = 2`

THESE LINE WILL NEED TO BE ADDED TO THE MAIN CODE TO SETUP GPIO PINS#####.

5.1.2.6 `int app.Motor2 = 3`

5.1.2.7 `int app.Motor3 = 4`

5.1.2.8 `int app.Motor4 = 5`

5.1.2.9 `int app.Motor5 = 6`

5.1.2.10 `int app.Motor6 = 7`

5.1.2.11 `dictionary app.motorDict`

**Initial value:**

```

1 = {
2     '0': {
3         'motor': 2,
4         'scale': [15, 17],
5         'reference': 6659
6     },
7     '1': {
8         'motor': 3,
9         'scale': [18, 27]
10    },
11    '2': {
12        'motor': 4,
13        'scale': [22, 23],
14        'reference': 6921
15    },
16    '3': {
17        'motor': 5,
18        'scale': [24, 25]
19    },
20    '4': {
21        'motor': 6,
22        'scale': [19, 16]
23    },
24    '5': {
25        'motor': 7,
26        'scale': [26, 20]
27    },
28 }

```

5.1.2.12 tuple app.Outs = (Motor1,Motor2,Motor3,Motor4,Motor5,Motor6,Sleep,Step,Dir,Scale1[1],Scale2[1],Scale3[1],Scale4[1],Scale5[1],Scale6[1])

5.1.2.13 list app.Scale1 = [15,17]

5.1.2.14 list app.Scale2 = [18, 27]

5.1.2.15 list app.Scale3 = [22, 23]

5.1.2.16 list app.Scale4 = [24, 25]

5.1.2.17 list app.Scale5 = [19, 16]

5.1.2.18 list app.Scale6 = [26, 20]

5.1.2.19 int app.Sleep = 12

5.1.2.20 int app.Step = 13

## 5.2 buttonPressed Namespace Reference

### Variables

- [button](#) = Button(18)

### 5.2.1 Variable Documentation

5.2.1.1 `buttonPressed.button = Button(18)`

## 5.3 `conversion_func` Namespace Reference

### Functions

- def [OunceToGram](#) (ounce)
- def [GramToOunce](#) (gram)
- def [KgToGram](#) (kilogram)
- def [GramToKg](#) (gram)
- def [PoundToGram](#) (Pound)
- def [PoundGramTo](#) (gram)
- def [TspToCup](#) (tsp)
- def [CupToTsp](#) (Cup)
- def [GramToTsp](#) (cursor, gram, name)
- def [TspToGram](#) (cursor, tsp, name)

### 5.3.1 Function Documentation

5.3.1.1 `def conversion_func.CupToTsp ( Cup )`

5.3.1.2 `def conversion_func.GramToKg ( gram )`

5.3.1.3 `def conversion_func.GramToOunce ( gram )`

5.3.1.4 `def conversion_func.GramToTsp ( cursor, gram, name )`

5.3.1.5 `def conversion_func.KgToGram ( kilogram )`

5.3.1.6 `def conversion_func.OunceToGram ( ounce )`

5.3.1.7 `def conversion_func.PoundGramTo ( gram )`

5.3.1.8 `def conversion_func.PoundToGram ( Pound )`

5.3.1.9 `def conversion_func.TspToCup ( tsp )`

5.3.1.10 `def conversion_func.TspToGram ( cursor, tsp, name )`

## 5.4 `database_func` Namespace Reference

### Functions

- def [printSortedTable](#) (cursor, tableName, sortBy)
- def [printTable](#) (cursor, tableName)
- def [printTuple](#) (cursor, tableName, attribute, field)
- def [retrieveTuple](#) (cursor, tableName, attribute, field)
- def [insertNewRecipe](#) (cursor, name, ingredient1, ingredient2, ingredient3, ingredient4, ingredient5, ingredient6, ingredient7, amount1, amount2, amount3, amount4, amount5, amount6, amount7)
- def [insertNewSpice](#) (cursor, name, gpt, available)
- def [deleteTuple](#) (cursor, tableName, attribute, field)



### 5.4.1 Function Documentation

5.4.1.1 `def database_func.deleteTuple ( cursor, tableName, attribute, field )`

5.4.1.2 `def database_func.insertNewRecipe ( cursor, name, ingredient1, ingredient2, ingredient3, ingredient4, ingredient5, ingredient6, ingredient7, amount1, amount2, amount3, amount4, amount5, amount6, amount7 )`

5.4.1.3 `def database_func.insertNewSpice ( cursor, name, gpt, available )`

5.4.1.4 `def database_func.printSortedTable ( cursor, tableName, sortBy )`

5.4.1.5 `def database_func.printTable ( cursor, tableName )`

5.4.1.6 `def database_func.printTuple ( cursor, tableName, attribute, field )`

5.4.1.7 `def database_func.retrieveTuple ( cursor, tableName, attribute, field )`

## 5.5 database\_header Namespace Reference

### Variables

- string `deleteRecipeTable` = 'DROP TABLE IF EXISTS 'Recipe';'
- string `deleteSpiceTable` = 'DROP TABLE IF EXISTS 'Spice';'
- string `createSpiceTable`
- string `createRecipeTable`

### 5.5.1 Variable Documentation

5.5.1.1 `string database_header.createRecipeTable`

5.5.1.2 `string database_header.createSpiceTable`

#### Initial value:

```
1 = '''CREATE TABLE Spice
2
3         (
4             id INTEGER PRIMARY KEY AUTOINCREMENT,
5             name text NOT NULL DEFAULT '',
6             grams_per_tsp real,
7             available BIT DEFAULT 0
8         )
9     '''
```

