***Errors and Visualization:***

*So, there are instances when we deal with errors in Programming and bas times we don’t really understand what the Error is trying to mean and just sit Clueless.*

*Now, to deal with that we are going to see a better way of doing the same with some different tools. Which we will cover in this documentation today.*

*Error visualization:*

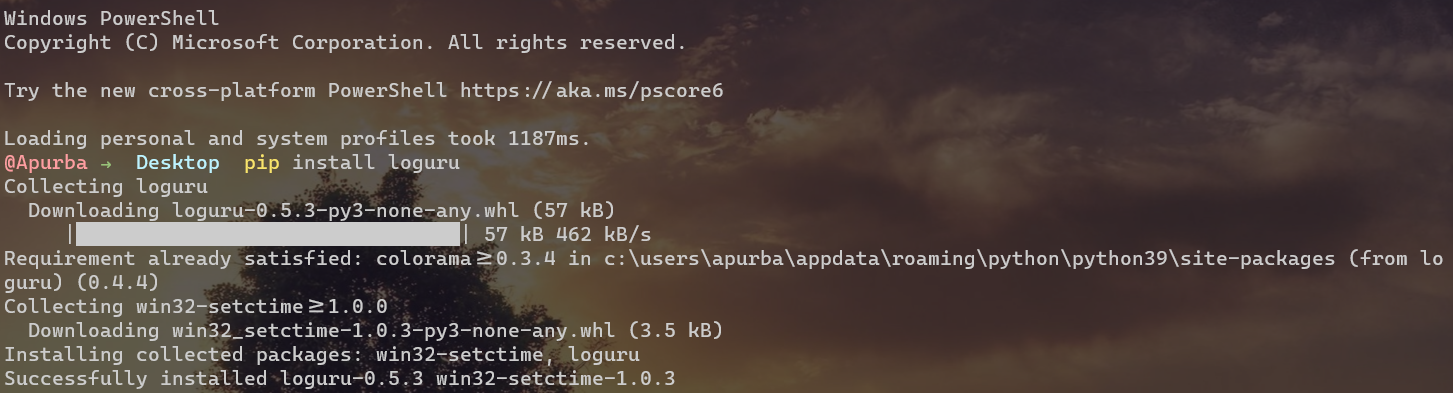
*Loguru*

*So, to help us with our Error Visualization in Python we have a Python Module called* ***Loguru*** *. It helps us finding the Errors where they are and explain it to us in a much better and Understandable way and highlights the portions and also shows the values we have entered for each instance or a variable.*

*Let us see a Example of what we have just read.*

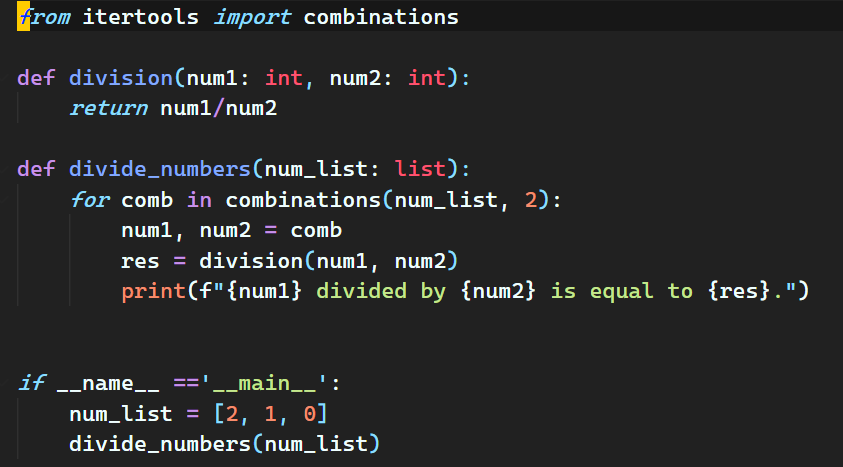
*To, Begin with first we will install the* ***Loguru*** *module we are talkig about.*

