4CS001

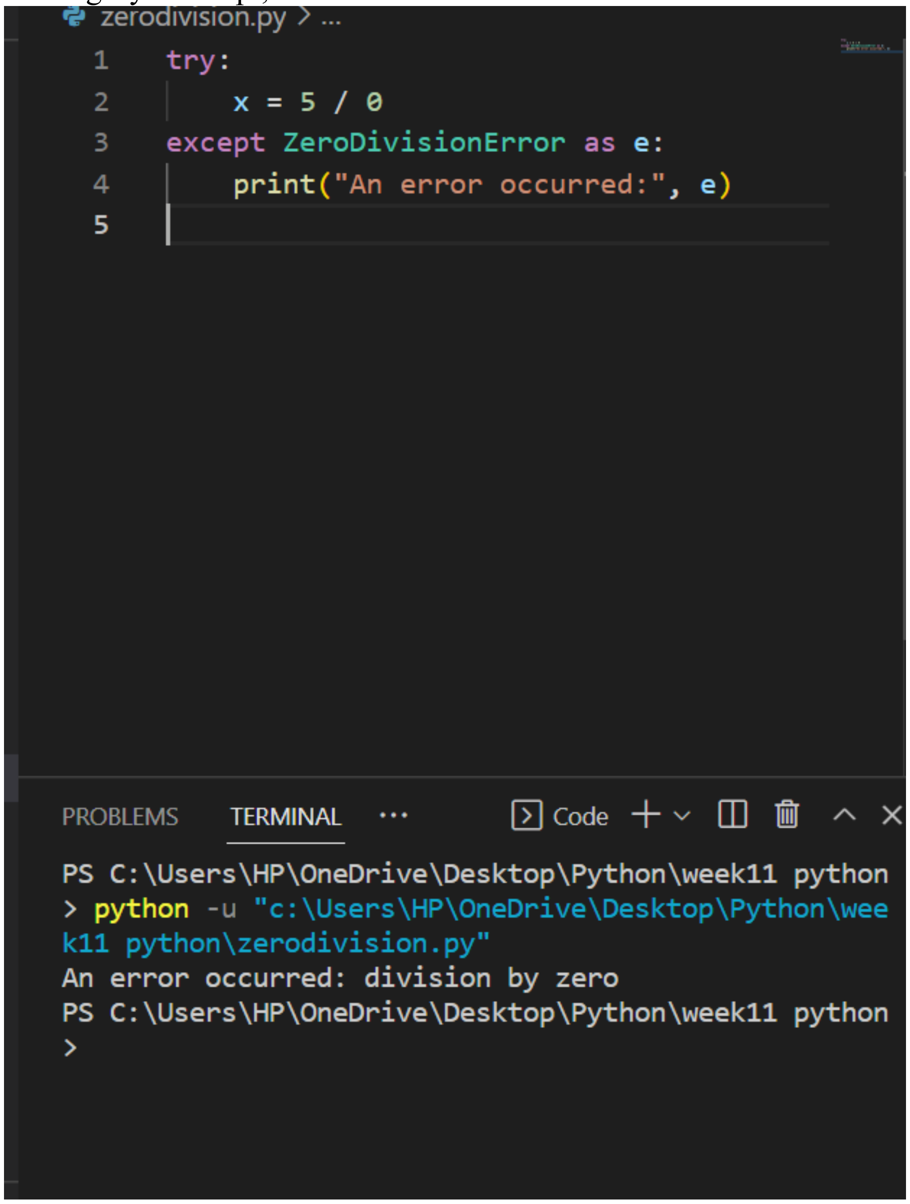
Python Workshop 11: Exception Handling

<u>Part 1</u>

1. How many except statements can a try-except block have?
   a) zero
   b) one
   c) one or more
   d) none of the above
2. When will the else part of try-except-else be executed?
   a) always
   b) when an exception occurs
   c) when no exception occurs
   d) when an exception occurs in to except block
3. Is the following Python code valid?
```
try:
    # Do something
except:
    # Do something
finally:
    # Do something
```
   a) no, there is no such thing as finally
   b) no, finally cannot be used with except
   c) no, finally must come before except
   d) yes
4. Is the following Python code valid?
```
try:
    # Do something
except:
    # Do something
else:
    # Do something
```
   a) no, there is no such thing as else
   b) no, else cannot be used with except
   c) no, else must come before except
   d) yes
5. When is the finally block executed?
   a) when there is no exception
   b) when there is an exception
   c) only if some condition that has been specified is satisfied
   d) always