5.5.1.3 string database\_header.deleteRecipeTable = 'DROP TABLE IF EXISTS 'Recipe';'

5.5.1.4 string database\_header.deleteSpiceTable = 'DROP TABLE IF EXISTS 'Spice';'

## 5.6 database\_main Namespace Reference

### Variables

- `conn` = `sqlite3.connect('BurrGrinder.db')`
- `c` = `conn.cursor()`
- list `spices`

### 5.6.1 Detailed Description

see  
<https://docs.python.org/2/library/sqlite3.html>  
 for some documentations on `sqlite`

### 5.6.2 Variable Documentation

5.6.2.1 `database_main.c` = `conn.cursor()`

5.6.2.2 `database_main.conn` = `sqlite3.connect('BurrGrinder.db')`

5.6.2.3 list `database_main.spices`

#### Initial value:

```
1 = [
2     ('Pepper', 6.4, 1),
3     ('Salt', 5.69, 1),
4     ('Paprika', 1.6, 0),
5     ('Parsley', 2.1, 0),
6     ('Cinnamon', 2.3, 0),
7     ('Cloves', 6.6, 0)
8 ]
```

## 5.7 example Namespace Reference

### Functions

- def `cleanAndExit` ()
- def `print_time` (threadName, delay)

### Variables

- `hx` = `HX711(22, 23)`
- `val` = `hx.get_weight(5)`

### 5.7.1 Function Documentation

5.7.1.1 `def example.cleanAndExit ( )`

5.7.1.2 `def example.print_time ( threadName, delay )`

### 5.7.2 Variable Documentation

5.7.2.1 `example.hx = HX711(22, 23)`

5.7.2.2 `example.val = hx.get_weight(5)`

## 5.8 hx711 Namespace Reference

### Classes

- class [HX711](#)

## 5.9 motors Namespace Reference

### Functions

- `def motor (toggle, motor)`
- `def steppermotor (rotation, direction)`

### Variables

- `int Motor1 = 2`  
*THESE LINE WILL NEED TO BE ADDED TO THE MAIN CODE TO SETUP GPIO PINS#####.*
- `int Motor2 = 3`
- `int Motor3 = 4`
- `int Motor4 = 5`
- `int Motor5 = 6`
- `int Motor6 = 7`
- `int Sleep = 12`
- `int Step = 13`
- `int Dir = 14`
- `int CW = 1`
- `int CCW = 0`
- `tuple Outs = (Motor1,Motor2,Motor3,Motor4,Motor5,Motor6,Sleep,Step,Dir)`

### 5.9.1 Function Documentation

5.9.1.1 `def motors.motor ( toggle, motor )`

5.9.1.2 `def motors.steppermotor ( rotation, direction )`

### 5.9.2 Variable Documentation

5.9.2.1 `int motors.CCW = 0`

5.9.2.2 `int motors.CW = 1`

5.9.2.3 `int motors.Dir = 14`

5.9.2.4 `int motors.Motor1 = 2`

THESE LINE WILL NEED TO BE ADDED TO THE MAIN CODE TO SETUP GPIO PINS#####.

5.9.2.5 `int motors.Motor2 = 3`

5.9.2.6 `int motors.Motor3 = 4`

5.9.2.7 `int motors.Motor4 = 5`

5.9.2.8 `int motors.Motor5 = 6`

5.9.2.9 `int motors.Motor6 = 7`

5.9.2.10 `tuple motors.Outs = (Motor1,Motor2,Motor3,Motor4,Motor5,Motor6,Sleep,Step,Dir)`

5.9.2.11 `int motors.Sleep = 12`

5.9.2.12 `int motors.Step = 13`

## 5.10 server Namespace Reference

### Classes

- class [grindThread](#)

### Functions

- def [grind\\_spices](#) ()
- def [toggle\\_motor](#) (toggle, motor)
- def [grindSpice](#) (motor, amount)

## Variables

- `app` = `Flask(__name__)`
- `methods`
- `int Motor1` = 2
- *THESE LINE WILL NEED TO BE ADDED TO THE MAIN CODE TO SETUP GPIO PINS#####.*
- `list Scale1` = [15,17]
- `int Motor2` = 3
- `list Scale2` = [18, 27]
- `int Motor3` = 4
- `list Scale3` = [22, 23]
- `int Motor4` = 5
- `list Scale4` = [24, 25]
- `int Motor5` = 6
- `list Scale5` = [19, 16]
- `int Motor6` = 7
- `list Scale6` = [26, 20]
- `int Sleep` = 12
- `int Step` = 13
- `int Dir` = 14
- `int CW` = 1
- `int CCW` = 0
- `tuple Outs` = (`Motor1`,`Motor2`,`Motor3`,`Motor4`,`Motor5`,`Motor6`,`Sleep`,`Step`,`Dir`,`Scale1`[1],`Scale2`[1],`Scale3`[1],`Scale4`[1],`Scale5`[1],`Scale6`[1])
- `tuple Ins` = (`Scale1`[0],`Scale2`[0],`Scale3`[0],`Scale4`[0],`Scale5`[0],`Scale6`[0])
- dictionary `motorDict`
- `host`
- `debug`

### 5.10.1 Function Documentation

- 5.10.1.1 `def server.grind_spices ( )`
- 5.10.1.2 `def server.grindSpice ( motor, amount )`
- 5.10.1.3 `def server.toggle_motor ( toggle, motor )`

### 5.10.2 Variable Documentation

- 5.10.2.1 `server.app` = `Flask(__name__)`
- 5.10.2.2 `int server.CCW` = 0
- 5.10.2.3 `int server.CW` = 1
- 5.10.2.4 `server.debug`
- 5.10.2.5 `int server.Dir` = 14
- 5.10.2.6 `server.host`
- 5.10.2.7 `tuple server.Ins` = (`Scale1`[0],`Scale2`[0],`Scale3`[0],`Scale4`[0],`Scale5`[0],`Scale6`[0])
- 5.10.2.8 `server.methods`
- 5.10.2.9 `int server.Motor1` = 2

THESE LINE WILL NEED TO BE ADDED TO THE MAIN CODE TO SETUP GPIO PINS#####.

