**DICE SIMULATOR USING PYTHON**

Introduction:

As we all know, the concept of dice rolling is based on the game of chance. We cannot predict the outcome of a trial and this is the reason behind its many applications in all walks of life. It also finds many applications in virtual world. Many online games require a dice as an essential part of the game, this invites many developers to create dice rolling simulators which help users play the game without a physical dice.

In this project I will attempt to create one such simulator using Python.

Let’s Begin:

This model will basically have two important components:

* Graphical User Interface(GUI)
* The main code

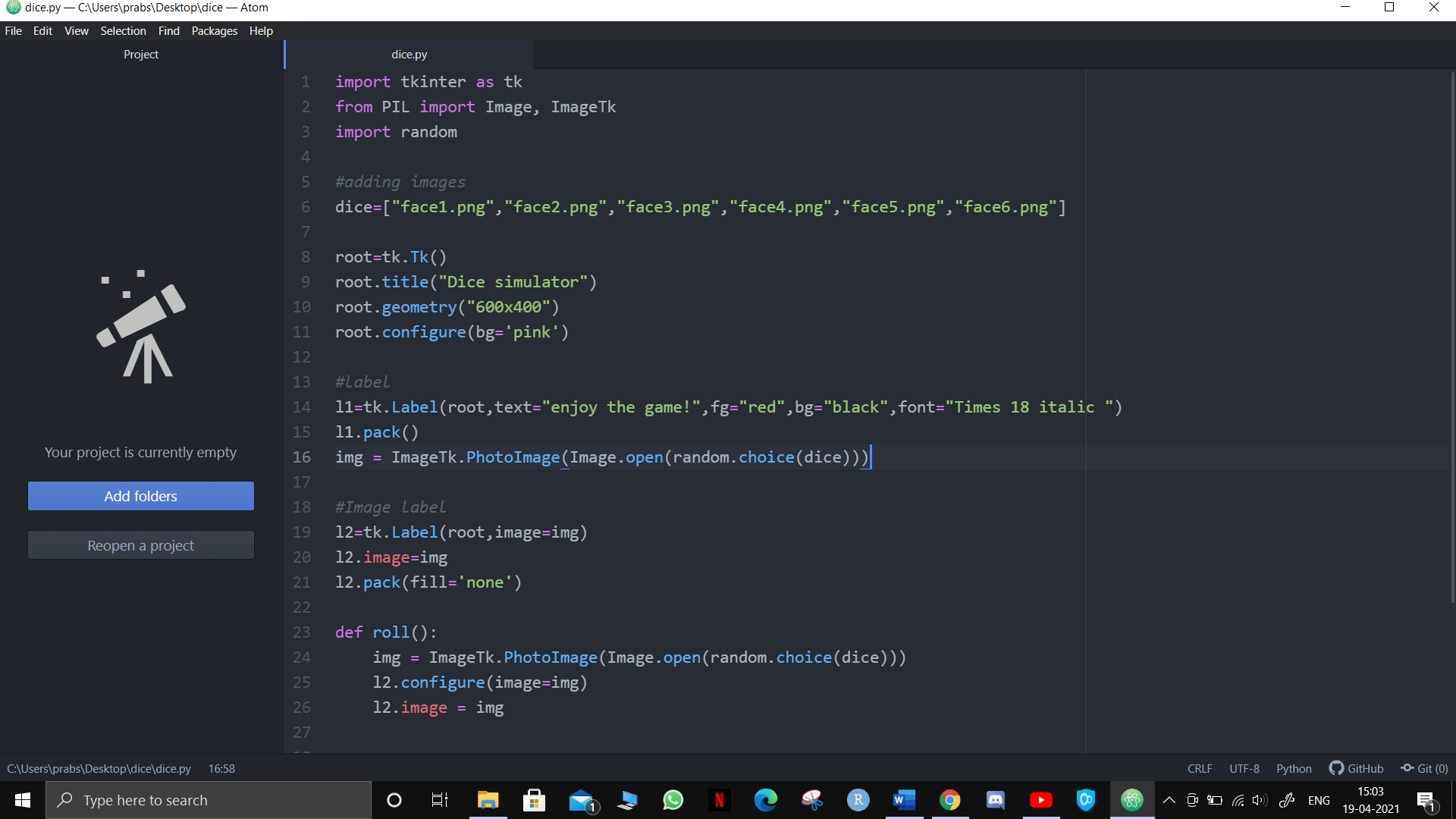
The GUI part is necessary to demonstrate our simulation, and the source code will run in the back end of the GUI.