Part 2

1. Using try…except, showcase the ZeroDivisionError.

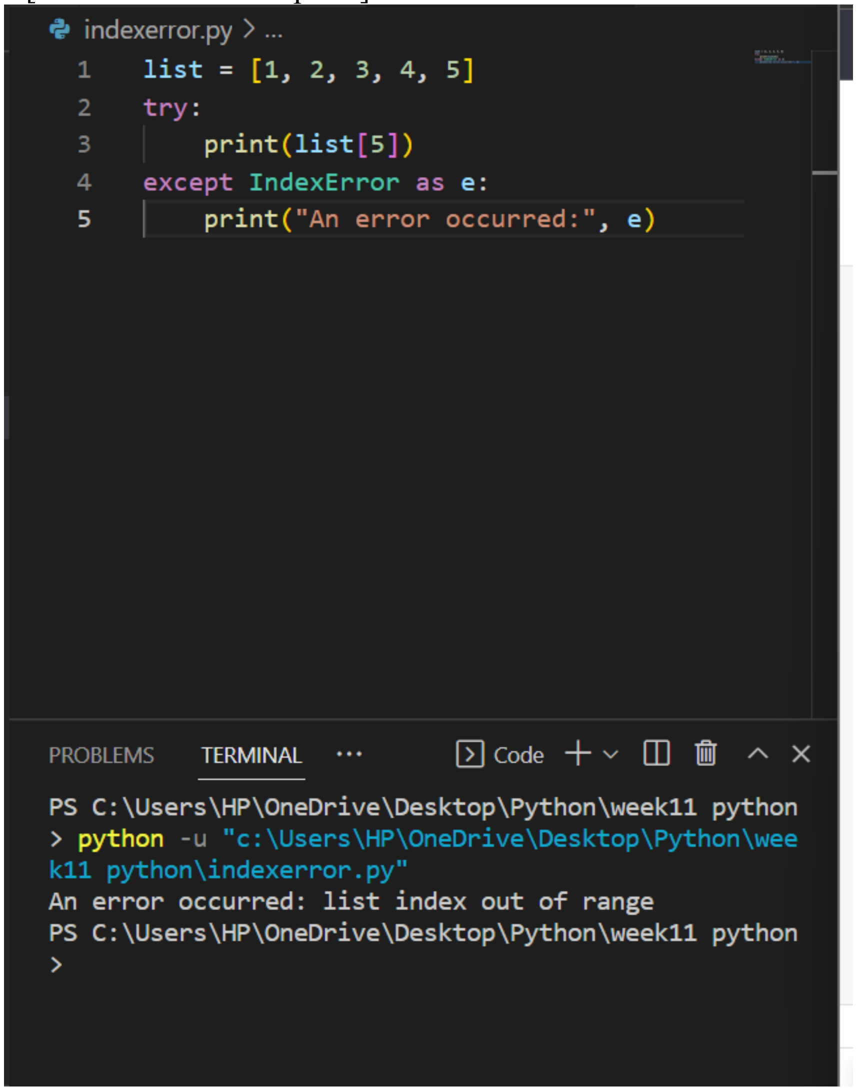


```python
try:
    x = 5 / 0
except ZeroDivisionError as e:
    print("An error occurred:", e)

```

```
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python
> python -u "c:\Users\HP\OneDrive\Desktop\Python\wee
k11 python\zerodivision.py"
An error occurred: division by zero
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python
>
```