5.10.2.10 `int server.Motor2 = 3`

5.10.2.11 `int server.Motor3 = 4`

5.10.2.12 `int server.Motor4 = 5`

5.10.2.13 `int server.Motor5 = 6`

5.10.2.14 `int server.Motor6 = 7`

5.10.2.15 `dictionary server.motorDict`

**Initial value:**

```

1 = {
2     '0': {
3         'motor': Motor1,
4         'scale': Scale1,
5         'reference': 6659
6     },
7     '1': {
8         'motor': Motor2,
9         'scale': Scale2
10    },
11    '2': {
12        'motor': Motor3,
13        'scale': Scale3,
14        'reference': 6921
15    },
16    '3': {
17        'motor': Motor4,
18        'scale': Scale4
19    },
20    '4': {
21        'motor': Motor5,
22        'scale': Scale5
23    },
24    '5': {
25        'motor': Motor6,
26        'scale': Scale6
27    },
28 }
```

5.10.2.16 `tuple server.Outs = (Motor1,Motor2,Motor3,Motor4,Motor5,Motor6,Sleep,Step,Dir,Scale1[1],Scale2[1],Scale3[1],Scale4[1],S`

5.10.2.17 `list server.Scale1 = [15,17]`

5.10.2.18 `list server.Scale2 = [18, 27]`

5.10.2.19 `list server.Scale3 = [22, 23]`

5.10.2.20 `list server.Scale4 = [24, 25]`

5.10.2.21 `list server.Scale5 = [19, 16]`

5.10.2.22 `list server.Scale6 = [26, 20]`

5.10.2.23 int server.Sleep = 12

5.10.2.24 int server.Step = 13

## 5.11 sqlite\_ex Namespace Reference

### Variables

- `conn` = `sqlite3.connect('example.db')`
- `c` = `conn.cursor()`
- tuple `t` = `('RHAT',)`

### 5.11.1 Detailed Description

example code from  
<https://docs.python.org/2/library/sqlite3.html>

### 5.11.2 Variable Documentation

5.11.2.1 `sqlite_ex.c` = `conn.cursor()`

5.11.2.2 `sqlite_ex.conn` = `sqlite3.connect('example.db')`

5.11.2.3 tuple `sqlite_ex.t` = `('RHAT',)`





## Chapter 6

# Class Documentation

### 6.1 app.grindThread Class Reference

Inheritance diagram for app.grindThread:

### 6.2 server.grindThread Class Reference

Inheritance diagram for server.grindThread:

Collaboration diagram for server.grindThread:

#### Public Member Functions

- def `__init__` (self, `motor`, `amount`)
- def `run` (self)

#### Public Attributes

- `motor`
- `amount`

#### 6.2.1 Constructor & Destructor Documentation

6.2.1.1 `def server.grindThread.__init__( self, motor, amount )`

#### 6.2.2 Member Function Documentation

6.2.2.1 `def server.grindThread.run ( self )`

#### 6.2.3 Member Data Documentation

6.2.3.1 `server.grindThread.amount`

6.2.3.2 `server.grindThread.motor`

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

- `/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/server.py`

## 6.3 hx711.HX711 Class Reference

### Public Member Functions

- def `__init__` (self, dout, pd\_sck, gain=128)
- def `is_ready` (self)
- def `set_gain` (self, gain)
- def `get_gain` (self)
- def `read` (self)
- def `get_binary_string` (self)
- def `get_np_arr8_string` (self)
- def `read_np_arr8` (self)
- def `read_long` (self)
- def `read_average` (self, times=3)
- def `read_median` (self, times=3)
- def `get_value` (self, times=3)
- def `get_value_A` (self, times=3)
- def `get_value_B` (self, times=3)
- def `get_weight` (self, times=3)
- def `get_weight_A` (self, times=3)
- def `get_weight_B` (self, times=3)
- def `tare` (self, times=15)
- def `tare_A` (self, times=15)
- def `tare_B` (self, times=15)
- def `set_reading_format` (self, byte\_format="LSB", bit\_format="MSB")
- def `set_offset` (self, offset)
- def `set_offset_A` (self, offset)
- def `set_offset_B` (self, offset)
- def `set_reference_unit` (self, reference\_unit)
- def `set_reference_unit_A` (self, reference\_unit)
- def `set_reference_unit_B` (self, reference\_unit)
- def `power_down` (self)
- def `power_up` (self)
- def `reset` (self)
- def `__init__` (self, dout, pd\_sck, gain=128)
- def `is_ready` (self)
- def `set_gain` (self, gain)
- def `get_gain` (self)
- def `read` (self)
- def `get_binary_string` (self)
- def `get_np_arr8_string` (self)
- def `read_np_arr8` (self)
- def `read_long` (self)
- def `read_average` (self, times=3)
- def `read_median` (self, times=3)
- def `get_value` (self, times=3)
- def `get_value_A` (self, times=3)
- def `get_value_B` (self, times=3)
- def `get_weight` (self, times=3)
- def `get_weight_A` (self, times=3)
- def `get_weight_B` (self, times=3)
- def `tare` (self, times=15)
- def `tare_A` (self, times=15)
- def `tare_B` (self, times=15)

