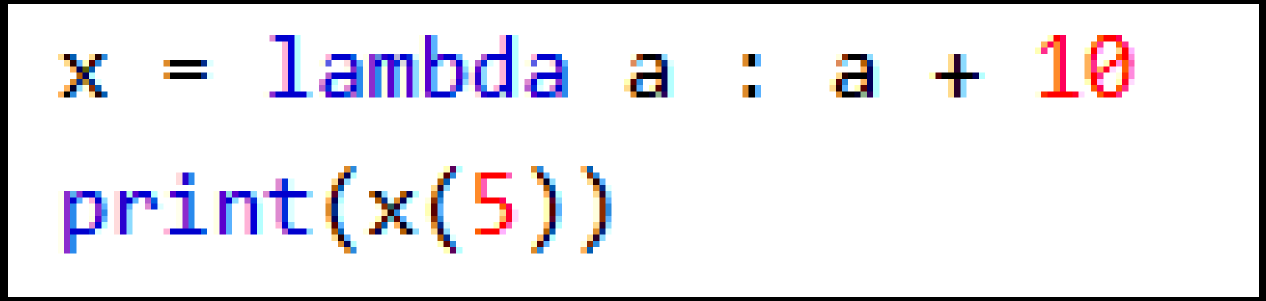
Python Lambda

A lambda function is a small anonymous function.

A lambda function can take any number of arguments, but can only have one expression.

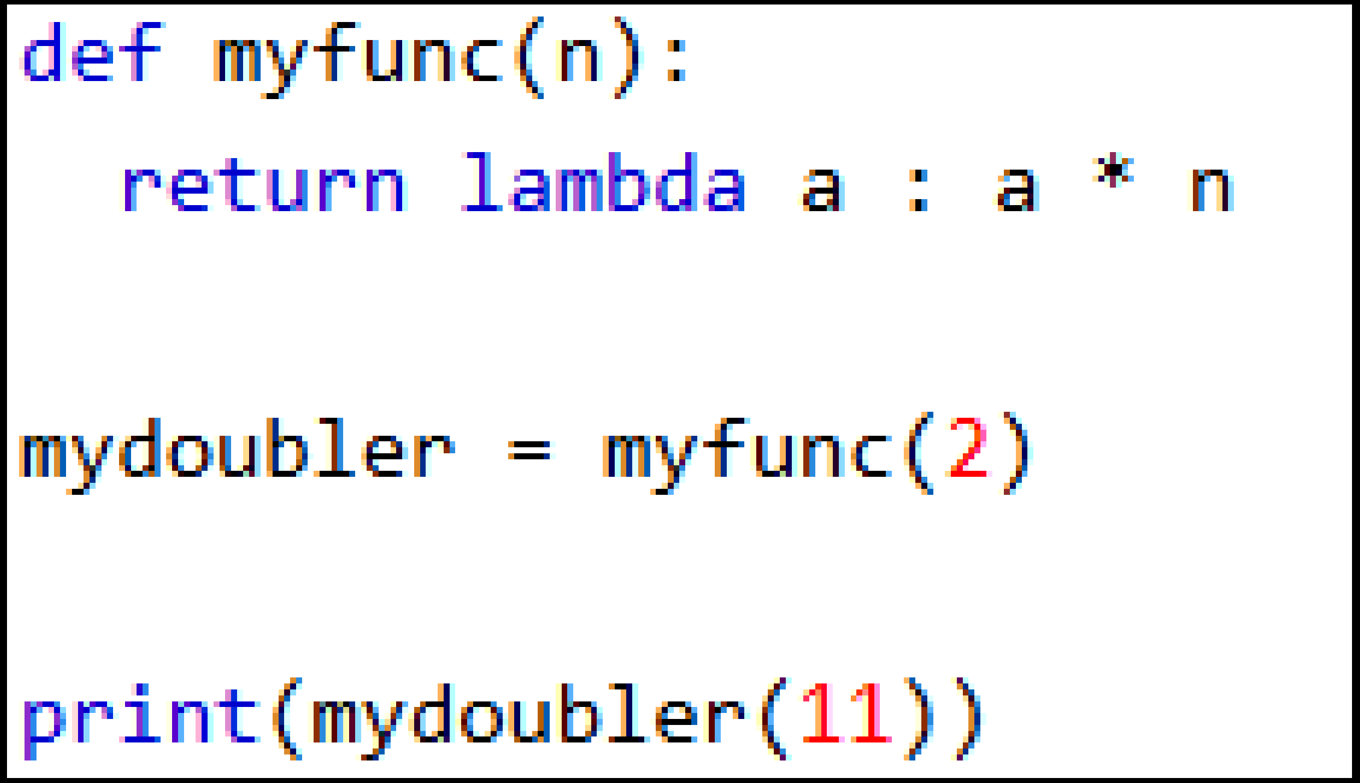
lambda arguments : expression



```python
x = lambda a : a + 10
print(x(5))
```

