Pool Club Management Software – API Documentation

Contents

[Common Elements amongst API Files 2](#_Toc151732995)

[Imports 2](#_Toc151732996)

[Variables 2](#_Toc151732997)

[login\_api.py 3](#_Toc151732998)

[Variables 3](#_Toc151732999)

[staff\_api.py 3](#_Toc151733000)

[Functions 3](#_Toc151733001)

[Variables 4](#_Toc151733002)

[maintenance\_api.py 4](#_Toc151733003)

[Functions 4](#_Toc151733004)

[Variables 4](#_Toc151733005)

[staff\_schedule\_api.py 4](#_Toc151733006)

[Functions 4](#_Toc151733007)

[Variables 4](#_Toc151733008)

[day\_read\_api.py 4](#_Toc151733009)

[Functions 4](#_Toc151733010)

[Variables 5](#_Toc151733011)

[week\_read\_api.py 5](#_Toc151733012)

[Functions 5](#_Toc151733013)

[Variables 5](#_Toc151733014)

[poolchem\_supp\_api.py 5](#_Toc151733015)

[Functions 5](#_Toc151733016)

[Variables 5](#_Toc151733017)

[daily\_report.py 5](#_Toc151733018)

[Functions 5](#_Toc151733019)

[Variables 6](#_Toc151733020)

[weekly\_report.py 7](#_Toc151733021)

[Functions 7](#_Toc151733022)

[Variables 7](#_Toc151733023)

[Common Elements amongst GUI files 9](#_Toc151733024)

[Imports 9](#_Toc151733025)

[Variables 9](#_Toc151733026)

[poolclub\_staff\_gui.py 10](#_Toc151733027)

[Functions 10](#_Toc151733028)

[Variables 10](#_Toc151733029)

[day\_read\_gui.py 14](#_Toc151733030)

[Functions 14](#_Toc151733031)

[Variables 14](#_Toc151733032)

[week\_read\_gui.py 18](#_Toc151733033)

[Functions 18](#_Toc151733034)

[Variables 18](#_Toc151733035)

[maintenance.py 22](#_Toc151733036)

[Functions 22](#_Toc151733037)

[Variables 22](#_Toc151733038)

[poolchem\_supply\_gui.py 25](#_Toc151733039)

[Functions 25](#_Toc151733040)

[Variables 26](#_Toc151733041)

[staff\_schedule\_gui.py 29](#_Toc151733042)

[Functions 29](#_Toc151733043)

[Variables 30](#_Toc151733044)

# Common Elements amongst API Files

## Imports

|  |  |
| --- | --- |
| Import Statement | Description/Purpose |
| mysql.connector | The library that connects Python to MySQL. |
| from tkinter import \* |  |
| from tkinter import ttk | Grants access to a set of Tk widgets |
| from tkinter.ttk import \* |  |
| from tkinter import messagebox | Imports the messagebox module from the tkinter library. |
| from fpdf import FPDF | Imports a class that allows the program to generate a PDF file. |

## Variables

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Variable | Variable Type | Description | Files included in |
| mydb | PooledMySQLConnection | MySQLConnection | Any | The command that allows the database to be reached. | All |
| host | Any | The variable that defines the host server. | All |
| user | Any | The variable that defines the user of the database | All |
| password | Any | The variable that defines the password used for the database. | All |
| database | Any | The name of the database schema. | All |
| mycursor | Any | MySQLCursor | The variable that defines the cursor that works to execute database commands. | All |
| inputs | Any | The input variable for the tuple that reads the desired field values. | staff\_api.py maintenance\_api.py staff\_schedule\_api.py day\_read\_api.py week\_read\_api.py poolchem\_supp\_api.oy |
| recordInput | list[str] | An array that passes in the appropriate variables to their corresponding MySQL records. | staff\_api.py maintenance\_api.py staff\_schedule\_api.py day\_read\_api.py week\_read\_api.py poolchem\_supp\_api.oy |
| sql | Literal['…'] | Passes MySQL commands to the database through Python’s connector. | staff\_api.py maintenance\_api.py staff\_schedule\_api.py day\_read\_api.py week\_read\_api.py poolchem\_supp\_api.oy |
| myresult | Any | List[RowType] | A variable that utilizes mycursor in order to fetch the appropriate amount of records for a for loop. | daily\_report.py weekly\_report.py |
| x | Any | RowType | A counter value used in a for loop in conjunction with myresult to display the records. | daily\_report.py weekly\_report.py |
| val | tuple[str] | A variable that specifies the value(s) to be passed to MySQL in conjunction with the sql variable. | All |