- def [set\\_reading\\_format](#) (self, [byte\\_format](#)="LSB", [bit\\_format](#)="MSB")
- def [set\\_offset](#) (self, offset)
- def [set\\_offset\\_A](#) (self, offset)
- def [set\\_offset\\_B](#) (self, offset)
- def [set\\_reference\\_unit](#) (self, reference\_unit)
- def [set\\_reference\\_unit\\_A](#) (self, reference\_unit)
- def [set\\_reference\\_unit\\_B](#) (self, reference\_unit)
- def [power\\_down](#) (self)
- def [power\\_up](#) (self)
- def [reset](#) (self)

### Public Attributes

- [PD\\_SCK](#)
- [DOUT](#)
- [GAIN](#)
- [REFERENCE\\_UNIT](#)
- [REFERENCE\\_UNIT\\_B](#)
- [OFFSET](#)
- [OFFSET\\_B](#)
- [lastVal](#)
- [isNegative](#)
- [MSBIndex24Bit](#)
- [MSBIndex32Bit](#)
- [LSByte](#)
- [MSByte](#)
- [MSBit](#)
- [LSBit](#)
- [byte\\_format](#)
- [bit\\_format](#)
- [byte\\_range\\_values](#)
- [bit\\_range\\_values](#)

### 6.3.1 Constructor & Destructor Documentation

6.3.1.1 `def hx711.HX711.__init__ ( self, dout, pd_sck, gain = 128 )`

6.3.1.2 `def hx711.HX711.__init__ ( self, dout, pd_sck, gain = 128 )`

### 6.3.2 Member Function Documentation

6.3.2.1 `def hx711.HX711.get_binary_string ( self )`

6.3.2.2 `def hx711.HX711.get_binary_string ( self )`

6.3.2.3 `def hx711.HX711.get_gain ( self )`

6.3.2.4 `def hx711.HX711.get_gain ( self )`

6.3.2.5 `def hx711.HX711.get_np_arr8_string ( self )`

6.3.2.6 `def hx711.HX711.get_np_arr8_string ( self )`

6.3.2.7 `def hx711.HX711.get_value ( self, times = 3 )`

6.3.2.8 `def hx711.HX711.get_value ( self, times = 3 )`

6.3.2.9 `def hx711.HX711.get_value_A ( self, times = 3 )`

6.3.2.10 `def hx711.HX711.get_value_A ( self, times = 3 )`

6.3.2.11 `def hx711.HX711.get_value_B ( self, times = 3 )`

6.3.2.12 `def hx711.HX711.get_value_B ( self, times = 3 )`

6.3.2.13 `def hx711.HX711.get_weight ( self, times = 3 )`

6.3.2.14 `def hx711.HX711.get_weight ( self, times = 3 )`

6.3.2.15 `def hx711.HX711.get_weight_A ( self, times = 3 )`

6.3.2.16 `def hx711.HX711.get_weight_A ( self, times = 3 )`

6.3.2.17 `def hx711.HX711.get_weight_B ( self, times = 3 )`

6.3.2.18 `def hx711.HX711.get_weight_B ( self, times = 3 )`

6.3.2.19 `def hx711.HX711.is_ready ( self )`

6.3.2.20 `def hx711.HX711.is_ready ( self )`

6.3.2.21 `def hx711.HX711.power_down ( self )`

6.3.2.22 `def hx711.HX711.power_down ( self )`

6.3.2.23 `def hx711.HX711.power_up ( self )`

6.3.2.24 `def hx711.HX711.power_up ( self )`

6.3.2.25 `def hx711.HX711.read ( self )`

6.3.2.26 `def hx711.HX711.read ( self )`

6.3.2.27 `def hx711.HX711.read_average ( self, times = 3 )`

- 6.3.2.28 `def hx711.HX711.read_average ( self, times = 3 )`
- 6.3.2.29 `def hx711.HX711.read_long ( self )`
- 6.3.2.30 `def hx711.HX711.read_long ( self )`
- 6.3.2.31 `def hx711.HX711.read_median ( self, times = 3 )`
- 6.3.2.32 `def hx711.HX711.read_median ( self, times = 3 )`
- 6.3.2.33 `def hx711.HX711.read_np_arr8 ( self )`
- 6.3.2.34 `def hx711.HX711.read_np_arr8 ( self )`
- 6.3.2.35 `def hx711.HX711.reset ( self )`
- 6.3.2.36 `def hx711.HX711.reset ( self )`
- 6.3.2.37 `def hx711.HX711.set_gain ( self, gain )`
- 6.3.2.38 `def hx711.HX711.set_gain ( self, gain )`
- 6.3.2.39 `def hx711.HX711.set_offset ( self, offset )`
- 6.3.2.40 `def hx711.HX711.set_offset ( self, offset )`
- 6.3.2.41 `def hx711.HX711.set_offset_A ( self, offset )`
- 6.3.2.42 `def hx711.HX711.set_offset_A ( self, offset )`
- 6.3.2.43 `def hx711.HX711.set_offset_B ( self, offset )`
- 6.3.2.44 `def hx711.HX711.set_offset_B ( self, offset )`
- 6.3.2.45 `def hx711.HX711.set_reading_format ( self, byte_format = "LSB", bit_format = "MSB" )`
- 6.3.2.46 `def hx711.HX711.set_reading_format ( self, byte_format = "LSB", bit_format = "MSB" )`
- 6.3.2.47 `def hx711.HX711.set_reference_unit ( self, reference_unit )`
- 6.3.2.48 `def hx711.HX711.set_reference_unit ( self, reference_unit )`
- 6.3.2.49 `def hx711.HX711.set_reference_unit_A ( self, reference_unit )`
- 6.3.2.50 `def hx711.HX711.set_reference_unit_A ( self, reference_unit )`