Python Lambda

The power of lambda is better shown when you use them as an anonymous function inside another function



```
def myfunc(n):
  return lambda a : a * n

mydoubler = myfunc(2)

print(mydoubler(11))
```

Let‘s Practice

Use a lambda function to calculate the product of two numbers.

Let‘s Practice

Use a lambda function to find the maximum of two numbers.

Let‘s Practice

Write a lambda function to check if a number is positive, negative, or zero.

Let‘s Practice

Given a list of (name, age) pairs, sort it using a lambda function by age.

File I/O in Python

Python can be used to perform operations on a file. (read & write data)

Types of all files

1.Text Files : .txt, .docx, .log etc.

2. Bina Files : .mp4, .mov, .png, .jpeg etc. na College

Open, read & close File

We have to open a file before reading or writing.

```
f = open( “file_name”,“mode”)
```

sample.txt
demo.docx

r : read mode
w : write mode

```
data = f.read( )

f.close( )
```

File I/O in Python

Character	Meaning
“r”	Open for reading (default)
“w”	Open for writing, truncating the file first
“x”	Create a new file and open it fir writing
“a”	Open for writing, appending to the end of the file if it exists
“b”	Binary mode
“t”	Text mode (default)
“+”	Open a disk file for updating (reading & writing)

Reading a file

```
data = f.read( ) #reads entire file

data = f.readline( ) #reads one line at a time
```

Writing to a file

```
f = open( “demo.txt”, “w”)

f.write( “this is a new line“ ) #overwrites the entire file
```

```
f = open( “demo.txt”, “a”)

f.write( “Adding some new line“ ) #adds to the file
```

Black Origin

Documents

[Stackoverflow]

with Syntax

```
with open( “demo.txt”,“a”) as f:
          data = f.read()
```

Deleting a File

using the os module
Module (like a code library) is a file written by another programmer that generally has a functions we can use.

```
import os

os.remove( filename )
```

Let‘s Practice

Write a Python program to create a new file named "notes.txt" and add the following data to it:

Hello everyone
We are learning about file handling
using JavaScript.
I enjoy coding in JavaScript.

Let‘s Practice

Write a function that replaces all occurrences of "JavaScript" with "Python" in the "notes.txt" file.