# login\_api.py

## Variables

|  |  |  |
| --- | --- | --- |
| Name of Variable | Variable Type | Description |
| username | str | The variable that the user inputs their username in. |
| password | str | The user’s password for their account. |

# staff\_api.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| addStaffMember(inputs) | Add a staff member to the database. |
| deleteStaffMember(inputs) | Removes a staff member from the database. |
| updateStaffMember(inputs) | Updates an existing staff member in the database. |
| displayStaffMember() | Displays existing staff members in the database based on specifications. |

## Variables

Same as shared variables above.

# maintenance\_api.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| addMaintenanceTask(inputs) | Add a maintenance task to the database. |
| deleteMaintenanceTask(inputs) | Removes a maintenance task from the database. |
| updateMaintenaceTask(inputs) | Updates a maintenance task from the database. |
| displayMaintenanceTask() | Displays existing scheduled tasks in the database based on specifications. |

## Variables

Same as shared variables above.

# staff\_schedule\_api.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| addScheduledTask(inputs) | Add a scheduled task to the database. |
| deleteScheduledTask(inputs) | Removes a scheduled task from the database. |
| updateScheduledTask(inputs) | Updates an existing scheduled task in the database. |
| displayScheduledTask() | Displays existing scheduled tasks in the database based on specifications. |

## Variables

Same as shared variables above.

# day\_read\_api.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| addDailyReading(inputs) | Add a daily reading record to the database. |
| deleteDailyReading(inputs) | Removes a daily reading record from the database. |
| updateDailyReading(inputs) | Updates an existing daily reading record in the database. |
| displayDailyReading() | Displays existing daily reading records in the database based on specifications. |

## Variables

Same as shared variables above.

# week\_read\_api.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| addWeeklyReading(inputs) | Add a daily reading record to the database. |
| deleteWeeklyReading(inputs) | Removes a daily reading record from the database. |
| updateWeeklyReading(inputs) | Updates an existing daily reading record in the database. |
| displayWeeklyReading() | Displays existing daily reading records in the database based on specifications. |

## Variables

Same as shared variables above.

# poolchem\_supp\_api.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| addPoolChemical(inputs) | Add a daily reading record to the database. |
| removePoolChemical(inputs) | Removes a daily reading record from the database. |
| updatePoolChemical(inputs) | Updates an existing daily reading record in the database. |
| displayPoolChemical() | Displays existing daily reading records in the database based on specifications. |

## Variables

Same as shared variables above.

# daily\_report.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| generateDailyReport(date) | Generates the daily report with the date passed in from the day\_read\_gui.py file. |

## Variables

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Variable | Variable Type | Description | Functions present in |
| mydb | PooledMySQLConnection | MySQLConnection | Any | The command that allows the database to be reached. | All |
| host | Any | The variable that defines the host server. | All |
| user | Any | The variable that defines the user of the database | All |
| password | Any | The variable that defines the password used for the database. | All |
| database | Any | The name of the database schema. | All |
| mycursor | Any | MySQLCursor | The variable that defines the cursor that works to execute database commands. | All |
| date | Any | The input variable used for the date of the record as specified by day\_read\_gui.py. | generateDailyReport(date) |
| title | str | The variable that defines the generated title of the document. | generateDailyReport(date) |
| pdf | pdf | A variable used to help specify the parameters of a generated PDF file. | generateDailyReport(date) |
| sql | Literal['…'] | Passes MySQL commands to the database through Python’s connector. | generateDailyReport(date) |
| myresult | Any | List[RowType] | A variable that utilizes mycursor in order to fetch the appropriate amount of records for a for loop. | generateDailyReport(date) |
| x | Any | RowType | A counter value used in a for loop in conjunction with myresult to display the records. | generateDailyReport(date) |
| val | tuple[str] | A variable that specifies the value(s) to be passed to MySQL in conjunction with the sql variable. | generateDailyReport(date) |

# weekly\_report.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| generateWeeklyReport(date) | Generates the daily report with the date passed in from the week\_read\_gui.py file. |