*Installation Instructions 🡪 pip install loguru*

**

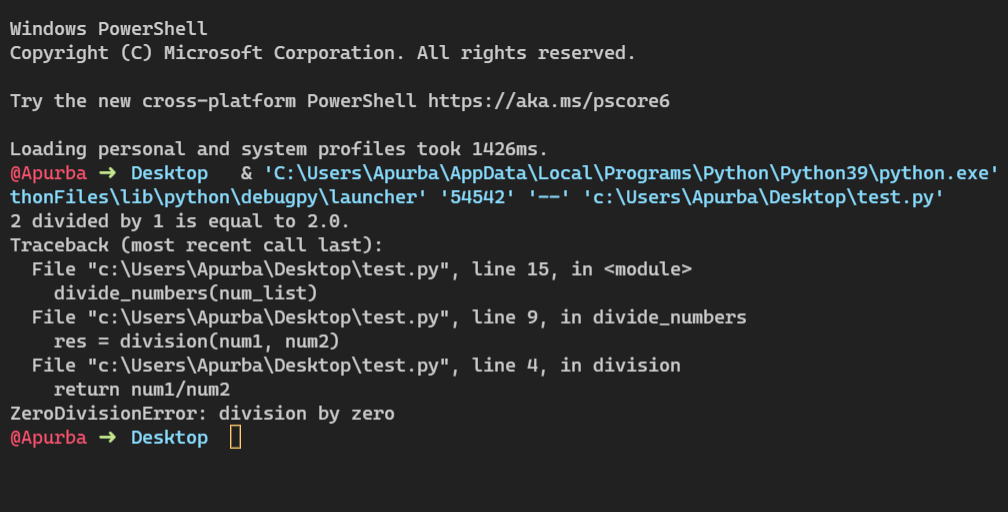
***Next up, we will see how will it helps us and how do we implement the following module in our code. 🡪***

*Now, what you see below is a Sample code of a division containing Functions, here we pass a list to a function and it divided each and every number with its succeeding number, and every time for doing that it takes help of another Function by Calling it.*

******

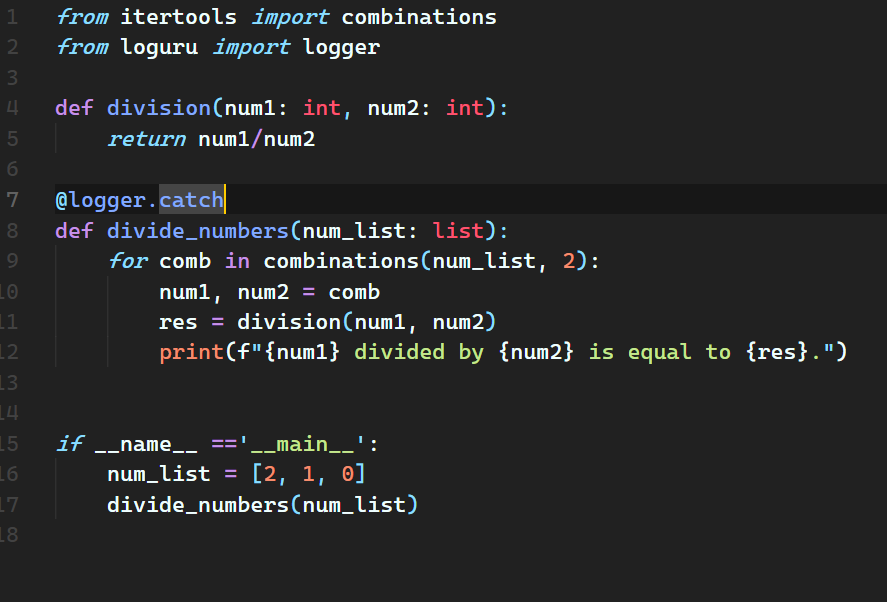
*Now, if we try running this Code, if you guys can notice it will throw a Error as we have passed ‘0’ as an in the list as a Argument in the Function which will eventually gives us* ***ZeroDivisionError****, Now we know the error but let’s suppose that this would be a long codebase and you can’t really understand the bug in your code in that case let’s first see the Error which we all will get.*