6.3.2.51 `def hx711.HX711.set_reference_unit_B ( self, reference_unit )`

6.3.2.52 `def hx711.HX711.set_reference_unit_B ( self, reference_unit )`

6.3.2.53 `def hx711.HX711.tare ( self, times = 15 )`

6.3.2.54 `def hx711.HX711.tare ( self, times = 15 )`

6.3.2.55 `def hx711.HX711.tare_A ( self, times = 15 )`

6.3.2.56 `def hx711.HX711.tare_A ( self, times = 15 )`

6.3.2.57 `def hx711.HX711.tare_B ( self, times = 15 )`

6.3.2.58 `def hx711.HX711.tare_B ( self, times = 15 )`

### 6.3.3 Member Data Documentation

6.3.3.1 `hx711.HX711.bit_format`

6.3.3.2 `hx711.HX711.bit_range_values`

6.3.3.3 `hx711.HX711.byte_format`

6.3.3.4 `hx711.HX711.byte_range_values`

6.3.3.5 `hx711.HX711.DOUT`

6.3.3.6 `hx711.HX711.GAIN`

6.3.3.7 `hx711.HX711.isNegative`

6.3.3.8 `hx711.HX711.lastVal`

6.3.3.9 `hx711.HX711.LSBit`

6.3.3.10 `hx711.HX711.LSByte`

6.3.3.11 `hx711.HX711.MSBitIndex24Bit`

6.3.3.12 `hx711.HX711.MSBitIndex32Bit`

6.3.3.13 `hx711.HX711.MSBit`

6.3.3.14 `hx711.HX711.MSByte`

6.3.3.15 `hx711.HX711.OFFSET`

6.3.3.16 `hx711.HX711.OFFSET_B`

6.3.3.17 `hx711.HX711.PD_SCK`

6.3.3.18 `hx711.HX711.REFERENCE_UNIT`

6.3.3.19 `hx711.HX711.REFERENCE_UNIT_B`

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

- `/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Scale_Code/hx711.py`

## Chapter 7

# File Documentation

### 7.1 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/conversion↵ \_func.py File Reference

#### Namespaces

- [conversion\\_func](#)

#### Functions

- def [conversion\\_func.OunceToGram](#) (ounce)
- def [conversion\\_func.GramToOunce](#) (gram)
- def [conversion\\_func.KgToGram](#) (kilogram)
- def [conversion\\_func.GramToKg](#) (gram)
- def [conversion\\_func.PoundToGram](#) (Pound)
- def [conversion\\_func.PoundGramTo](#) (gram)
- def [conversion\\_func.TspToCup](#) (tsp)
- def [conversion\\_func.CupToTsp](#) (Cup)
- def [conversion\\_func.GramToTsp](#) (cursor, gram, name)
- def [conversion\\_func.TspToGram](#) (cursor, tsp, name)

### 7.2 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/database↵ \_func.py File Reference

#### Namespaces

- [database\\_func](#)

#### Functions

- def [database\\_func.printSortedTable](#) (cursor, tableName, sortBy)
- def [database\\_func.printTable](#) (cursor, tableName)
- def [database\\_func.printTuple](#) (cursor, tableName, attribute, field)
- def [database\\_func.retrieveTuple](#) (cursor, tableName, attribute, field)
- def [database\\_func.insertNewRecipe](#) (cursor, name, ingredient1, ingredient2, ingredient3, ingredient4, ingredient5, ingredient6, ingredient7, amount1, amount2, amount3, amount4, amount5, amount6, amount7)
- def [database\\_func.insertNewSpice](#) (cursor, name, gpt, available)
- def [database\\_func.deleteTuple](#) (cursor, tableName, attribute, field)

### 7.3 `/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/database_header.py` File Reference ↩↪

#### Namespaces

- `database_header`

#### Variables

- string `database_header.deleteRecipeTable` = 'DROP TABLE IF EXISTS 'Recipe';'
- string `database_header.deleteSpiceTable` = 'DROP TABLE IF EXISTS 'Spice';'
- string `database_header.createSpiceTable`
- string `database_header.createRecipeTable`

### 7.4 `/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/database_main.py` File Reference ↩↪

#### Namespaces

- `database_main`

#### Variables

- `database_main.conn` = `sqlite3.connect('BurrGrinder.db')`
- `database_main.c` = `conn.cursor()`
- list `database_main.spices`

### 7.5 `/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/sqlite_ex.py` File Reference ↩↪

#### Namespaces

- `sqlite_ex`

#### Variables

- `sqlite_ex.conn` = `sqlite3.connect('example.db')`
- `sqlite_ex.c` = `conn.cursor()`
- tuple `sqlite_ex.t` = ('RHAT',)



## 7.6 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/app.py File Reference

### Classes

- class [app.grindThread](#)

### Namespaces

- [app](#)

### Functions

- def [app.toggle\\_motor](#) (toggle, motor)
- def [app.grindSpice](#) (motor, amount)
- def [app.hello](#) (kwargs)
- def [app.toggle](#) (kwargs)
- def [app.grindSpices](#) (kwargs)
- def [app.application](#) (request)
- def [app.cleanAndExit](#) ()