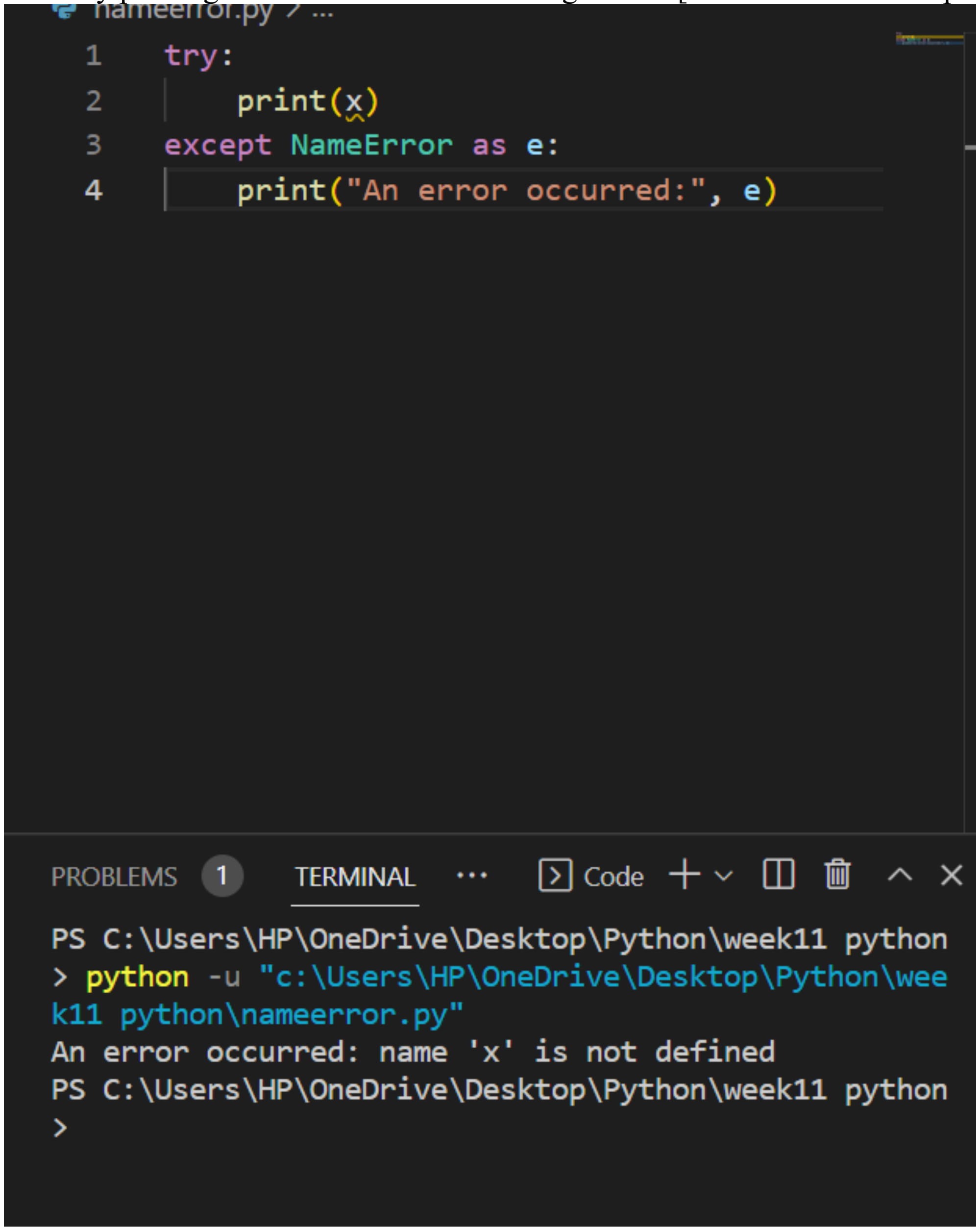
2. Create a simple list containing five elements and try to print the sixth element of the list. [use IndexError exception]



```python
list = [1, 2, 3, 4, 5]
try:
    print(list[5])
except IndexError as e:
    print("An error occurred:", e)
```

```
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python
> python -u "c:\Users\HP\OneDrive\Desktop\Python\wee
k11 python\indexerror.py"
An error occurred: list index out of range
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python
>
```