Python has many GUI frame works and one such handy framework is the “Tkinter” package which is inbuilt in python. It will provide a basic interface to our simulation and give an app like look.

Steps:

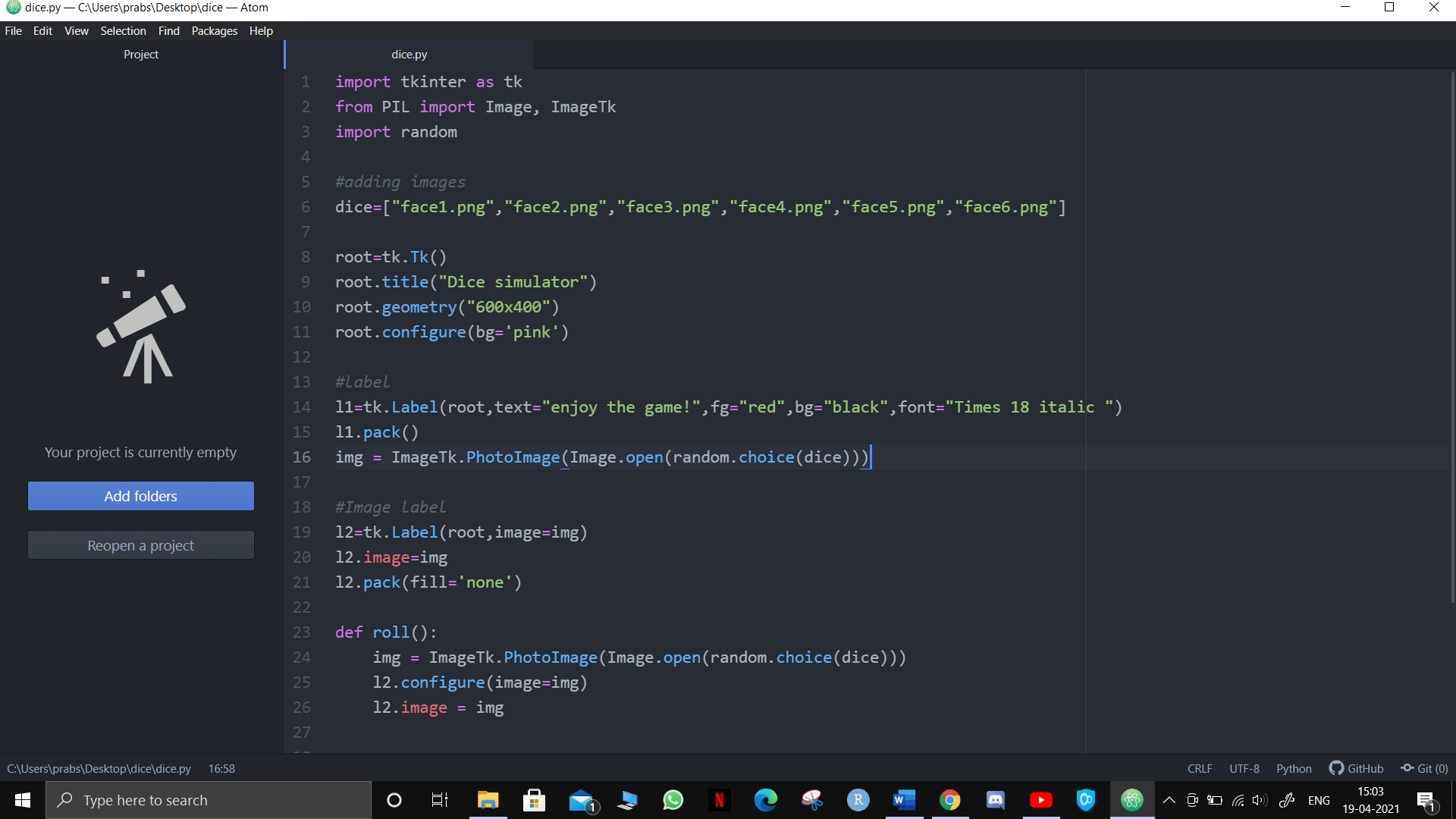
Here are the basic steps of our project:

1. Import necessary libraries.



* Tkinter library is necessary for for the GUI
* ImageTk modules are necessary for modifying the image attributes of PIL images.
* And random module is necessary for displaying random images from a list of images

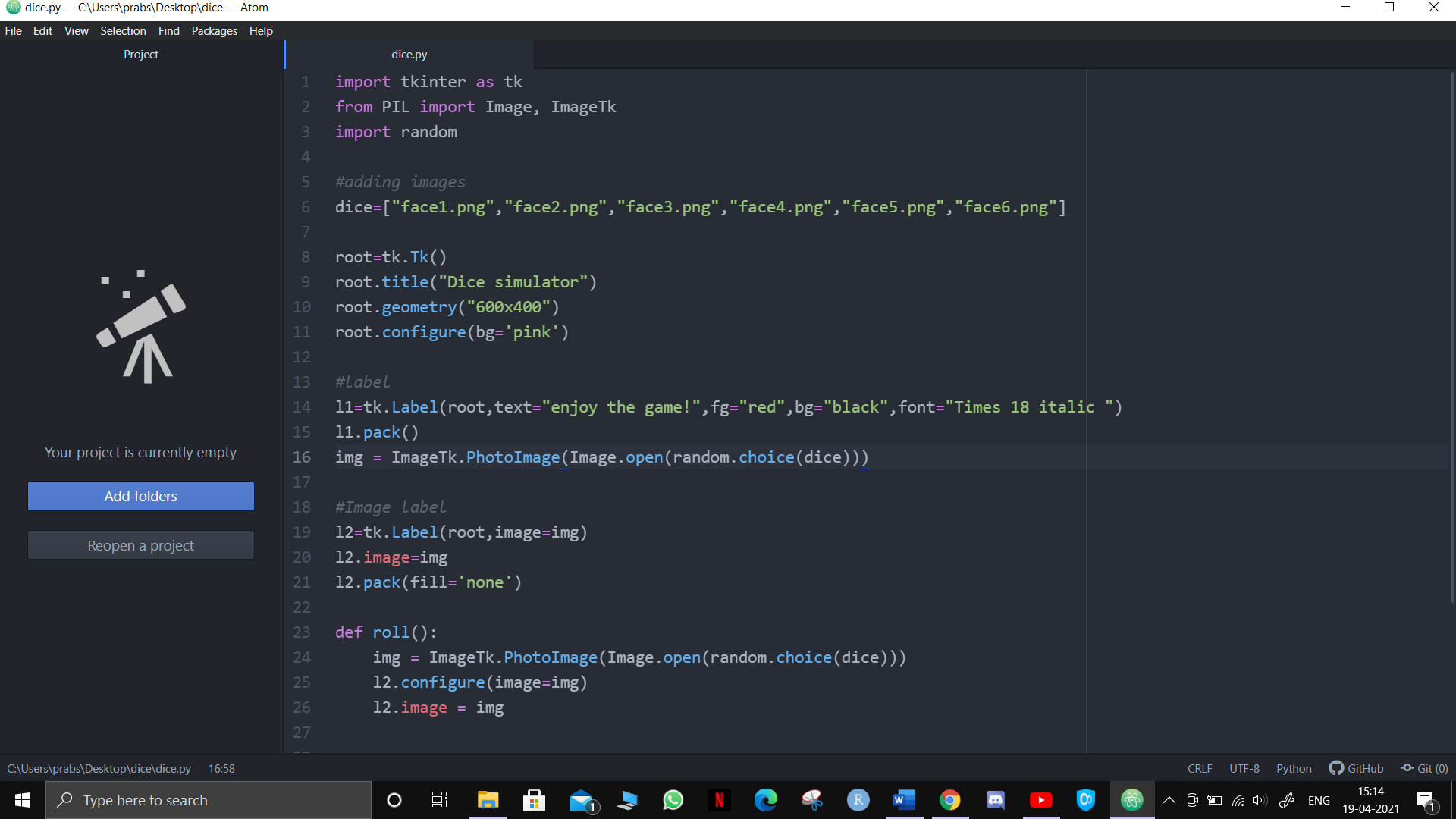
1. Creating a list of images for each face of the dice.