## Variables

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Variable | Variable Type | Description | Functions present in |
| mydb | PooledMySQLConnection | MySQLConnection | Any | The command that allows the database to be reached. | All |
| host | Any | The variable that defines the host server. | All |
| user | Any | The variable that defines the user of the database | All |
| password | Any | The variable that defines the password used for the database. | All |
| database | Any | The name of the database schema. | All |
| mycursor | Any | MySQLCursor | The variable that defines the cursor that works to execute database commands. | All |
| date | Any | The input variable used for the date of the record as specified by day\_read\_gui.py. | generateDailyReport(date) |
| title | str | The variable that defines the generated title of the document. | generateDailyReport(date) |
| pdf | pdf | A variable used to help specify the parameters of a generated PDF file. | generateDailyReport(date) |
| sql | Literal['…'] | Passes MySQL commands to the database through Python’s connector. | generateDailyReport(date) |
| myresult | Any | List[RowType] | A variable that utilizes mycursor in order to fetch the appropriate amount of records for a for loop. | generateDailyReport(date) |
| x | Any | RowType | A counter value used in a for loop in conjunction with myresult to display the records. | generateDailyReport(date) |
| val | tuple[str] | A variable that specifies the value(s) to be passed to MySQL in conjunction with the sql variable. | generateDailyReport(date) |

# Common Elements amongst GUI files

## Imports

|  |  |
| --- | --- |
| Import Statement | Description/Purpose |
| mysql.connector | The library that connects Python to MySQL. |
| from tkinter import \* |  |
| from tkinter import ttk | Grants access to a set of Tk widgets |
| from tkinter.ttk import \* |  |
| from tkinter import messagebox | Imports the messagebox module from the tkinter library. |

## Variables

|  |  |
| --- | --- |
| Function Name | Description |
| clear\_entries() | Clears the entry fields. |
| select\_record() | Autofills the entry fields when clicking on a record displayed in the treeview. |
| generateTree() | Generates the treeview of a table from the database |
| addRecord() | Passes on the values from the entry fields to the API in order to add the record to the database. |
| removeRecord() | Passes on the values from the entry fields to the API in order to remove matching records from the database. |
| updateRecord() | Passes on the values from the entry fields to the API in order to update the record of the specified ID value in the database. |
| addRecord() | Passes on the values from the entry fields to the API in order to add the record to the database. |

# poolclub\_staff\_gui.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| clear\_entries() | Clears the entry fields. |
| select\_record() | Autofills the entry fields when clicking on a record displayed in the treeview. |
| generateTree() | Generates the treeview of the staff table. |
| addRecord() | Passes on the values from the entry fields to staff\_api.py in order to add the record to the database. |
| removeRecord() | Passes on the values from the entry fields to staff\_api.py in order to remove matching records from the database. |
| updateRecord() | Passes on the values from the entry fields to staff\_api.py in order to update the record of the specified ID value in the database. |

## Variables

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Variable | Variable Type | Description | Functions present in |
| mydb | PooledMySQLConnection | MySQLConnection | Any | The command that allows the database to be reached. | display() |
| host | Any | The variable that defines the host server. | display() |
| user | Any | The variable that defines the user of the database. | display() |
| password | Any | The variable that defines the password used for the database. | display() |
| database | Any | The name of the database schema. | display() |
| mycursor | Any | MySQLCursor | The variable that defines the cursor that works to execute database commands. | display()  generateTree() |
| selected | str | Allows the program to select values. | select\_record() |
| values[] | Tuple | Specifies that a set of values needs to be grabbed. | select\_record() |
| myresult | Any | List[RowType] | A variable that utilizes mycursor in order to fetch the appropriate amount of records for a for loop. | generateTree() |
| x | Any | RowType | A counter value used in a for loop in conjunction with myresult to display the records. |  |
| results | list[str] | Grabs the values from the entry fields to pass into the week\_read\_api.py module. | addRecord()  removeRecord()  updateRecord() |
| label | Label | Used to define the header. | display() |
| style | Style | Used to help define what different components of the GUI look like. | display() |
| pandedwindow | PandedWindow | Used to call upon the Ttk PandedWindow widget, which “displays a number of subwindows, stacked either vertically or horizontally.” In this context, it was used as a divider between the header and the rest of the interface. | display() |
| treeframe | Frame | Sets up the frame for the treeview of the staff MySQL table. | display() |
| scrollbar | Scrollbar | Defines the scrollbar to be attached to the treeview. | display() |
| staff\_tree | Treeview | Defines the treeview of the staff MySQL table. | display() |
| staff\_tree['columns'] | weekly\_tree: Treeview | Defines an array of headers for the treeview of the staff database. | display() |
| sql | Literal['…'] | Passes MySQL commands to the database through Python’s connector. |  |
| input\_frame | Frame | Sets up the frame for the input fields and labels. | display() |
| id\_label | Label | The label for the ID Num entry field. | display() |
| id\_entry | Entry | The entry field for the value of ID Num. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| fn\_label | Label | The label for the First Name entry field. | display() |
| fn\_entry | Entry | The entry field for the value of First Name. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| ln\_label | Label | The label for the Last Name entry field. | display() |
| ln\_entry | Entry | The entry field for the value of Last Name. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| hd\_label | Label | The label for the Hire Date entry field. | display() |
| hd\_entry | Entry | The entry field for the value of Hire Date. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| sd\_label | Label | The label for the Separation Date entry field. | display() |
| sd\_entry | Entry | The entry field for the value of Separation Date. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| rfs\_label | Label | The label for Reason for Separation entry field. | display() |
| rfs\_entry | Entry | The entry field for the value of Reason for Separation. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| pswd\_label | Label | The label for the Password entry field. | display() |
| pswd\_entry | Entry | The entry field for the value of Pasword. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| ut\_label | Label | The label for the User Type entry field. | display() |
| ut\_entry | Entry | The entry field for the value of User Type. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| button\_frame | LabelFrame | Sets up the frame for the various button options. | display() |
| add\_button | Button | The button that calls the addRecord() function when clicked. | display() |
| remove\_button | Button | The button that calls the removeRecord() function when clicked. | display() |
| update\_button | Button | The button that calls the updateRecord() function when clicked. | display() |
| display\_button | Button | The button that calls the displayRecord() function when clicked. | display() |
| clear\_button | Button | The button that calls the clear\_entries() function when clicked. | display() |