***Error:***

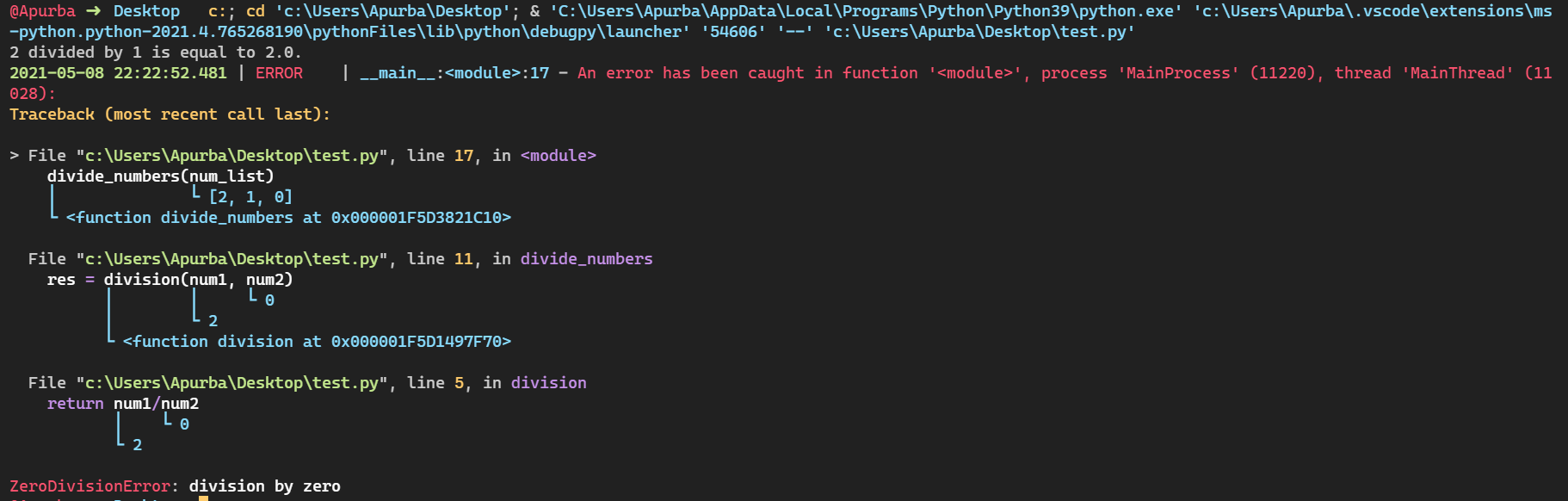
**

*Now, this is something we all have expected, but what if we had a Visual Representation, let’s see how can we achieve that with* ***Loguru***

*To do that we made a simple change into our code, we just import the* ***Loguru*** *Module into our script and then Exactly tracks the part which gives the error, in this case we are getting error from the* ***Function*** *itself.*