Here we stored a list of six images which will be later used for simulation.

Note: the images should be stored in the same directory as your python code for them to be used directly.

1. Creating the framework of the tkinter window.

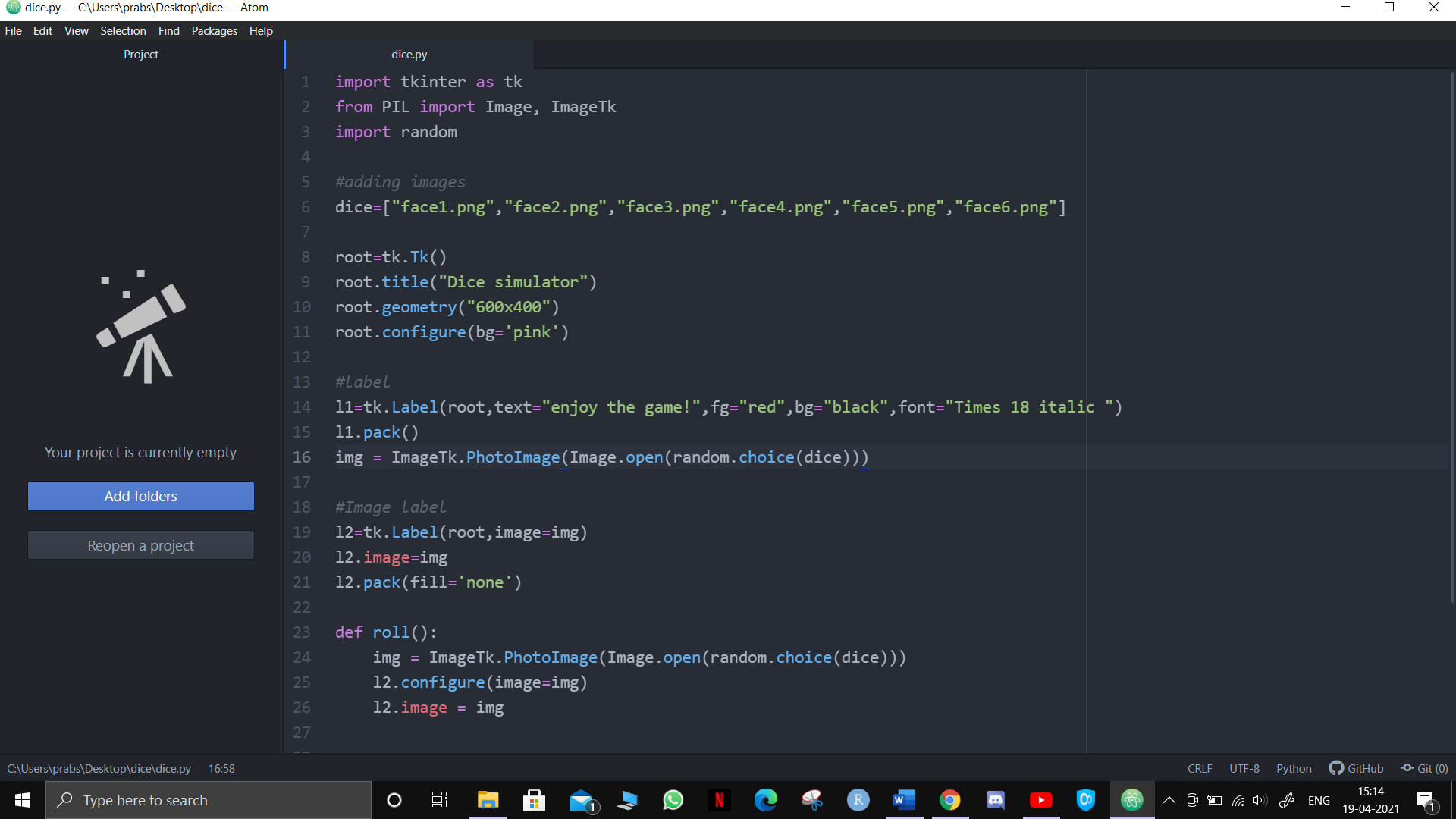


In these set of commands;

* we declared a variable and created the root window of our GUI.
* We assigned the window the title of our project
* We then gave it adequate dimensions and finally,
* Added some background color.