# day\_read\_gui.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| clear\_entries() | Clears the entry fields. |
| select\_record() | Autofills the entry fields when clicking on a record displayed in the treeview. |
| generateTree() | Generates the treeview of the daily\_readings table. |
| addRecord() | Passes on the values from the entry fields to day\_read\_api.py in order to add the record to the database. |
| removeRecord() | Passes on the values from the entry fields to day\_read\_api.py in order to remove matching records from the database. |
| updateRecord() | Passes on the values from the entry fields to day\_read\_api.pu in order to update the record of the specified ID value in the database. |
| waterQualRegReport() | Passes on the value from d\_entry to daily\_report.py in order to generate a weekly readings water quality regulation report from the database. |

## Variables

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Variable | Variable Type | Description | Functions present in |
| mydb | PooledMySQLConnection | MySQLConnection | Any | The command that allows the database to be reached. | display() |
| host | Any | The variable that defines the host server. | display() |
| user | Any | The variable that defines the user of the database | display() |
| password | Any | The variable that defines the password used for the database. | display() |
| database | Any | The name of the database schema. | display() |
| mycursor | Any | MySQLCursor | The variable that defines the cursor that works to execute database commands. | display()  generateTree() |
| selected | str | Allows the program to select values. | select\_record() |
| values[] | Tuple | Specifies that a set of values needs to be grabbed. | select\_record() |
| myresult | Any | List[RowType] | A variable that utilizes mycursor in order to fetch the appropriate amount of records for a for loop. | generateTree() |
| x | Any | RowType | A counter value used in a for loop in conjunction with myresult to display the records. |  |
| results | list[str] | Grabs the values from the entry fields to pass into the week\_read\_api.py module. | addRecord()  removeRecord()  updateRecord() |
| report\_date | str | Grabs the value from the d\_entry field in order to pass it into the weekly\_report.py module. | waterQualRegReport() |
| label | Label | Used to define the header. | display() |
| style | Style | Used to help define what different components of the GUI look like. | display() |
| pandedwindow | PandedWindow | Used to call upon the Ttk PandedWindow widget, which “displays a number of subwindows, stacked either vertically or horizontally.” In this context, it was used as a divider between the header and the rest of the interface. | display() |
| treeframe | Frame | Sets up the frame for the treeview of the daily\_readings MySQL table. | display() |
| scrollbar | Scrollbar | Defines the scrollbar to be attached to the treeview. | display() |
| daily\_tree | Treeview | Defines the treeview of the daily\_readings MySQL table. | display() |
| daily\_tree['columns'] | weekly\_tree: Treeview | Defines an array of headers for the treeview of the daily\_readings database. | display() |
| sql | Literal['…'] | Passes MySQL commands to the database through Python’s connector. |  |
| input\_frame | Frame | Sets up the frame for the input fields and labels. | display() |
| id\_label | Label | The label for the Day ID entry field. | display() |
| id\_entry | Entry | The entry field for the value of Day ID. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| d\_label | Label | The label for the Date entry field. | display() |
| d\_entry | Entry | The entry field for the value of Date. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| t\_label | Label | The label for the Time entry field. | display() |
| t\_entry | Entry | The entry field for the value of Time. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| ph\_label | Label | The label for the pH entry field. | display() |
| ph\_entry | Entry | The entry field for the value of pH. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| tChl\_label | Label | The label for the Total Chlorine entry field. | display() |
| tChl\_entry | Entry | The entry field for the value of Total Chlorine. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| fChl\_label | Label | The label for the Free Chlorine entry field. | display() |
| fChl\_entry | Entry | The entry field for the value of Free Chlorine. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| cChl\_label | Label | The label for the Combined Chlorine entry field. | display() |
| cChl\_entry | Entry | The entry field for the value of Combined Chlorine. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| button\_frame | LabelFrame | Sets up the frame for the various button options. | display() |
| add\_button | Button | The button that calls the addRecord() function when clicked. | display() |
| remove\_button | Button | The button that calls the removeRecord() function when clicked. | display() |
| update\_button | Button | The button that calls the updateRecord() function when clicked. | display() |
| display\_button | Button | The button that calls the displayRecord() function when clicked. | display() |
| clear\_button | Button | The button that calls the clear\_entries() function when clicked. | display() |
| wqrr\_button | Button | The button that calls the waterQualRegReport() function when clicked. | display() |