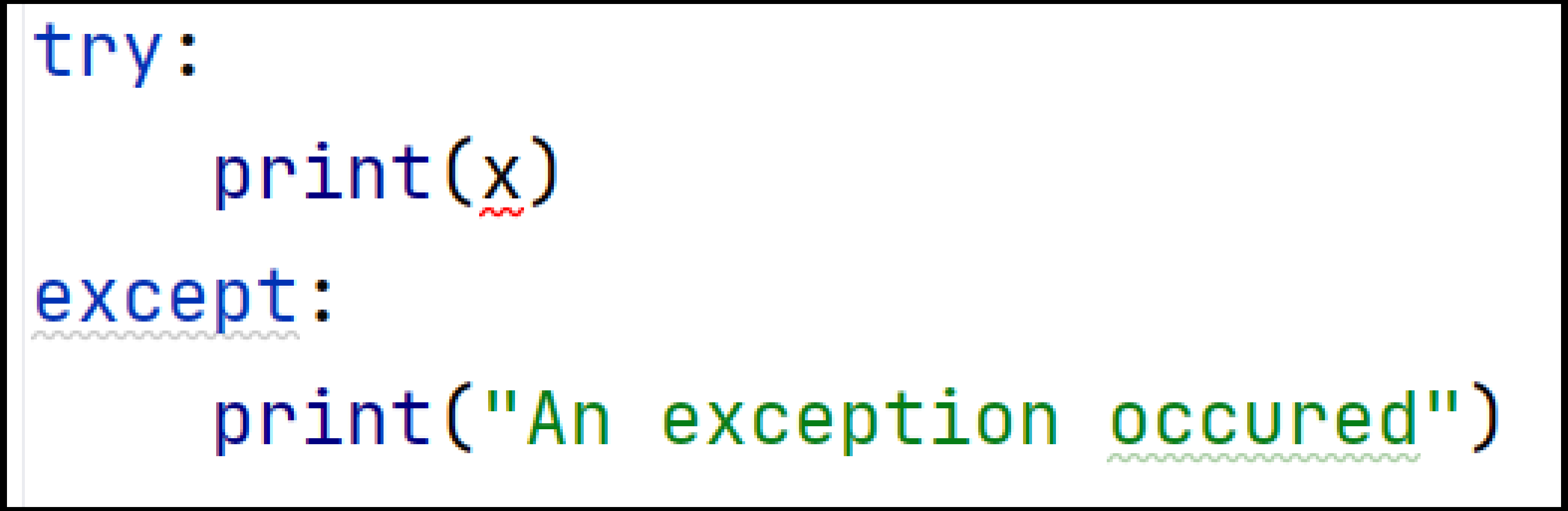
Let‘s Practice

Write a function to check if the word "learning" exists in "notes.txt". Print "Found" if it exists, otherwise print "Not Found".

Exception Handling

When an error occurs, or exception as we call it, Python will normally stop and generate an error message.

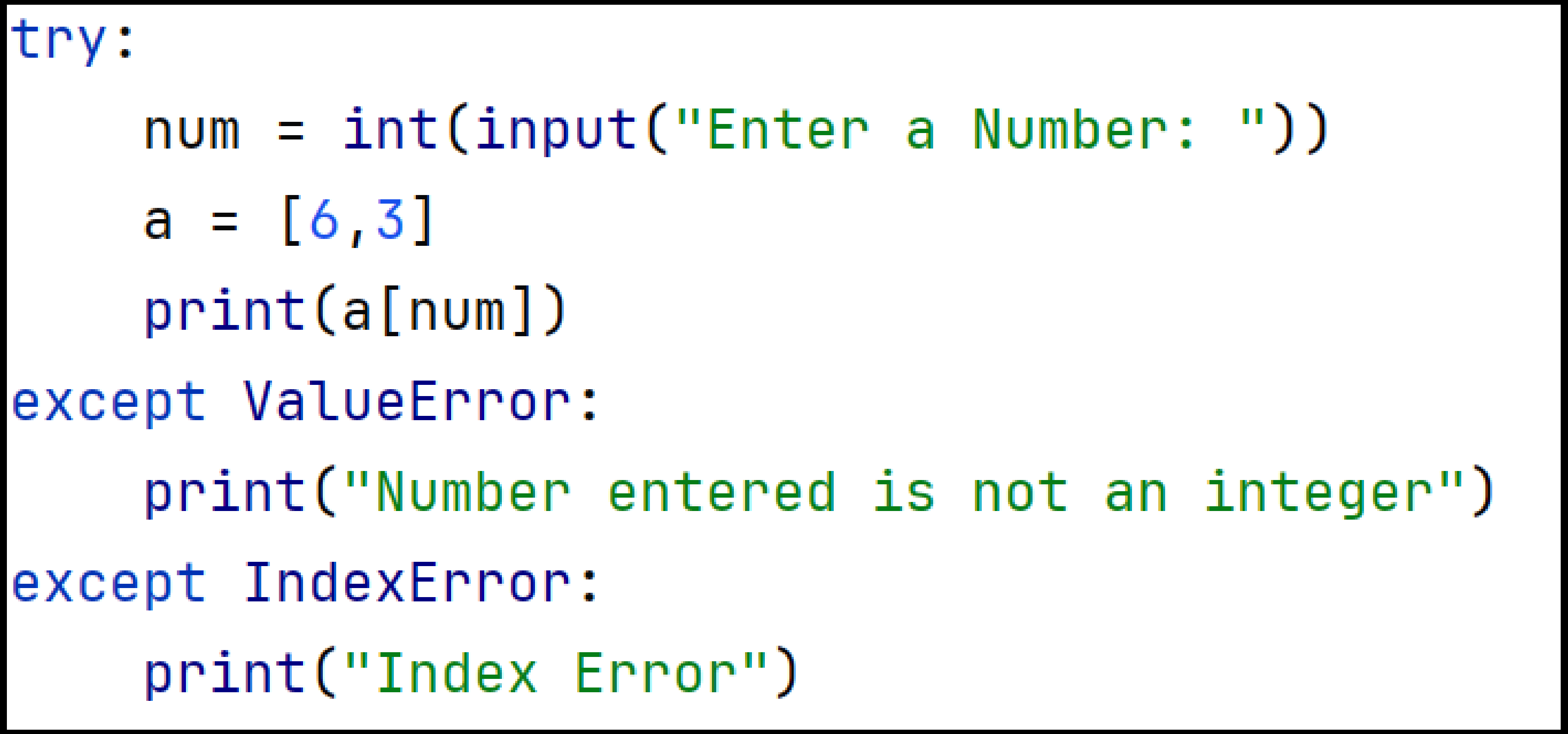
These exceptions can be handled using the try statement:



```
try:
    print(x)
except:
    print("An exception occured")
```