### Variables

- int [app.Motor1](#) = 2  
*THESE LINE WILL NEED TO BE ADDED TO THE MAIN CODE TO SETUP GPIO PINS#####.*
- list [app.Scale1](#) = [15,17]
- int [app.Motor2](#) = 3
- list [app.Scale2](#) = [18, 27]
- int [app.Motor3](#) = 4
- list [app.Scale3](#) = [22, 23]
- int [app.Motor4](#) = 5
- list [app.Scale4](#) = [24, 25]
- int [app.Motor5](#) = 6
- list [app.Scale5](#) = [19, 16]
- int [app.Motor6](#) = 7
- list [app.Scale6](#) = [26, 20]
- int [app.Sleep](#) = 12
- int [app.Step](#) = 13
- int [app.Dir](#) = 14
- int [app.CW](#) = 1
- int [app.CCW](#) = 0
- tuple [app.Outs](#) = (Motor1,Motor2,Motor3,Motor4,Motor5,Motor6,Sleep,Step,Dir,Scale1[1],Scale2[1],Scale3[1],Scale4[1],Scale5[1],Scale6[1])
- tuple [app.Ins](#) = (Scale1[0],Scale2[0],Scale3[0],Scale4[0],Scale5[0],Scale6[0])
- dictionary [app.motorDict](#)

## 7.7 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/buttonPressed.py File Reference

### Namespaces

- [buttonPressed](#)

## Variables

- `buttonPressed.button` = `Button(18)`

## 7.8 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-↵ Code/src/motors.py File Reference

## Namespaces

- `motors`

## Functions

- `def motors.motor` (toggle, motor)
- `def motors.steppermotor` (rotation, direction)

## Variables

- `int motors.Motor1` = 2  
*THESE LINE WILL NEED TO BE ADDED TO THE MAIN CODE TO SETUP GPIO PINS#####.*
- `int motors.Motor2` = 3
- `int motors.Motor3` = 4
- `int motors.Motor4` = 5
- `int motors.Motor5` = 6
- `int motors.Motor6` = 7
- `int motors.Sleep` = 12
- `int motors.Step` = 13
- `int motors.Dir` = 14
- `int motors.CW` = 1
- `int motors.CCW` = 0
- `tuple motors.Outs` = (Motor1, Motor2, Motor3, Motor4, Motor5, Motor6, Sleep, Step, Dir)

## 7.9 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-↵ Code/src/server.py File Reference

## Classes

- `class server.grindThread`

## Namespaces

- `server`

## Functions

- def `server.grind_spices` ()
- def `server.toggle_motor` (toggle, motor)
- def `server.grindSpice` (motor, amount)

## Variables

- `server.app` = Flask(\_\_name\_\_)
- `server.methods`
- int `server.Motor1` = 2
- *THESE LINE WILL NEED TO BE ADDED TO THE MAIN CODE TO SETUP GPIO PINS#####.*
- list `server.Scale1` = [15,17]
- int `server.Motor2` = 3
- list `server.Scale2` = [18, 27]
- int `server.Motor3` = 4
- list `server.Scale3` = [22, 23]
- int `server.Motor4` = 5
- list `server.Scale4` = [24, 25]
- int `server.Motor5` = 6
- list `server.Scale5` = [19, 16]
- int `server.Motor6` = 7
- list `server.Scale6` = [26, 20]
- int `server.Sleep` = 12
- int `server.Step` = 13
- int `server.Dir` = 14
- int `server.CW` = 1
- int `server.CCW` = 0
- tuple `server.Outs` = (Motor1,Motor2,Motor3,Motor4,Motor5,Motor6,Sleep,Step,Dir,Scale1[1],Scale2[1],Scale3[1],Scale4[1],Scale5[1],Scale6[1])
- tuple `server.Ins` = (Scale1[0],Scale2[0],Scale3[0],Scale4[0],Scale5[0],Scale6[0])
- dictionary `server.motorDict`
- `server.host`
- `server.debug`

## 7.10 /home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Scale\_Code/example.py File Reference ↩

## Namespaces

- `example`

## Functions

- def `example.cleanAndExit` ()

## Variables

- `example.hx` = HX711(22, 23)
- `example.val` = hx.get\_weight(5)

## 7.11 `/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/example.py` File Reference ↩

### Namespaces

- [example](#)

### Functions

- `def example.print\_time (threadName, delay)`

## 7.12 `/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Scale_Code/hx711.py` File Reference ↩

### Classes

- class [hx711.HX711](#)

### Namespaces

- [hx711](#)

## 7.13 `/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/hx711.py` File Reference ↩

### Classes

- class [hx711.HX711](#)

### Namespaces

- [hx711](#)

# Index

/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/app.py, [29](#)

/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/buttonPressed.py, [29](#)

/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/example.py, [32](#)

/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/hx711.py, [32](#)

/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/motors.py, [30](#)

/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Motor-Code/src/server.py, [30](#)

/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Scale\_Code/example.py, [31](#)

/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/Scale\_Code/hx711.py, [32](#)

/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/conversion\_func.py, [27](#)

/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/database\_func.py, [27](#)

/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/database\_header.py, [28](#)

/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/database\_main.py, [28](#)

/home/uvnahn21/Desktop/UTA/Fall 2018/cse 4316-005/burr grinder code/db/sqlite\_ex.py, [28](#)