# week\_read\_gui.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| clear\_entries() | Clears the entry fields. |
| select\_record() | Autofills the entry fields when clicking on a record displayed in the treeview. |
| generateTree() | Generates the treeview of the weekly\_readings table. |
| addRecord() | Passes on the values from the entry fields to week\_read\_api.py in order to add the record to the database. |
| removeRecord() | Passes on the values from the entry fields to week\_read\_api.py in order to remove matching records from the database. |
| updateRecord() | Passes on the values from the entry fields to week\_read\_api.py in order to update the record of the specified ID value in the database. |
| waterQualRegReport() | Passes on the value from d\_entry to weekly\_report.py in order to generate a weekly readings water quality regulation report from the database. |

## Variables

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Variable | Variable Type | Description | Functions present in |
| mydb | PooledMySQLConnection | MySQLConnection | Any | The command that allows the database to be reached. | display() |
| host | Any | The variable that defines the host server. | display() |
| user | Any | The variable that defines the user of the database | display() |
| password | Any | The variable that defines the password used for the database. | display() |
| database | Any | The name of the database schema. | display() |
| mycursor | Any | MySQLCursor | The variable that defines the cursor that works to execute database commands. | display()  generateTree() |
| selected | str | Allows the program to select values. | select\_record() |
| values[] | Tuple | Specifies that a set of values needs to be grabbed. | select\_record() |
| myresult | Any | List[RowType] | A variable that utilizes mycursor in order to fetch the appropriate amount of records for a for loop. | generateTree() |
| x | Any | RowType | A counter value used in a for loop in conjunction with myresult to display the records. |  |
| results | list[str] | Grabs the values from the entry fields to pass into the week\_read\_api.py module. | addRecord()  removeRecord()  updateRecord() |
| report\_date | str | Grabs the value from the d\_entry field in order to pass it into the weekly\_report.py module. | waterQualRegReport() |
| label | Label | Used to define the header. | display() |
| style | Style | Used to help define what different components of the GUI look like. | display() |
| pandedwindow | PandedWindow | Used to call upon the Ttk PandedWindow widget, which “displays a number of subwindows, stacked either vertically or horizontally.” In this context, it was used as a divider between the header and the rest of the interface. | display() |
| treeframe | Frame | Sets up the frame for the treeview of the weekly\_readings MySQL table. | display() |
| scrollbar | Scrollbar | Defines the scrollbar to be attached to the treeview. | display() |
| weekly\_tree | Treeview | Defines the treeview of the weekly\_readings MySQL table. | display() |
| weekly\_tree['columns'] | weekly\_tree: Treeview | Defines an array of headers for the treeview of the weekly\_readings database. | display() |
| sql | Literal['…'] | Passes MySQL commands to the database through Python’s connector. |  |
| input\_frame | Frame | Sets up the frame for the input fields and labels. | display() |
| id\_label | Label | The label for the Week ID entry field. | display() |
| id\_entry | Entry | The entry field for the value of Week ID. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| d\_label | Label | The label for the Date entry field. | display() |
| d\_entry | Entry | The entry field for the value of Date. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| t\_label | Label | The label for the Time entry field. | display() |
| t\_entry | Entry | The entry field for the value of Time. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| th\_label | Label | The label for the Total Hardness entry field. | display() |
| th\_entry | Entry | The entry field for the value of Total Hardness. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| ta\_label | Label | The label for the Total Alkalinity entry field. | display() |
| ta\_entry | Entry | The entry field for the value of Total Alkalinity. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| cya\_label | Label | The label for the CYA (stabilizer) entry field. | display() |
| cya\_entry | Entry | The entry field for the value of CYA (stabilizer). | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| button\_frame | LabelFrame | Sets up the frame for the various button options. | display() |
| add\_button | Button | The button that calls the addRecord() function when clicked. | display() |
| remove\_button | Button | The button that calls the removeRecord() function when clicked. | display() |
| update\_button | Button | The button that calls the updateRecord() function when clicked. | display() |
| display\_button | Button | The button that calls the displayRecord() function when clicked. | display() |
| clear\_button | Button | The button that calls the clear\_entries() function when clicked. | display() |
| wqrr\_button | Button | The button that calls the waterQualRegReport() function when clicked. | display() |