**

*And by doing this, the Error we get here is,*

**

***How about this Visual Representation Errors and Exactly knowing for which values we are getting the Errors?***

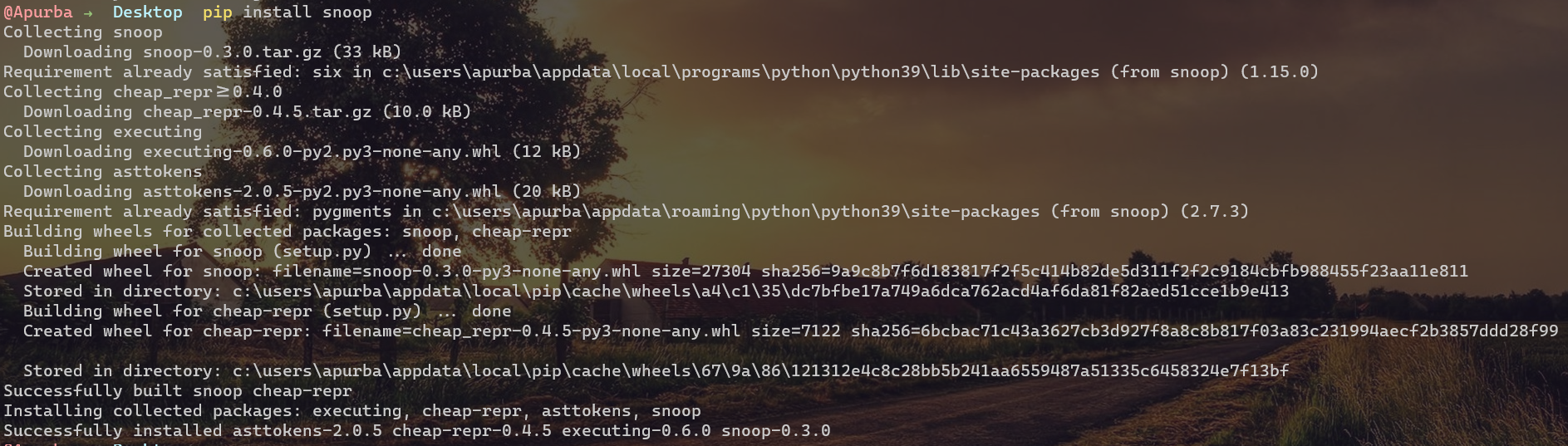
*Code Visualzation:*

*Snoop*

*What if there is no error in the code, but we want to figure out what is going on in the code? That is when* ***snoop*** *comes in handy.*

***Snoop****is a Python package that prints the lines of code being executed along with the values of each variable by adding only one decorator.*

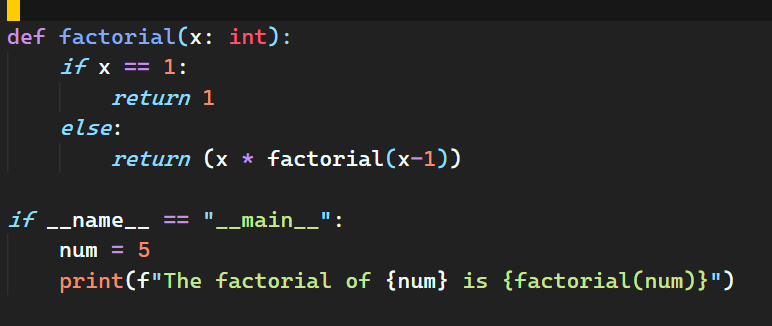
*Installation Instructions 🡪 pip install snoop*

**

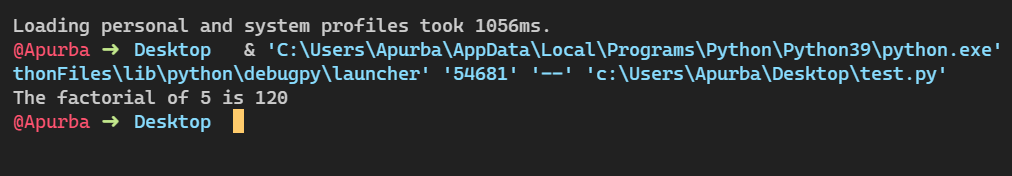
***Next up, we will see how will it helps us and how do we implement the following module in our code. 🡪***

*Now, here we have a Simple Code defining what a Factorial is and helping us to get the Factorial of a Number using the help of a Function.*

***Note: We have used Recursive Function in here in this Code.***

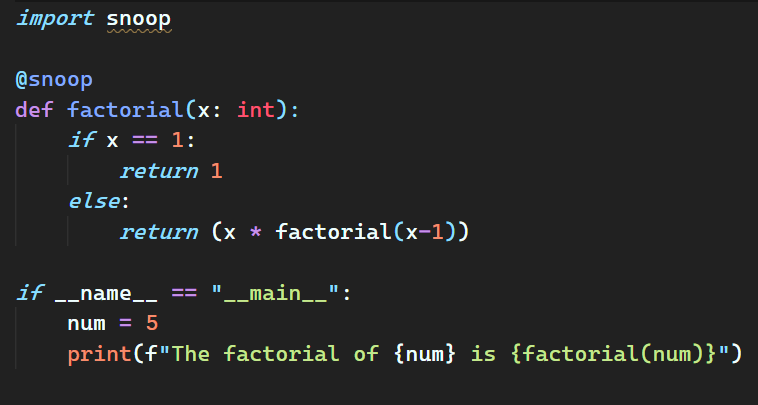
**

*Now, the Output is pretty predictable, it will print the factorial of the Number just like this 🡪*

