|  |
| --- |
| **DIGITAL CLOCK**  **21CSS101J – PROGRAMMING FOR PROBLEM SOLVING**  **Mini Project Report**  *Submitted by*  **Amitabh Garg [Reg. No.: RA2211003011402]**  **B.Tech. CSE - Core**  **SRMIST-01.jpg**  **SCHOOL OF COMPUTING**  **COLLEGE OF ENGINEERING AND TECHNOLOGY**  **SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**  **(Under Section 3 of UGC Act, 1956)**  S.R.M. NAGAR, KATTANKULATHUR – 603 203  KANCHEEPURAM DISTRICT  **December 2022** |

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Title** | **Page No.** |
| 1 | Problem Statement |  |
| 2 | Methodology / Procedure |  |
| 3 | Coding (C or Python) |  |
| 4 | Results |  |
| 5 | Conclusion |  |

**Problem Statement:**

Create a digital clock using Python.

**Methodology / Procedure:**

1.The code starts by importing the necessary modules.

2.The first module is the tkinter library, which provides basic functionality for creating graphical user interfaces (GUIs).

3.Next, the strftime function is imported to retrieve system time.

4.Next, a window is created and given a title of “Clock.”

5.A function called time() is then created to display the current time on the label widget.

6.This function uses the strftime() function to format the time string according to system conventions.

7.The last part of this code sets up styling for the label widget so that it will look nicer.

8.Finally, an instance of Label is created and placed at the center of the window.

9.The time() function is executed, and your output should look like this: Clock: Tue Dec 12 08:00:00 2016

10.The code creates a window and assigns it the title “Clock”.

11.The time() function is then called to display the current time on the label widget.

12.The lbl.config() function is used to set the text of the label widget.

13.The after() function is used to delay displaying the time for 1000 milliseconds.

14.Finally, the style of the label widget is modified with lbl.pack().

**Coding (Python):**

# 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=('Times New Roman', 40, 'bold'),

background='green',

foreground='black')

# Placing clock at the centre

# of the tkinter window

lbl.pack(anchor='center')

time()

mainloop()

**Graphical user interface, text, application

Description automatically generatedResults:**

Graphical user interface, website

Description automatically generated

**Conclusion:**

Hence the program is executed & verified successfully.