# maintenance.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| clear\_entries() | Clears the entry fields. |
| select\_record() | Autofills the entry fields when clicking on a record displayed in the treeview. |
| generateTree() | Generates the treeview of the maintenance. |
| addRecord() | Passes on the values from the entry fields to maintenance\_api.py in order to add the record to the database. |
| removeRecord() | Passes on the values from the entry fields to maintenance\_api.py in order to remove matching records from the database. |
| updateRecord() | Passes on the values from the entry fields to maintenance\_api.py in order to update the record of the specified ID value in the database. |

## Variables

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Variable | Variable Type | Description | Functions present in |
| mydb | PooledMySQLConnection | MySQLConnection | Any | The command that allows the database to be reached. | display() |
| host | Any | The variable that defines the host server. | display() |
| user | Any | The variable that defines the user of the database. | display() |
| password | Any | The variable that defines the password used for the database. | display() |
| database | Any | The name of the database schema. | display() |
| mycursor | Any | MySQLCursor | The variable that defines the cursor that works to execute database commands. | display()  generateTree() |
| selected | str | Allows the program to select values. | select\_record() |
| values[] | Tuple | Specifies that a set of values needs to be grabbed. | select\_record() |
| myresult | Any | List[RowType] | A variable that utilizes mycursor in order to fetch the appropriate amount of records for a for loop. | generateTree() |
| x | Any | RowType | A counter value used in a for loop in conjunction with myresult to display the records. |  |
| results | list[str] | Grabs the values from the entry fields to pass into the week\_read\_api.py module. | addRecord()  removeRecord()  updateRecord() |
| label | Label | Used to define the header. | display() |
| style | Style | Used to help define what different components of the GUI look like. | display() |
| pandedwindow | PandedWindow | Used to call upon the Ttk PandedWindow widget, which “displays a number of subwindows, stacked either vertically or horizontally.” In this context, it was used as a divider between the header and the rest of the interface. | display() |
| treeframe | Frame | Sets up the frame for the treeview of the maintenance MySQL table. | display() |
| scrollbar | Scrollbar | Defines the scrollbar to be attached to the treeview. | display() |
| main\_tree | Treeview | Defines the treeview of the maintenance MySQL table. | display() |
| main\_tree['columns'] | main\_tree: Treeview | Defines an array of headers for the treeview of the maintenance database. | display() |
| sql | Literal['…'] | Passes MySQL commands to the database through Python’s connector. |  |
| input\_frame | Frame | Sets up the frame for the input fields and labels. | display() |
| id\_label | Label | The label for the Item Number entry field. | display() |
| id\_entry | Entry | The entry field for the value of Item Number. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| d\_label | Label | The label for the Date entry field. | display() |
| d\_entry | Entry | The entry field for the value of Date. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| task\_label | Label | The label for the Task entry field. | display() |
| task\_entry | Entry | The entry field for the value of Task. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| sr\_label | Label | The label for the Staff Representative entry field. | display() |
| sr\_entry | Entry | The entry field for the value of Staff Representative. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| s\_label | Label | The label for the Status entry field. | display() |
| s\_entry | Entry | The entry field for the value of Status. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| button\_frame | LabelFrame | Sets up the frame for the various button options. | display() |
| add\_button | Button | The button that calls the addRecord() function when clicked. | display() |
| remove\_button | Button | The button that calls the removeRecord() function when clicked. | display() |
| update\_button | Button | The button that calls the updateRecord() function when clicked. | display() |
| display\_button | Button | The button that calls the displayRecord() function when clicked. | display() |
| clear\_button | Button | The button that calls the clear\_entries() function when clicked. | display() |

