**A Multi-Measure Unit Converter with GUI**

**App Designed and Developed By:** Adebayo Onajoko

**Programming Language Used:** Python (Jupyter Notebook Environment)

**Python Libraries Used for building the App:** Pandas and Tkinter

**Brief Description of the Application:**

This application is a multi-measure Unit Converter that was designed to convert units for different measures. The measures included are time, mass, length, volume, speed, area, fuel consumption and digital storage. The conversion is made possible by utilizing Pandas library to read from the excel sheet (Conversion Chart) that contains the 7 different measures, the various units and the corresponding conversion rates between the units. The App was developed using Python language in Jupyter Notebook and the Tkinter library was used for the App GUI.

**Fig 1.0 Original Look of the App**

**The Steps to using the App:**

In order to understand how the App works, let’s briefly use a mass conversion of 100.25Kg to Pounds(lb.) as a case study. The steps to carry out the conversion of the case above are as follows:

1. Choose the measure of interest from the measure dropdown box from the available 7 and in the above case study, mass is selected from the measure dropdown box.
2. Enter the scalar quantity you wish to convert – for the case under consideration, a numerical value of 100.25 is entered into the "Qty to Convert" box
3. Select Kilogram from the "Convert From " dropdown box and select Pound from the "Convert To" dropdown box
4. Click on the "**CONVERT**” button.

The look of the App before and after clicking on the "**CONVERT**” button are shown in the link below:

**Fig 1.1 The App before clicking** "**CONVERT**"

**Fig 1.2 The App after clicking** "**CONVERT**"

In order to carry out a different unit conversion- choose a different measure, enter the numerical quantity you intend to convert, select the unit you want to convert from and the unit you want to convert to and then click the "**CONVERT**" button or you just click on the "**CLEAR**" button which clears all the present inputs and the App is then ready for a new set of inputs.

**Exception handling feature of the App:**

The App is also equipped with exception handling features which enables the App to catch exceptions such as wrong combination of inputs into the App or inputting a non-numerical value into the "Qty to Covert" box which is meant to accept only integers or floating numbers. The above exception handling cases are captured in the screenshots which can be accessed through the links below:

**Fig 1.3 A non-numerical value inputted**

**Fig 1.4 An invalid value inputted**

**Final word on the use of the App:**

It should be noted that the App can only be replicated in its full capacity if the code inside the Jupyter Notebook file, the excel worksheet (conversion chart) and all the necessary image files are all in the same directory.