\_\_init\_\_  
  hx711::HX711, [23](#)  
  server::grindThread, [21](#)

amount  
  server::grindThread, [21](#)

app, [9](#)  
  application, [10](#)  
  CCW, [10](#)  
  cleanAndExit, [10](#)  
  CW, [10](#)  
  Dir, [10](#)  
  grindSpice, [10](#)  
  grindSpices, [10](#)  
  hello, [10](#)  
  Ins, [10](#)  
  Motor1, [10](#)  
  Motor2, [10](#)  
  Motor3, [10](#)  
  Motor4, [10](#)  
  Motor5, [10](#)  
  Motor6, [10](#)  
  motorDict, [10](#)  
  Outs, [11](#)  
  Scale1, [11](#)  
  Scale2, [11](#)  
  Scale3, [11](#)  
  Scale4, [11](#)  
  Scale5, [11](#)  
  Scale6, [11](#)  
  server, [17](#)  
  Sleep, [11](#)  
  Step, [11](#)  
  toggle, [10](#)  
  toggle\_motor, [10](#)  
  app.grindThread, [21](#)  
  application  
    app, [10](#)

bit\_format  
  hx711::HX711, [26](#)

bit\_range\_values  
  hx711::HX711, [26](#)

button  
  buttonPressed, [12](#)

buttonPressed, [11](#)  
  button, [12](#)

byte\_format  
  hx711::HX711, [26](#)

byte\_range\_values  
  hx711::HX711, [26](#)

c  
  database\_main, [14](#)  
  sqlite\_ex, [19](#)

CCW  
  app, [10](#)  
  motors, [16](#)  
  server, [17](#)

cleanAndExit  
  app, [10](#)  
  example, [15](#)

- conn
  - database\_main, 14
  - sqlite\_ex, 19
- conversion\_func, 12
  - CupToTsp, 12
  - GramToKg, 12
  - GramToOunce, 12
  - GramToTsp, 12
  - KgToGram, 12
  - OunceToGram, 12
  - PoundGramTo, 12
  - PoundToGram, 12
  - TspToCup, 12
  - TspToGram, 12
- createRecipeTable
  - database\_header, 13
- createSpiceTable
  - database\_header, 13
- CupToTsp
  - conversion\_func, 12
- CW
  - app, 10
  - motors, 16
  - server, 17
- DOUT
  - hx711::HX711, 26
- database\_func, 12
  - deleteTuple, 13
  - insertNewRecipe, 13
  - insertNewSpice, 13
  - printSortedTable, 13
  - printTable, 13
  - printTuple, 13
  - retrieveTuple, 13
- database\_header, 13
  - createRecipeTable, 13
  - createSpiceTable, 13
  - deleteRecipeTable, 13
  - deleteSpiceTable, 14
- database\_main, 14
  - c, 14
  - conn, 14
  - spices, 14
- debug
  - server, 17
- deleteRecipeTable
  - database\_header, 13
- deleteSpiceTable
  - database\_header, 14
- deleteTuple
  - database\_func, 13
- Dir
  - app, 10
  - motors, 16
  - server, 17
- example, 14
  - cleanAndExit, 15
  - hx, 15
  - print\_time, 15
  - val, 15
- GAIN
  - hx711::HX711, 26
- get\_binary\_string
  - hx711::HX711, 23
- get\_gain
  - hx711::HX711, 23
- get\_np\_arr8\_string
  - hx711::HX711, 23, 24
- get\_value
  - hx711::HX711, 24
- get\_value\_A
  - hx711::HX711, 24
- get\_value\_B
  - hx711::HX711, 24
- get\_weight
  - hx711::HX711, 24
- get\_weight\_A
  - hx711::HX711, 24
- get\_weight\_B
  - hx711::HX711, 24
- GramToKg
  - conversion\_func, 12
- GramToOunce
  - conversion\_func, 12
- GramToTsp
  - conversion\_func, 12
- grind\_spices
  - server, 17
- grindSpice
  - app, 10
  - server, 17
- grindSpices
  - app, 10
- hello
  - app, 10
- host
  - server, 17
- hx
  - example, 15
- hx711, 15
- hx711.HX711, 22
- hx711::HX711
  - \_\_init\_\_, 23
  - bit\_format, 26
  - bit\_range\_values, 26
  - byte\_format, 26
  - byte\_range\_values, 26
  - DOUT, 26
  - GAIN, 26
  - get\_binary\_string, 23
  - get\_gain, 23
  - get\_np\_arr8\_string, 23, 24
  - get\_value, 24
  - get\_value\_A, 24