# poolchem\_supply\_gui.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| clear\_entries() | Clears the entry fields. |
| select\_record() | Autofills the entry fields when clicking on a record displayed in the treeview. |
| generateTree() | Generates the treeview of the pool\_chemical\_supply table. |
| addRecord() | Passes on the values from the entry fields to poolchem\_supp\_api.py in order to add the record to the database. |
| removeRecord() | Passes on the values from the entry fields to poolchem\_supp\_api.py in order to remove matching records from the database. |
| updateRecord() | Passes on the values from the entry fields to poolchem\_supp\_api.py in order to update the record of the specified ID value in the database. |

## Variables

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Variable | Variable Type | Description | Functions present in |
| mydb | PooledMySQLConnection | MySQLConnection | Any | The command that allows the database to be reached. | display() |
| host | Any | The variable that defines the host server. | display() |
| user | Any | The variable that defines the user of the database. | display() |
| password | Any | The variable that defines the password used for the database. | display() |
| database | Any | The name of the database schema. | display() |
| mycursor | Any | MySQLCursor | The variable that defines the cursor that works to execute database commands. | display()  generateTree() |
| selected | str | Allows the program to select values. | select\_record() |
| values[] | Tuple | Specifies that a set of values needs to be grabbed. | select\_record() |
| myresult | Any | List[RowType] | A variable that utilizes mycursor in order to fetch the appropriate amount of records for a for loop. | generateTree() |
| x | Any | RowType | A counter value used in a for loop in conjunction with myresult to display the records. |  |
| results | list[str] | Grabs the values from the entry fields to pass into the week\_read\_api.py module. | addRecord()  removeRecord()  updateRecord() |
| label | Label | Used to define the header. | display() |
| style | Style | Used to help define what different components of the GUI look like. | display() |
| pandedwindow | PandedWindow | Used to call upon the Ttk PandedWindow widget, which “displays a number of subwindows, stacked either vertically or horizontally.” In this context, it was used as a divider between the header and the rest of the interface. | display() |
| treeframe | Frame | Sets up the frame for the treeview of the pool\_chemical\_supply MySQL table. | display() |
| scrollbar | Scrollbar | Defines the scrollbar to be attached to the treeview. | display() |
| chem\_tree | Treeview | Defines the treeview of the pool\_chemical\_supply MySQL table. | display() |
| chem\_tree['columns'] | weekly\_tree: Treeview | Defines an array of headers for the treeview of the staff table. | display() |
| sql | Literal['…'] | Passes MySQL commands to the database through Python’s connector. |  |
| input\_frame | Frame | Sets up the frame for the input fields and labels. | display() |
| id\_label | Label | The label for the Chemical ID entry field. | display() |
| id\_entry | Entry | The entry field for the value of Chemical ID. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| cn\_label | Label | The label for the Chemical Nameentry field. | display() |
| cn\_entry | Entry | The entry field for the value of Chemical Name. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| ap\_label | Label | The label for the Amount Purchased entry field. | display() |
| ap\_entry | Entry | The entry field for the value of Amount Purchased. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| dp\_label | Label | The label for the Date Purchased entry field. | display() |
| dp\_entry | Entry | The entry field for the value of Date Purchased. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| au\_label | Label | The label for the Amount Used entry field. | display() |
| au\_entry | Entry | The entry field for the value of Amount Used. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| du\_label | Label | The label for Date Used entry field. | display() |
| du\_entry | Entry | The entry field for the value of Date Used. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| ao\_label | Label | The label for the Amount On-hand entry field. | display() |
| ao\_entry | Entry | The entry field for the value of Amount On-hand. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| button\_frame | LabelFrame | Sets up the frame for the various button options. | display() |
| add\_button | Button | The button that calls the addRecord() function when clicked. | display() |
| remove\_button | Button | The button that calls the removeRecord() function when clicked. | display() |
| update\_button | Button | The button that calls the updateRecord() function when clicked. | display() |
| display\_button | Button | The button that calls the displayRecord() function when clicked. | display() |
| clear\_button | Button | The button that calls the clear\_entries() function when clicked. | display() |