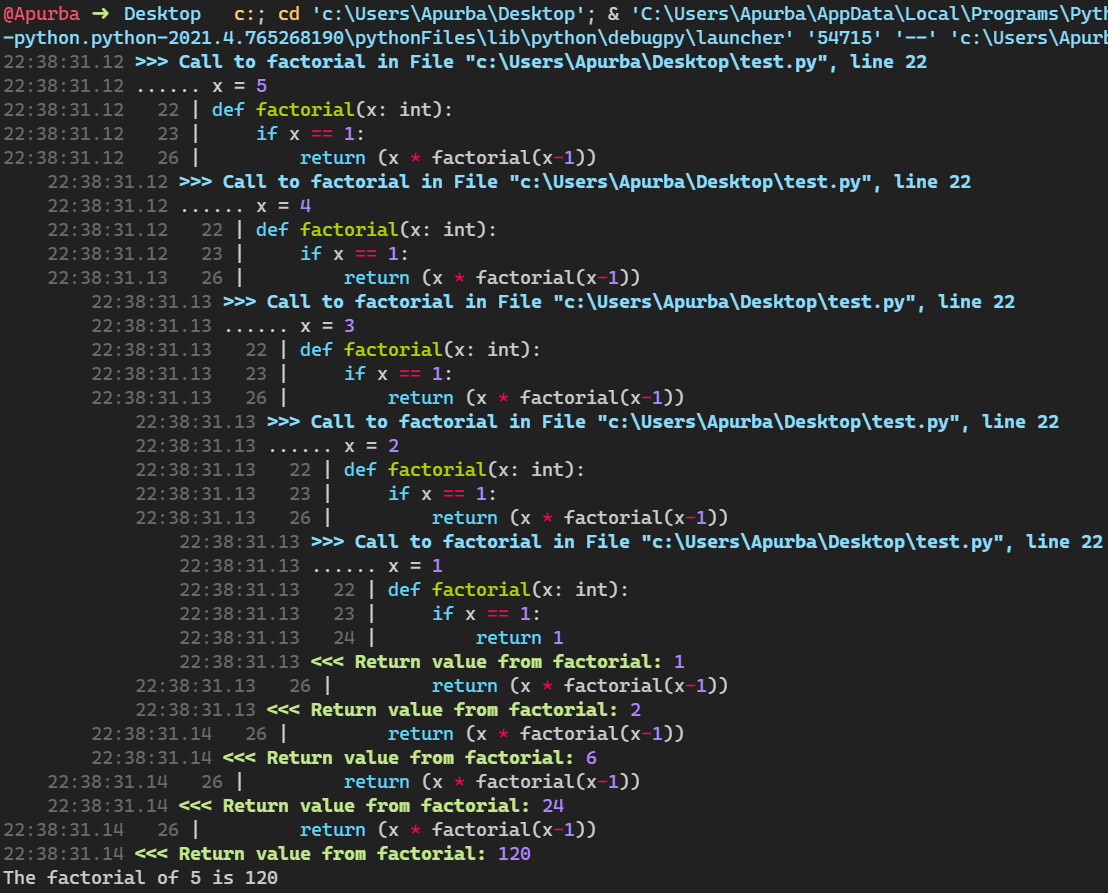
**

*But, what if I were to tell you, there is more you can do with it, but not just fetch the Output and understand how the Code Works in the Background and how does the interpreter runs it in Background. Lets’ see how can we achieve that using* ***Snoop. Also, we will use Snoop here as a Decorator as we do in Python.***

***After we implement the Snoop Code the Code we have written will look somewhat like this…***

******

*After this as we see how the Output looks now, much better right?*

**

*Isn’t it great? Look, how better it is to understand what’s going inside…*

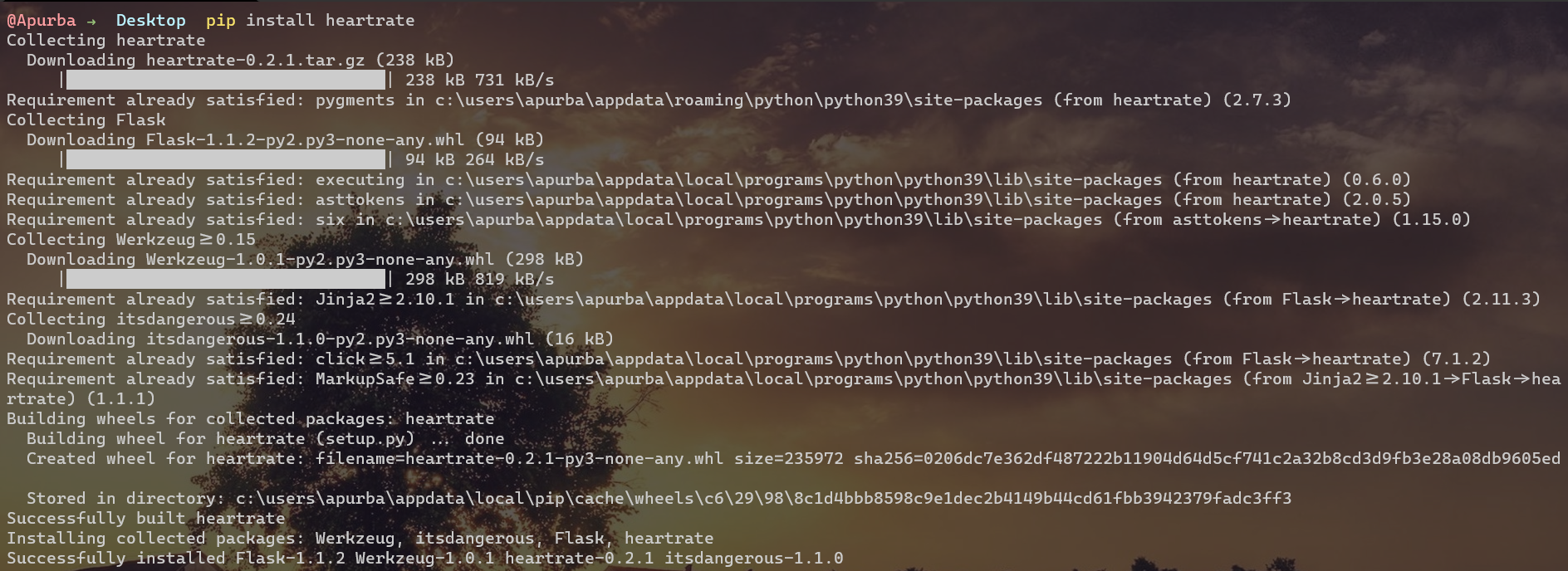
***Note: The Process using Snoop does take some time but it is better at times when you wanna understand what’s going inside.***

*Heartrate*

*Think suppose we have written a piece of code and we wanna see how that is actually working inside as we really don’t get what it’s happening as the Code is misbehaving i.e it is not giving the Output we are expecting it to give.*

*In that case, If you want to visualize which lines are executed and how many times they are executed, try* ***heartrate****.* ***Heartrate*** *is also created by the creator of* ***snoop****.*

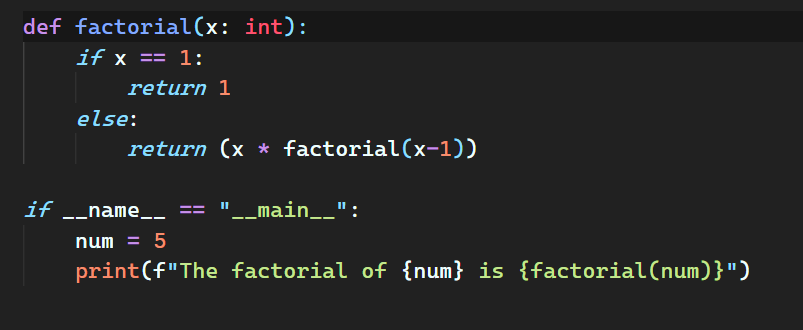
*Installation Instructions 🡪 pip install heartrate*

**

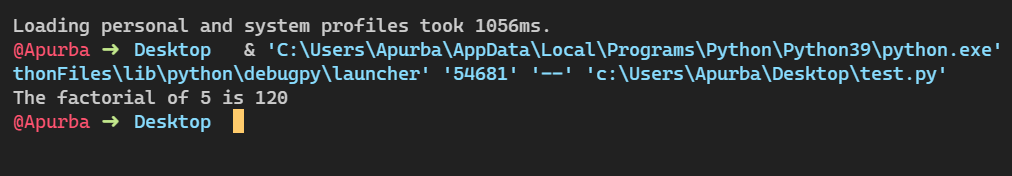
***Next up, we will see how will it helps us and how do we implement the following module in our code. 🡪***

*Now, here also we have the same Sample Code defining what a Factorial is and helping us to get the Factorial of a Number using the help of a Function.*

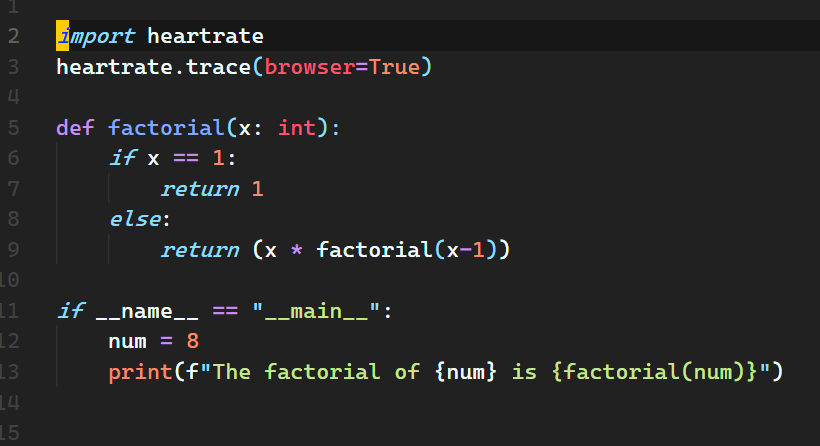
***Note: We have used Recursive Function in here in this Code.***

**

*The Output here is pretty Predictable and as we can see here just like this as we have got previously.*

**

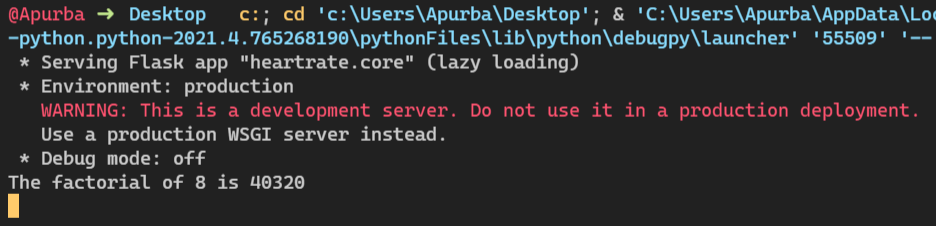
*But after implementing* ***Heartrate*** *we get to see as the code as,*