1. Adding label window to our simulator.

Now that we have created a root window lets create a label window:



* Here our label variable is l1.
* Using the Label function of tk we customized the text of our label window.

Output:

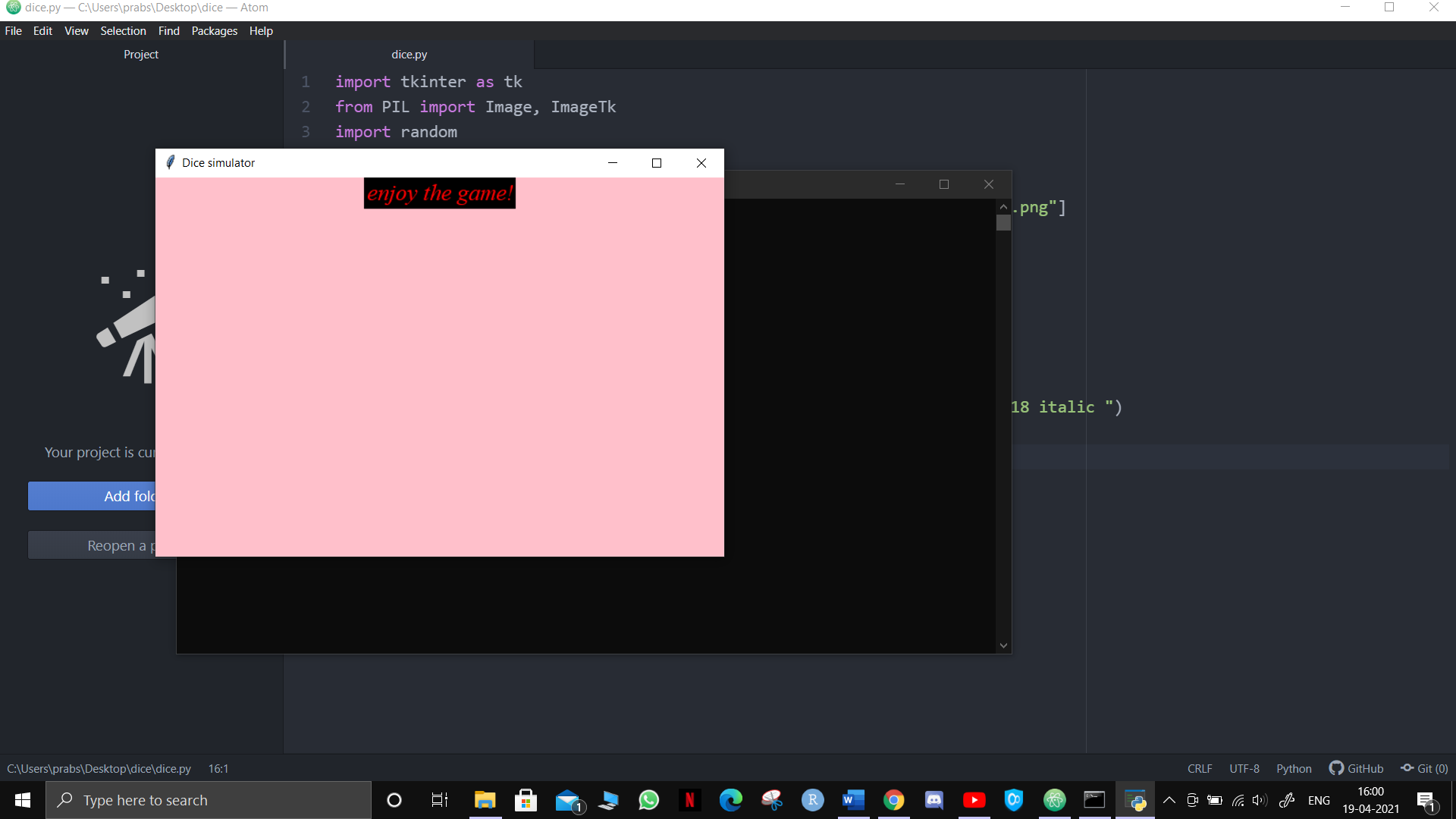
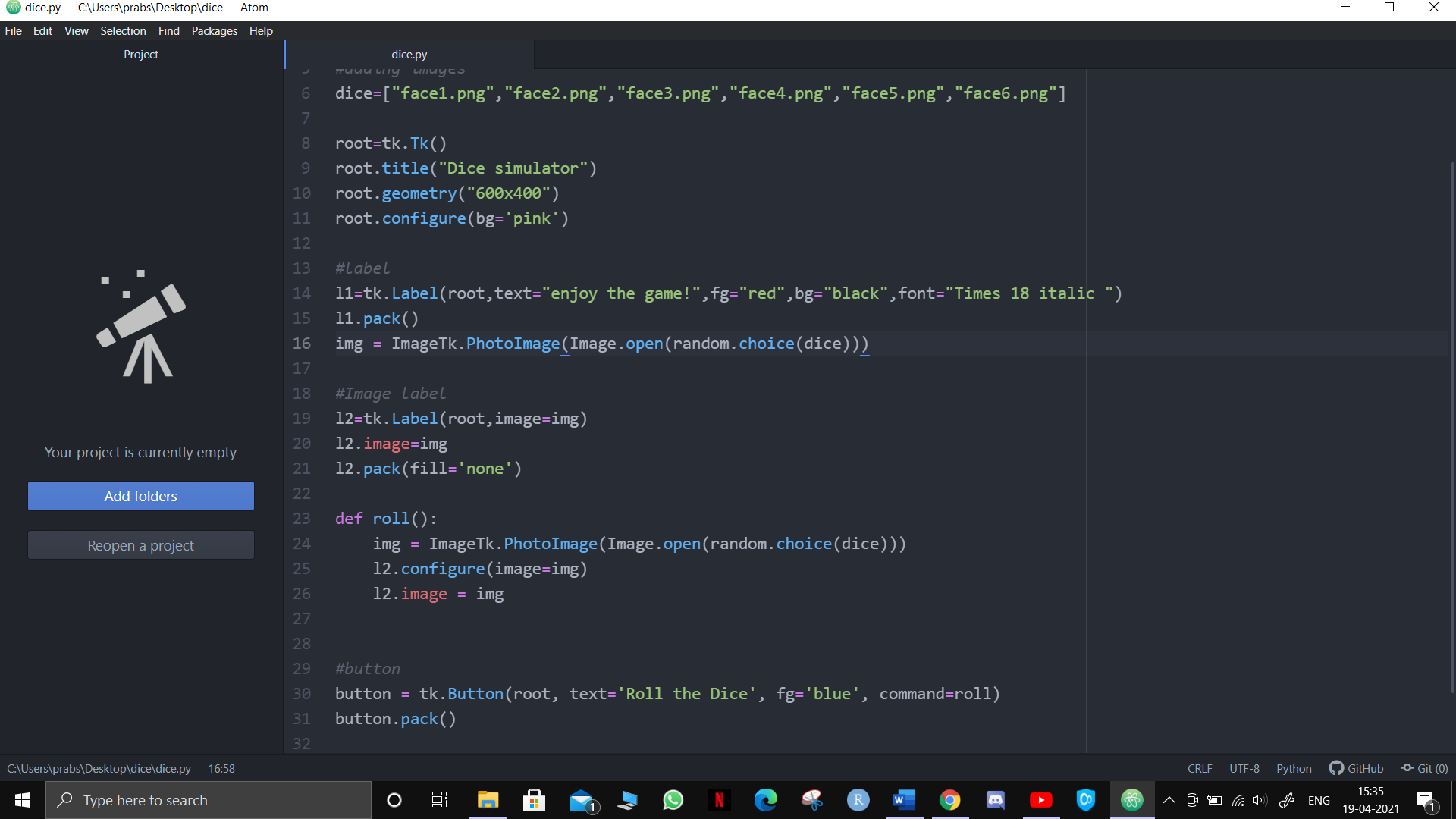
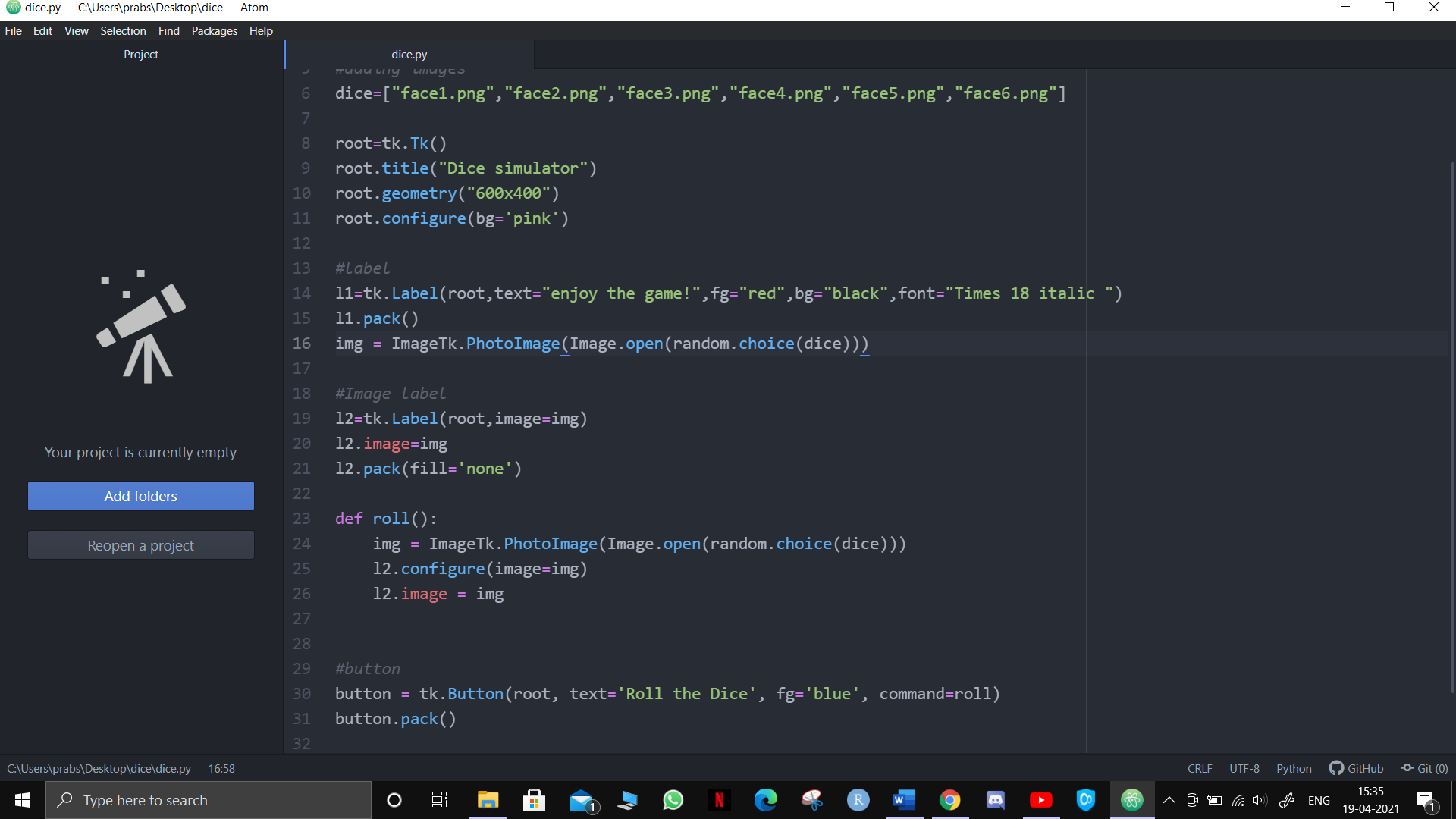


