Debugging Python Scripts

Introduction

Imagine one of your colleagues has written a Python script that's failing to run correctly. They're asking for your help to debug it. In this lab, you'll look into why the script is crashing and apply the problem-solving steps that we've already learned to get information, find the root cause, and remediate the problem.

Reproduce the error

The script sent by your colleague is in the **scripts** directory. Let's navigate to **scripts** directory using the following command:

```
cd ~/scripts
```

Now, use the following command to list all the files in this directory:

```
ls
```

You should now be able to see the file named **greetings.py** which was sent by your colleague.

To view the contents of the file, use the following command:

```
cat greetings.py
```

Let's update the file's permissions.

```
sudo chmod 777 greetings.py
```

Now let's reproduce the error by running the file using the following command:

```
./greetings.py
```

Enter your name at the prompt.

The output should throw an error as shown below:

```
gcpstaging99646_student@linux-instance:~/scripts$ ./greetings.py
Hello!, What's your name?Alex
Traceback (most recent call last):
  File "./greetings.py", line 10, in <module>
    greeting()
  File "./greetings.py", line 8, in greeting
    print("hello " + name + ", your random number is " + number)
TypeError: Can't convert 'int' object to str implicitly
gcpstaging99646_student@linux-instance:~/scripts$
```

Great job! You have successfully reproduced the error.

Find the root cause of the issue

Now that we have successfully reproduced the error, let's find its root cause.

The error message indicates that something in the code is trying to concatenate a string and an integer.

```
gcpstaging99646_student@linux-instance:~/scripts$ ./greetings.py
Hello!, What's your name?Alex
Traceback (most recent call last):
  File "./greetings.py", line 10, in <module>
    greeting()
  File "./greetings.py", line 8, in greeting
    print("hello " + name + ", your random number is " + number)
TypeError: Can't convert 'int' object to str implicitly
gcpstaging99646_student@linux-instance:~/scripts$
```

When we look at the code, we can see that there are two different data types used, string and int. The variable **name** takes string values and the variable **number** stores integer (int) values.

So, the print statement within the script concatenates both string and integer values, which is causing the error.

```
print("hello " + name + ", your random number is " + number)
```

So, we can conclude that the root cause of the issue is within the print statement, which is trying to concatenate two different data types (i.e., string and int).

Debug the issue

In the previous section, we found the root cause of the issue, now let's debug the issue.

The print statement within the script is trying to concatenate two different data types. In Python, you can't add two different data types directly. So in this case, we can't add a string data type with an int data type. To add them, we have to turn the number into a string using *str()* function.

Once the integer value is converted to a string data type using *str()* function, the concatenate operation will work because the data types will be similar.

str() function takes in an integer as a parameter and converts it into string data type. So in our case, we will pass the variable **number** to the str() function i.e., **str(number)**.

Open the greetings.py file using nano editor.

```
nano greetings.py
```

Replace the print statement within the script with the following statement:

```
print("hello " + name + ", your random number is " + str(number))
```

Save the file by pressing Ctrl-o, followed by the Enter key and Ctrl-x.

Now run the file again.

```
./greetings.py
```

Enter your name for the prompt. You should now see the correct output.

```
gcpstaging99646_student@linux-instance:~/scripts$ ./greetings.py
Hello!, What's your name?Alex
hello Alex, your random number is 77
```