**

*And the Output is what we have right here,*

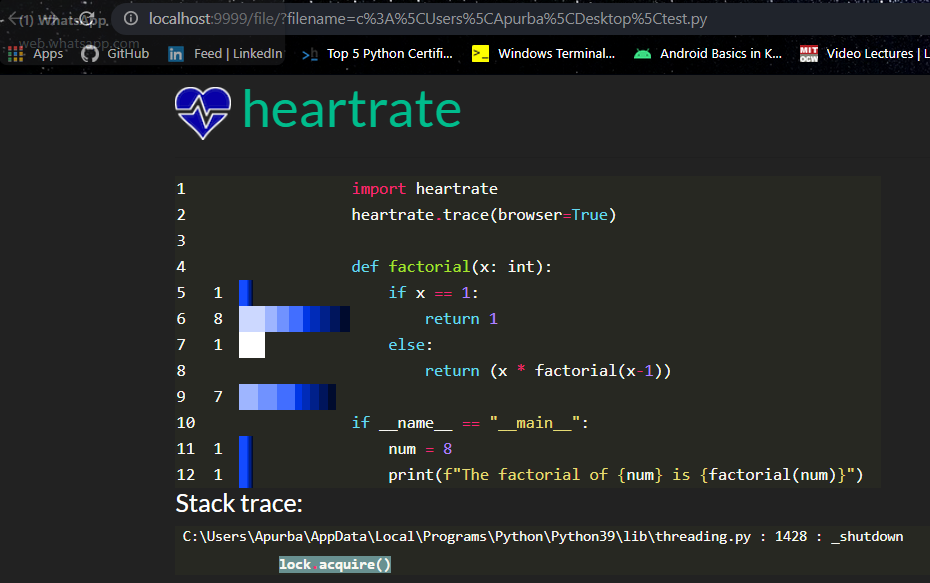
*Note: It would open a tab in your browser in Localhost in your machine in port* ***localhost:9999***

*So, you will see something like this in Your Terminal or run window…*

**

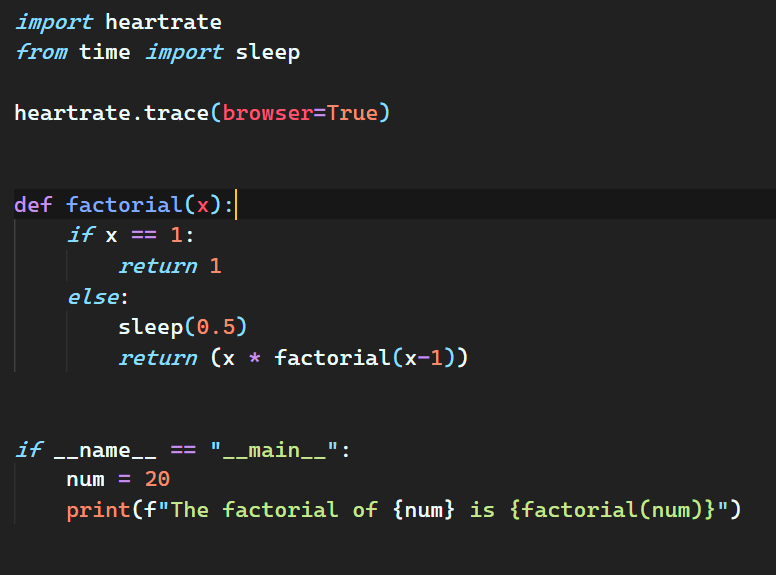
***Note: Since it is a Server running at address localhost:9999, it won’t stop until you stop the program flow.***

*And you will notice such a window in your browser just like in below with an image of your code and number of times each line is executed…*

**

***Note: The numbers here shows the number of times each line is executed and the wheels here shows the visual representation of the same… The lighter the color the more recent it is in execution, the longer the bars means the more number of hits.***

*Now let’s see what it is like to visualize the execution of a Python program in real-time using heartrate. Let’s add sleep(0.5) so that the program runs a little bit slower and increase num to 20 .*

**

*So the terminal remains the same, but the browser will show something different, as soon as it pops up look over there and you will see that there is an animation where the wheels are rotation as the the programs flow,*

*Do try it out…*

*That’s it guys for the tutorial and Documentation…*

***© iamapurba2003, Stark-Corp***