CREATE A DIGITAL CLOCK BY USING TKINTER

CODE EXPLANATION:

- THE CODE STARTS BY IMPORTING THE NECESSARY MODULES.
- THE FIRST MODULE IS THE TKINTER LIBRARY, WHICH PROVIDES BASIC FUNCTIONALITY FOR CREATING GRAPHICAL USER INTERFACES (GUIS).
- NEXT, THE STRFTIME FUNCTION IS IMPORTED TO RETRIEVE SYSTEM TIME.
- NEXT, A WINDOW IS CREATED AND GIVEN A TITLE OF “CLOCK.”
- A FUNCTION CALLED TIME() IS THEN CREATED TO DISPLAY THE CURRENT TIME ON THE LABEL WIDGET.
- THIS FUNCTION USES THE STRFTIME() FUNCTION TO FORMAT THE TIME STRING ACCORDING TO SYSTEM CONVENTIONS.
- THE LAST PART OF THIS CODE SETS UP STYLING FOR THE LABEL WIDGET SO THAT IT WILL LOOK NICER.
- FINALLY, AN INSTANCE OF LABEL IS CREATED AND PLACED AT THE CENTER OF THE WINDOW.

- THE TIME() FUNCTION IS EXECUTED, AND YOUR OUTPUT SHOULD LOOK LIKE THIS: CLOCK: TUE DEC 12 08:00:00 2016
- THE CODE CREATES A WINDOW AND ASSIGNS IT THE TITLE “CLOCK”.
- THE TIME() FUNCTION IS THEN CALLED TO DISPLAY THE CURRENT TIME ON THE LABEL WIDGET.
- THE LBL.CONFIG() FUNCTION IS USED TO SET THE TEXT OF THE LABEL WIDGET.
- THE AFTER() FUNCTION IS USED TO DELAY DISPLAYING THE TIME FOR 1000 MILLISECONDS.
- FINALLY, THE STYLE OF THE LABEL WIDGET IS MODIFIED WITH LBL.PACK().

Code:

```
# importing whole module
from tkinter import *
from tkinter.ttk import *

# importing strftime function to
# retrieve system's time
from time import strftime

# creating tkinter window
root = Tk()
root.title('Clock')

# This function is used to
# display time on the label

def time():
        string = strftime('%H:%M:%S %p')
        lbl.config(text=string)
        lbl.after(1000, time)
```

```
# STYLING THE LABEL WIDGET SO THAT CLOCK
# WILL LOOK MORE ATTRACTIVE
LBL = LABEL(ROOT, FONT=('CALIBRI', 40, 'BOLD'),
                        BACKGROUND='PURPLE',
                        FOREGROUND='WHITE')

# PLACING CLOCK AT THE CENTRE
# OF THE TKINTER WINDOW
LBL.PACK(ANCHOR='CENTER')
TIME()

MAINLOOP()
```

OUTPUT: