Indian Institute of Information Technology, Nagpur



Attendance Record and Management

ITW-1

Computer Science & Engineering

3rd Semester

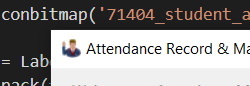
|  |  |
| --- | --- |
| Prepared By- | |
| RAJ ARYAN | BT19CSE043 |
| PRATEEK SHENDE | BT19CSE051 |
| SARTHAK GUPTA | BT19CSE054 |
| HEMANSHU CHAUDHARI | BT19CSE056 |
| KRISH RUSTAGI | BT19CSE089 |

**ACKNOWLEDGEMENT**

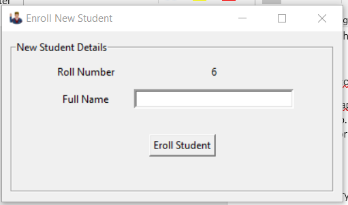
**Index**

|  |  |  |
| --- | --- | --- |
| Serial No. | Description | Page No. |
| 1 | Building the Graphical User Interface (GUI) |  |
| 2 | Creating Dataframe by Reading Excel |  |
| 3 | Display the Dataframe Record on GUI |  |
| 4 | Calculating the Percentage from Dataframe |  |
| 5 | Updating the Dataframe |  |
| 6 | Saving onto the Excel File |  |
| 7 | Modifying Records |  |
| 8 | References |  |

**BUILDING THE GRAPHICAL INTERFACE**

**Bitmapicon**

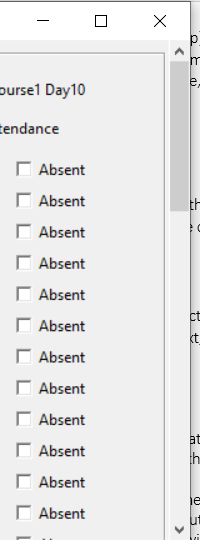
Iconbitmap(bitmap) sets the icon of the window/frame widget to bitmap. The bitmap must be an ico type, but not png or jpg type, otherwise, the image will not display as the icon.

**Geometry**

The geometry method is a fundamental one which decides the size, position and some other attributes of the screen layout we are going to create.

**Canvas**

The Canvas is a rectangular area intended for drawing pictures or other complex layouts. You can place graphics, text, widgets or frames on a Canvas.

**Scrollbar**

Canvas is not updated automatically when its content is modified, so we need to define it and update it manually using the scrollregion argument:

The Scrollbar frame works differently than other Tkniter widgets.

The values (0,0) tell the canvas on which position to draw the window. The argument anchor= ”nw” tells the canvas to place the frame's top left corner on position (0,0.

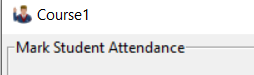
Finally, we have to configure the canvas so that when its y-position changes, the scrollbar moves:

That means that we'll be able to set the Canvas size to whatever we want, and then when we scroll, we'll be moving along the window inside it.

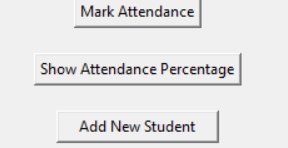
In order to do so, we must also specify what the scrollable area will be. Often, the scrollable area matches the contents of the inner window—but it can also be different if we wish.

**Frame.Bind**

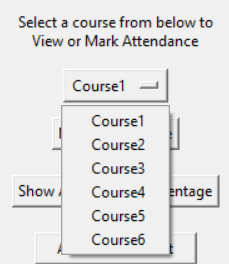
This is about binding events to functions and methods so that when the event occurs that specific function is executed.

**Frame Label**

A labelframe is a simple container widget. Its primary purpose is to act as a spacer or container for complex window layouts. This widget has the features of a frame plus the ability to display a label.

**Buttons**

The Button widget is used to add buttons in a Python application. These buttons can display text that convey the purpose of the buttons. You can attach a function or a method to a button which is called automatically when you click the button.

**Dropdown**

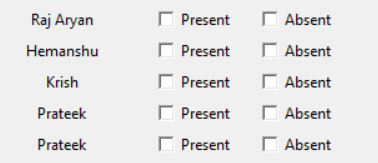
Dropdowns are toggleable, contextual overlays for displaying lists of links and more.

The value selected from the options provide in the dropdown menu can be access from the variable assigned to it by using the method, variable.get().

**Input Box**

The entry widget is used to enter text strings. This widget allows the user to enter one line of text, in a single font. The value entered in the input box can be access from the variable assigned to it by using the method, variable.get().

**Checkbox**

The Checkbutton widget is used to display a number of options to a user as toggle buttons. The user can then select one or more options by clicking the button corresponding to each option.

The value selected from the options provide in the dropdown menu can be access from the variable assigned to it by using the method, variable.get().

**Withdraw**

Removes the window from the screen, without destroying it.

**Deiconify**

Displays the window, after using either the iconify or the withdraw methods.

**READING FROM AN EXCEL RECORD**

**What is DataFrame**

DataFrame is a 2-dimensional labeled data structure with columns of potentially different types. You can think of it like a spreadsheet or SQL table, or a dict of Series objects. It is generally the most commonly used pandas object. Like Series, DataFrame accepts many different kinds of input:

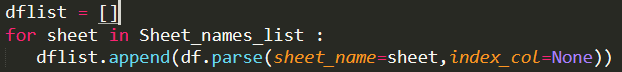
* Dict of 1D ndarrays, lists, dicts, or Series
* 2-D numpy.ndarray
* Structured or record ndarray
* A Series
* Another DataFrame

In the project we are working on dict of Series objects which is present in an excel file named ‘Output.xlsx’.

**How we Read**

To read the excel file named ‘output.xlsx’ we are using Python Pandas library with the excel file pd.ExcelFile(‘output.xlsx) where ExcelFile is equivalent to read\_excel(Excel\_File). pd.Excelfile returns DataFrame or dict of DataFrames.

**Used a list of DataFrames**

We are using a list of dataframes so as to read all the different sheets per course present in the excel sheet ‘output.xlsx’. Since we have 6 courses, so a total of 6 sheets would be available for the dataframes. To read that we have created a blank list initially and later we are appending the DataFrames for each of the 6 sheets.7u

**dataframe.parse**

Dataframe .parse is used to parse the specified sheets in the dataframe. Till now all the sheet list names are present in the Sheet\_names\_list. From here we are fetching the data present in the excel sheet.

**index\_col = None**

Column(s) to use as the row labels of the DataFrame, either given as string name or column index. If a sequence of int / str is given, a MultiIndex is used. index\_col=None/False can be used to force pandas to not use the first column as the index, e.g. when you have a malformed file with delimiters at the end of each line. Since we want the first column of each excel sheet to be empty we are using such labels.

Displaying the data from dataframe to UI

-dynamically selecting with list item to show

-looking for empty columns to take input for that day

-checkbox variable

-dropdown variable

- variable.get()

-different rows

-checkboxes along with name of student

-

Calculating %

-selecting the list item based on the selection of course

-reading number of presents in a single row

-reading total classes attended

-calculating

- displaying the output with name of student

-

Updating dataframe list

-taking input from the checkbox

-present variable

-absent variable

-markpresent()

-markabsent()

Saving the record

-save()

-excelwriter

-why all list object being written to the file

Modifying the record

* Add student
* Taking input from the input box
* Updating dataframe
* Save
* Delete student
* Taking input roll number
* Taking name input
* Searching the record
* Deleting the row
* Save()

References