# staff\_schedule\_gui.py

## Functions

|  |  |
| --- | --- |
| Function Name | Description |
| clear\_entries() | Clears the entry fields. |
| select\_record() | Autofills the entry fields when clicking on a record displayed in the treeview. |
| generateTree() | Generates the treeview of the schedule table. |
| addRecord() | Passes on the values from the entry fields to staff\_schedule\_api.py in order to add the record to the database. |
| removeRecord() | Passes on the values from the entry fields to staff\_schedule\_api.py in order to remove matching records from the database. |
| updateRecord() | Passes on the values from the entry fields to staff\_schedule\_api.py in order to update the record of the specified ID value in the database. |

## Variables

|  |  |  |  |
| --- | --- | --- | --- |
| Name of Variable | Variable Type | Description | Functions present in |
| mydb | PooledMySQLConnection | MySQLConnection | Any | The command that allows the database to be reached. | display() |
| host | Any | The variable that defines the host server. | display() |
| user | Any | The variable that defines the user of the database. | display() |
| password | Any | The variable that defines the password used for the database. | display() |
| database | Any | The name of the database schema. | display() |
| mycursor | Any | MySQLCursor | The variable that defines the cursor that works to execute database commands. | display()  generateTree() |
| selected | str | Allows the program to select values. | select\_record() |
| values[] | Tuple | Specifies that a set of values needs to be grabbed. | select\_record() |
| myresult | Any | List[RowType] | A variable that utilizes mycursor in order to fetch the appropriate amount of records for a for loop. | generateTree() |
| x | Any | RowType | A counter value used in a for loop in conjunction with myresult to display the records. |  |
| results | list[str] | Grabs the values from the entry fields to pass into the staff\_schedule\_api.py module. | addRecord()  removeRecord()  updateRecord() |
| label | Label | Used to define the header. | display() |
| style | Style | Used to help define what different components of the GUI look like. | display() |
| pandedwindow | PandedWindow | Used to call upon the Ttk PandedWindow widget, which “displays a number of subwindows, stacked either vertically or horizontally.” In this context, it was used as a divider between the header and the rest of the interface. | display() |
| treeframe | Frame | Sets up the frame for the treeview of the schedule MySQL table. | display() |
| scrollbar | Scrollbar | Defines the scrollbar to be attached to the treeview. | display() |
| sched\_tree | Treeview | Defines the treeview of the schedule MySQL table. | display() |
| sched\_tree['columns'] | weekly\_tree: Treeview | Defines an array of headers for the treeview of the schedule table. | display() |
| sql | Literal['…'] | Passes MySQL commands to the database through Python’s connector. |  |
| input\_frame | Frame | Sets up the frame for the input fields and labels. | display() |
| id\_label | Label | The label for the ID Num entry field. | display() |
| id\_entry | Entry | The entry field for the value of ID Num. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| d\_label | Label | The label for the Date entry field. | display() |
| d\_entry | Entry | The entry field for the value of Date. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| s\_label | Label | The label for the Shift entry field. | display() |
| s\_entry | Entry | The entry field for the value of Shift. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| ma\_po\_label | Label | The label for the Manager/Pool Operator entry field. | display() |
| ma\_po\_entry | Entry | The entry field for the value of Manager/Pool Operator. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| l\_label | Label | The label for the Lifeguards entry field. | display() |
| l\_entry | Entry | The entry field for the value of Lifeguards. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| a\_label | Label | The label for Admin entry field. | display() |
| a\_entry | Entry | The entry field for the value of Admin. | display()  select\_record()  addRecord()  removeRecord()  updateRecord() |
| button\_frame | LabelFrame | Sets up the frame for the various button options. | display() |
| add\_button | Button | The button that calls the addRecord() function when clicked. | display() |
| remove\_button | Button | The button that calls the removeRecord() function when clicked. | display() |
| update\_button | Button | The button that calls the updateRecord() function when clicked. | display() |
| display\_button | Button | The button that calls the displayRecord() function when clicked. | display() |
| clear\_button | Button | The button that calls the clear\_entries() function when clicked. | display() |