- get\_value\_B, [24](#)
- get\_weight, [24](#)
- get\_weight\_A, [24](#)
- get\_weight\_B, [24](#)
- is\_ready, [24](#)
- isNegative, [26](#)
- LSBit, [26](#)
- LSByte, [26](#)
- lastVal, [26](#)
- MSBIndex24Bit, [26](#)
- MSBIndex32Bit, [26](#)
- MSBit, [26](#)
- MSByte, [26](#)
- OFFSET\_B, [26](#)
- OFFSET, [26](#)
- PD\_SCK, [26](#)
- power\_down, [24](#)
- power\_up, [24](#)
- REFERENCE\_UNIT\_B, [26](#)
- REFERENCE\_UNIT, [26](#)
- read, [24](#)
- read\_average, [24](#)
- read\_long, [25](#)
- read\_median, [25](#)
- read\_np\_arr8, [25](#)
- reset, [25](#)
- set\_gain, [25](#)
- set\_offset, [25](#)
- set\_offset\_A, [25](#)
- set\_offset\_B, [25](#)
- set\_reading\_format, [25](#)
- set\_reference\_unit, [25](#)
- set\_reference\_unit\_A, [25](#)
- set\_reference\_unit\_B, [25](#), [26](#)
- tare, [26](#)
- tare\_A, [26](#)
- tare\_B, [26](#)
- Ins
  - app, [10](#)
  - server, [17](#)
- insertNewRecipe
  - database\_func, [13](#)
- insertNewSpice
  - database\_func, [13](#)
- is\_ready
  - hx711::HX711, [24](#)
- isNegative
  - hx711::HX711, [26](#)
- KgToGram
  - conversion\_func, [12](#)
- LSBit
  - hx711::HX711, [26](#)
- LSByte
  - hx711::HX711, [26](#)
- lastVal
  - hx711::HX711, [26](#)
- MSBIndex24Bit
  - hx711::HX711, [26](#)
- MSBIndex32Bit
  - hx711::HX711, [26](#)
- MSBit
  - hx711::HX711, [26](#)
- MSByte
  - hx711::HX711, [26](#)
- methods
  - server, [17](#)
- motor
  - motors, [16](#)
  - server::grindThread, [21](#)
- Motor1
  - app, [10](#)
  - motors, [16](#)
  - server, [17](#)
- Motor2
  - app, [10](#)
  - motors, [16](#)
  - server, [17](#)
- Motor3
  - app, [10](#)
  - motors, [16](#)
  - server, [18](#)
- Motor4
  - app, [10](#)
  - motors, [16](#)
  - server, [18](#)
- Motor5
  - app, [10](#)
  - motors, [16](#)
  - server, [18](#)
- Motor6
  - app, [10](#)
  - motors, [16](#)
  - server, [18](#)
- motorDict
  - app, [10](#)
  - server, [18](#)
- motors, [15](#)
  - CCW, [16](#)
  - CW, [16](#)
  - Dir, [16](#)
  - motor, [16](#)
  - Motor1, [16](#)
  - Motor2, [16](#)
  - Motor3, [16](#)
  - Motor4, [16](#)
  - Motor5, [16](#)
  - Motor6, [16](#)
  - Outs, [16](#)
  - Sleep, [16](#)
  - Step, [16](#)
  - steppermotor, [16](#)
- OFFSET\_B
  - hx711::HX711, [26](#)
- OFFSET

- hx711::HX711, 26
- OunceToGram
  - conversion\_func, 12
- Outs
  - app, 11
  - motors, 16
  - server, 18
- PD\_SCK
  - hx711::HX711, 26
- PoundGramTo
  - conversion\_func, 12
- PoundToGram
  - conversion\_func, 12
- power\_down
  - hx711::HX711, 24
- power\_up
  - hx711::HX711, 24
- print\_time
  - example, 15
- printSortedTable
  - database\_func, 13
- printTable
  - database\_func, 13
- printTuple
  - database\_func, 13
- REFERENCE\_UNIT\_B
  - hx711::HX711, 26
- REFERENCE\_UNIT
  - hx711::HX711, 26
- read
  - hx711::HX711, 24
- read\_average
  - hx711::HX711, 24
- read\_long
  - hx711::HX711, 25
- read\_median
  - hx711::HX711, 25
- read\_np\_arr8
  - hx711::HX711, 25
- reset
  - hx711::HX711, 25
- retrieveTuple
  - database\_func, 13
- run
  - server::grindThread, 21
- Scale1
  - app, 11
  - server, 18
- Scale2
  - app, 11
  - server, 18
- Scale3
  - app, 11
  - server, 18
- Scale4
  - app, 11
- server, 18
- Scale5
  - app, 11
  - server, 18
- Scale6
  - app, 11
  - server, 18
- server, 16
  - app, 17
  - CCW, 17
  - CW, 17
  - debug, 17
  - Dir, 17
  - grind\_spices, 17
  - grindSpice, 17
  - host, 17
  - Ins, 17
  - methods, 17
  - Motor1, 17
  - Motor2, 17
  - Motor3, 18
  - Motor4, 18
  - Motor5, 18
  - Motor6, 18
  - motorDict, 18
  - Outs, 18
  - Scale1, 18
  - Scale2, 18
  - Scale3, 18
  - Scale4, 18
  - Scale5, 18
  - Scale6, 18
  - Sleep, 18
  - Step, 19
  - toggle\_motor, 17
- server.grindThread, 21
- server::grindThread
  - \_\_init\_\_, 21
  - amount, 21
  - motor, 21
  - run, 21
- set\_gain
  - hx711::HX711, 25
- set\_offset
  - hx711::HX711, 25
- set\_offset\_A
  - hx711::HX711, 25
- set\_offset\_B
  - hx711::HX711, 25
- set\_reading\_format
  - hx711::HX711, 25
- set\_reference\_unit
  - hx711::HX711, 25
- set\_reference\_unit\_A
  - hx711::HX711, 25
- set\_reference\_unit\_B
  - hx711::HX711, 25, 26
- Sleep



- app, [11](#)
  - motors, [16](#)
  - server, [18](#)
- spices
  - database\_main, [14](#)
- sqlite\_ex, [19](#)
  - c, [19](#)
  - conn, [19](#)
  - t, [19](#)
- Step
  - app, [11](#)
  - motors, [16](#)
  - server, [19](#)
- steppermotor
  - motors, [16](#)
- t
  - sqlite\_ex, [19](#)
- tare
  - hx711::HX711, [26](#)
- tare\_A
  - hx711::HX711, [26](#)
- tare\_B
  - hx711::HX711, [26](#)
- toggle
  - app, [10](#)
- toggle\_motor
  - app, [10](#)
  - server, [17](#)
- TspToCup
  - conversion\_func, [12](#)
- TspToGram
  - conversion\_func, [12](#)
- val
  - example, [15](#)