Image label:



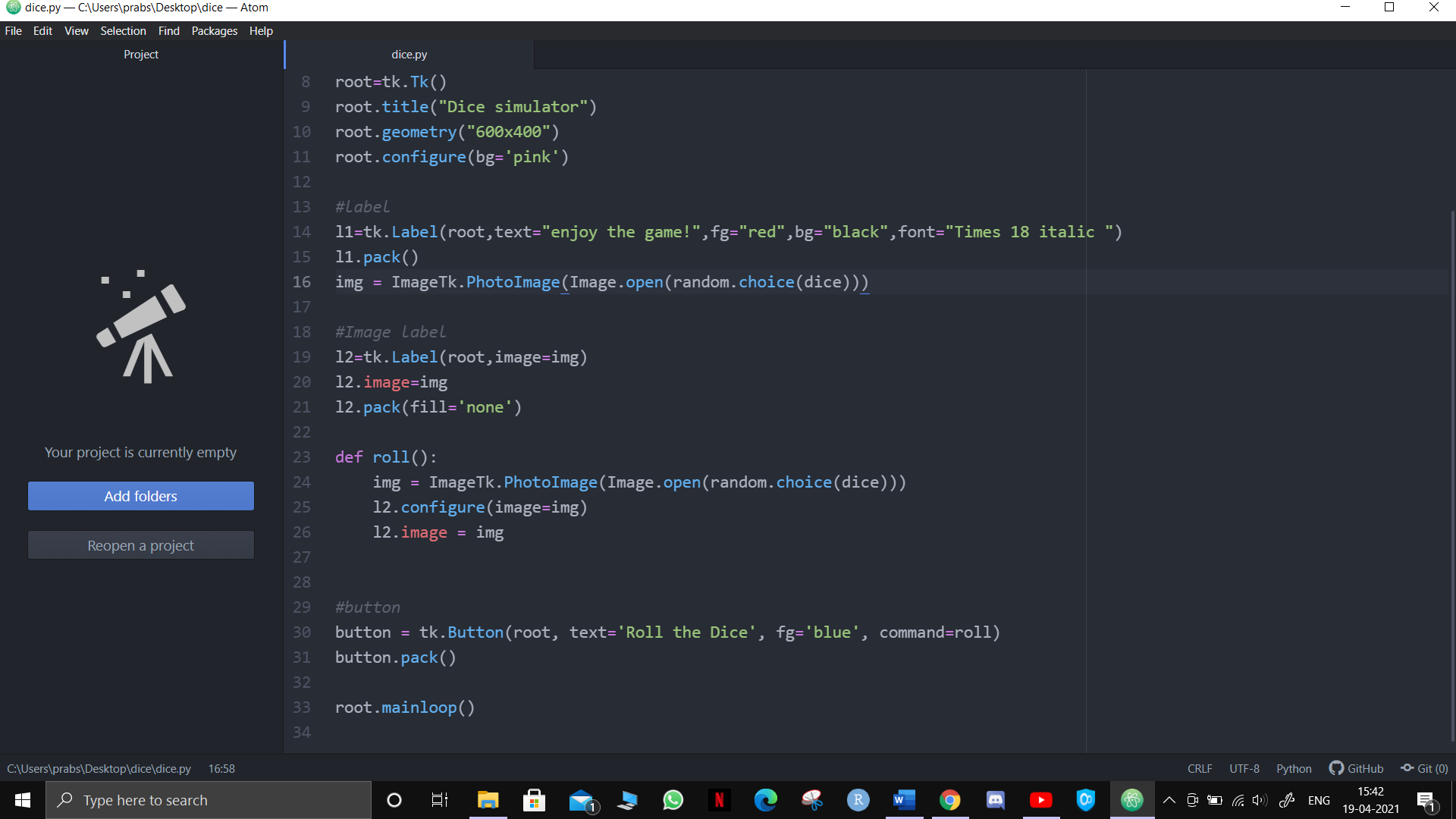
* Now the next step is to display an image for our label, for this we used the random function to display any random image from our ‘dice’ list, so that when we open our simulator, we will see a random dice face being displayed.  
  "PhotoImage()" function returns the image object and it provides with a set of attributes for image modifications.
* Then we will append this image in our l2 label.

1. Creating our roll function



* Roll is our main function in which we will perform the rolling operation.
* The command for random image will be same as before and we will again append the randomly generated image to label2(l2).

1. Ready to roll



* Finally we added a button that would call the roll function and perform the rolling operation.
* The dice is ready to be rolled

**Final simulator:**

