# Alarm Clock with Python

An alarm clock is a clock with a function that can be activated to ring at a time set in advance, used to wake someone up. In this article, I’ll walk you through how to write a Python program to create an alarm clock with Python.

## How to Create an Alarm Clock with Python?

Our task here is to write a python script that creates an alarm clock. For this task, I will be using the DateTime module in Python to create an alarm clock and the sound library in Python to play the alarm sound.

The DateTime module comes preinstalled in the Python programming language so you can easily import it in your program. The playsound library can be easily installed by using a pip command; pip install playsound. I hope you will be able to install it in your systems, now let’s see how to write a program to create an alarm with Python.

## Alarm Clock with Python

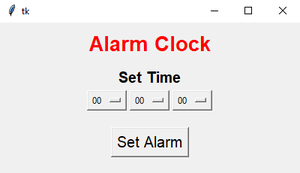
Before writing the program we should know that you also need an alarm tone which will ring at the time of the alarm. So we need download an alarm tune from here. Now as we are ready with the libraries and the alarm song, let’s see how to write a program to create an alarm clock with Python:

CODE

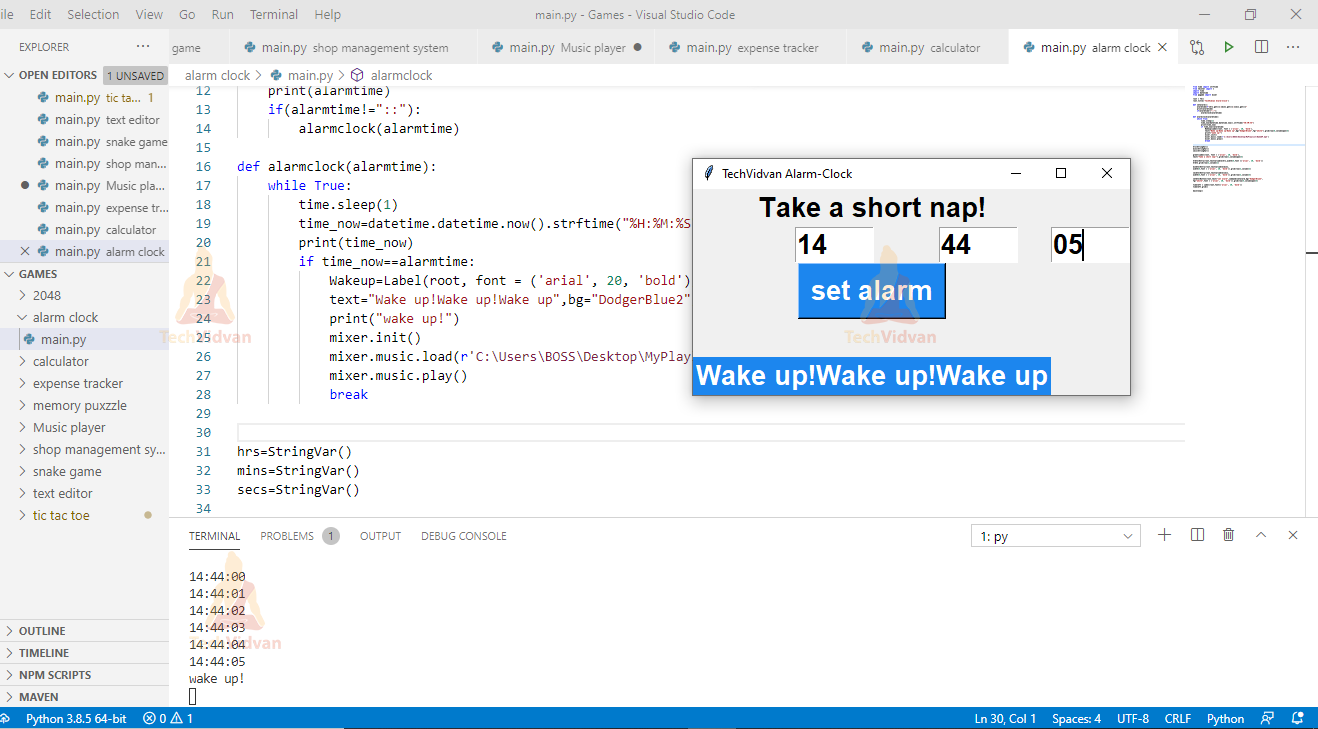
|  |
| --- |
| from datetime import datetime |
|  | from playsound import playsound |
|  | alarm\_time = input("Enter the time of alarm to be set:HH:MM:SS\n") |
|  | alarm\_hour=alarm\_time[0:2] |
|  | alarm\_minute=alarm\_time[3:5] |
|  | alarm\_seconds=alarm\_time[6:8] |
|  | alarm\_period = alarm\_time[9:11].upper() |
|  | print("Setting up alarm..") |
|  | while True: |
|  | now = datetime.now() |
|  | current\_hour = now.strftime("%I") |
|  | current\_minute = now.strftime("%M") |
|  | current\_seconds = now.strftime("%S") |
|  | current\_period = now.strftime("%p") |
|  | if(alarm\_period==current\_period): |
|  | if(alarm\_hour==current\_hour): |
|  | if(alarm\_minute==current\_minute): |
|  | if(alarm\_seconds==current\_seconds): |
|  | print("Wake Up!") |
|  | playsound('audio.mp3') |
|  | Break |

The user input should be in a format of hours: minutes: and then seconds. You will start listening to the song as you will reach the time that has been set. To test your code set the time 2 or 3 minutes later from the time you are giving the user input.

Screen Shorts



Coding in python



### Summary

This idea can be implemented in software applications also, so you now have an idea of what can be a good Python project other than just designing the User interface of an application.