Many Exceptions

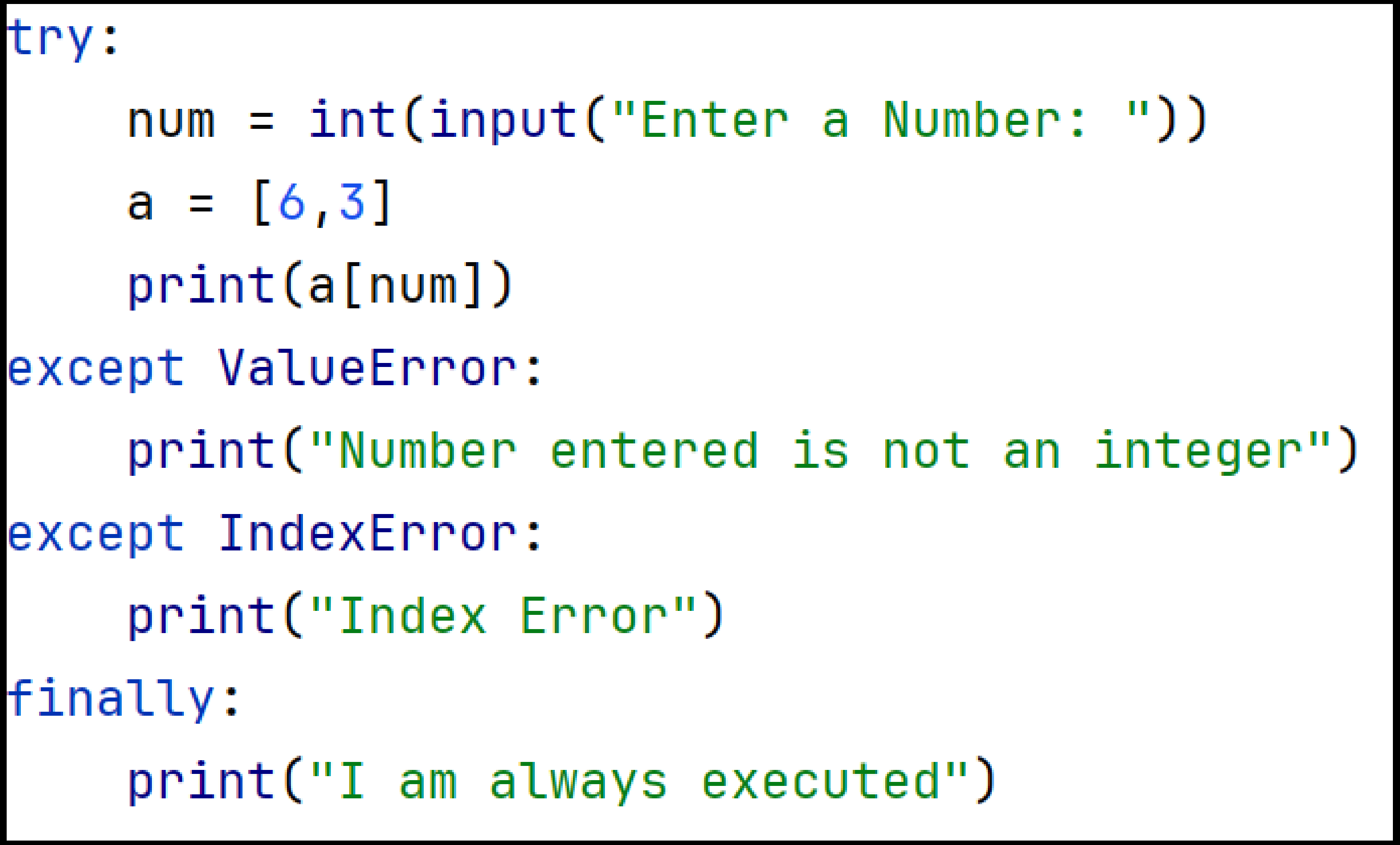
You can define as many exception blocks as you want, e.g. if you want to execute a special block of code for a special kind of error:.



```python
try:
    num = int(input("Enter a Number: "))
    a = [6,3]
    print(a[num])
except ValueError:
    print("Number entered is not an integer")
except IndexError:
    print("Index Error")
```

Finally Exceptions

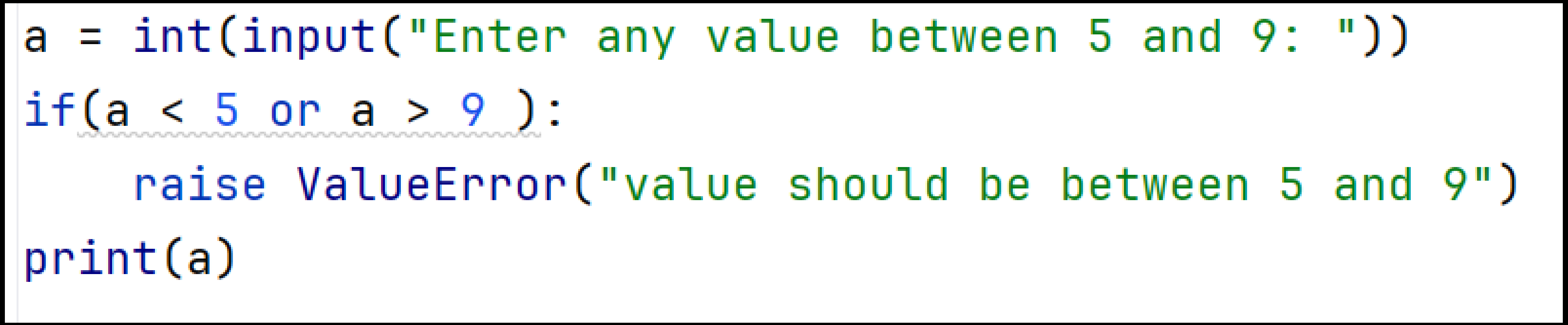
The finally block, if specified, will be executed regardless if the try block raises an error or not.



```
try:
    num = int(input("Enter a Number: "))
    a = [6,3]
    print(a[num])
except ValueError:
    print("Number entered is not an integer")
except IndexError:
    print("Index Error")
finally:
    print("I am always executed")
```

Raise an Exceptions

As a Python developer you can choose to throw an exception if a condition occurs.
To throw (or raise) an exception, use the raise keyword.

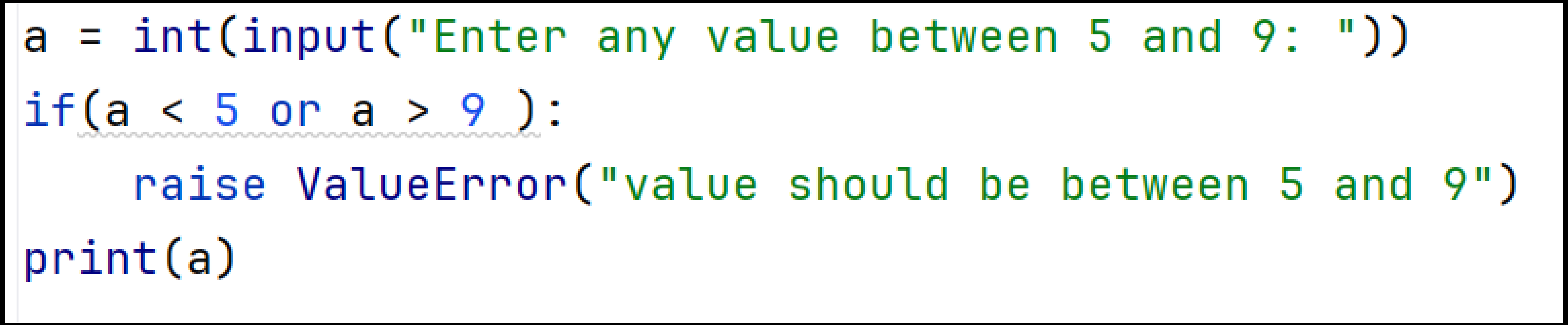


```python
a = int(input("Enter any value between 5 and 9: "))
if(a < 5 or a > 9 ):
    raise ValueError("value should be between 5 and 9")
print(a)
```

Black Origin

Raise an Exceptions

As a Python developer you can choose to throw an exception if a condition occurs.
To throw (or raise) an exception, use the raise keyword.



```python
a = int(input("Enter any value between 5 and 9: "))
if(a < 5 or a > 9 ):
    raise ValueError("value should be between 5 and 9")
print(a)
```