3. Try printing a variable without declaring it first. [use NameError exception]

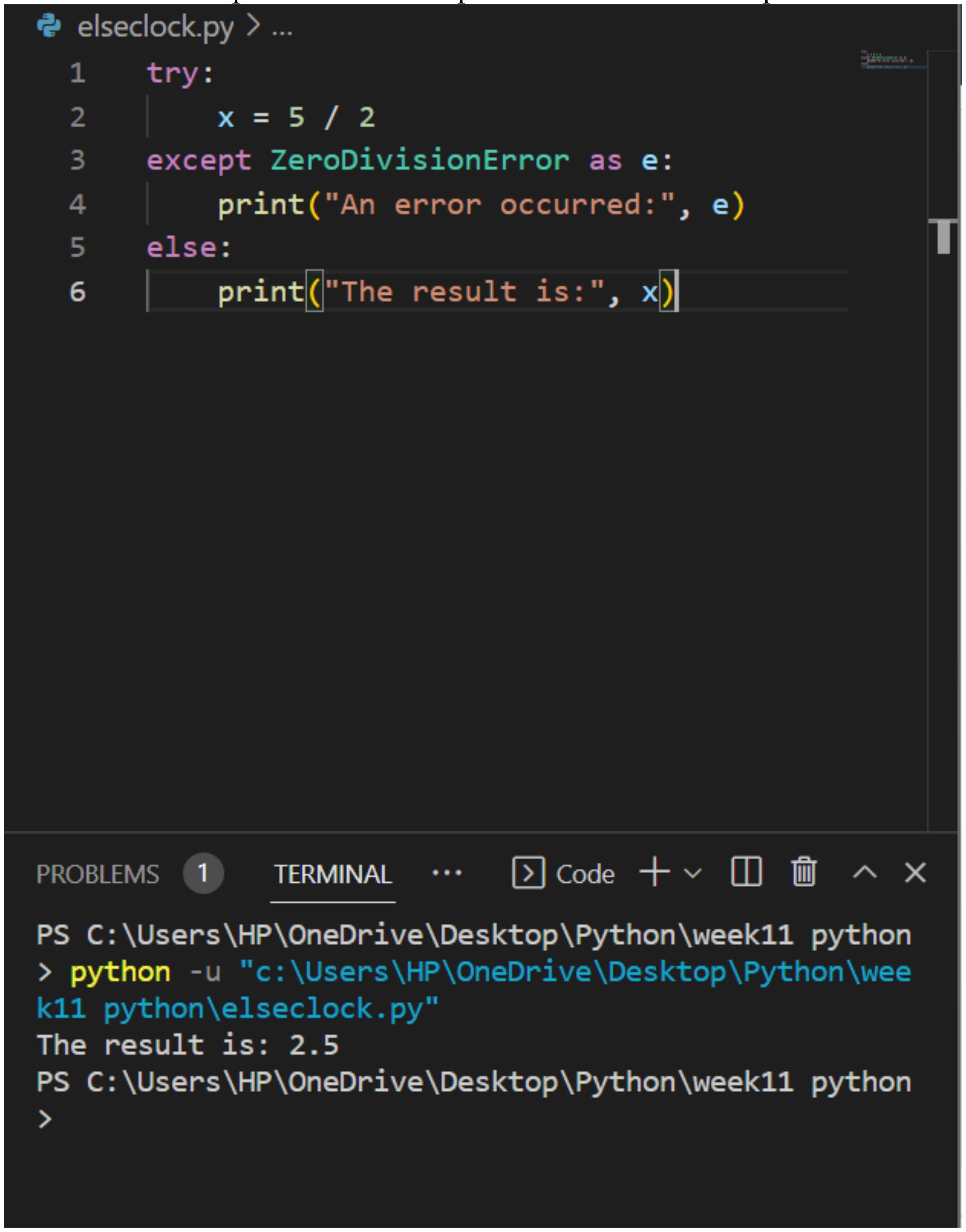


nameerror.py > ...

```python
try:
    print(x)
except NameError as e:
    print("An error occurred:", e)
```

```
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python
> python -u "c:\Users\HP\OneDrive\Desktop\Python\wee
k11 python\nameerror.py"
An error occurred: name 'x' is not defined
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python
>
```

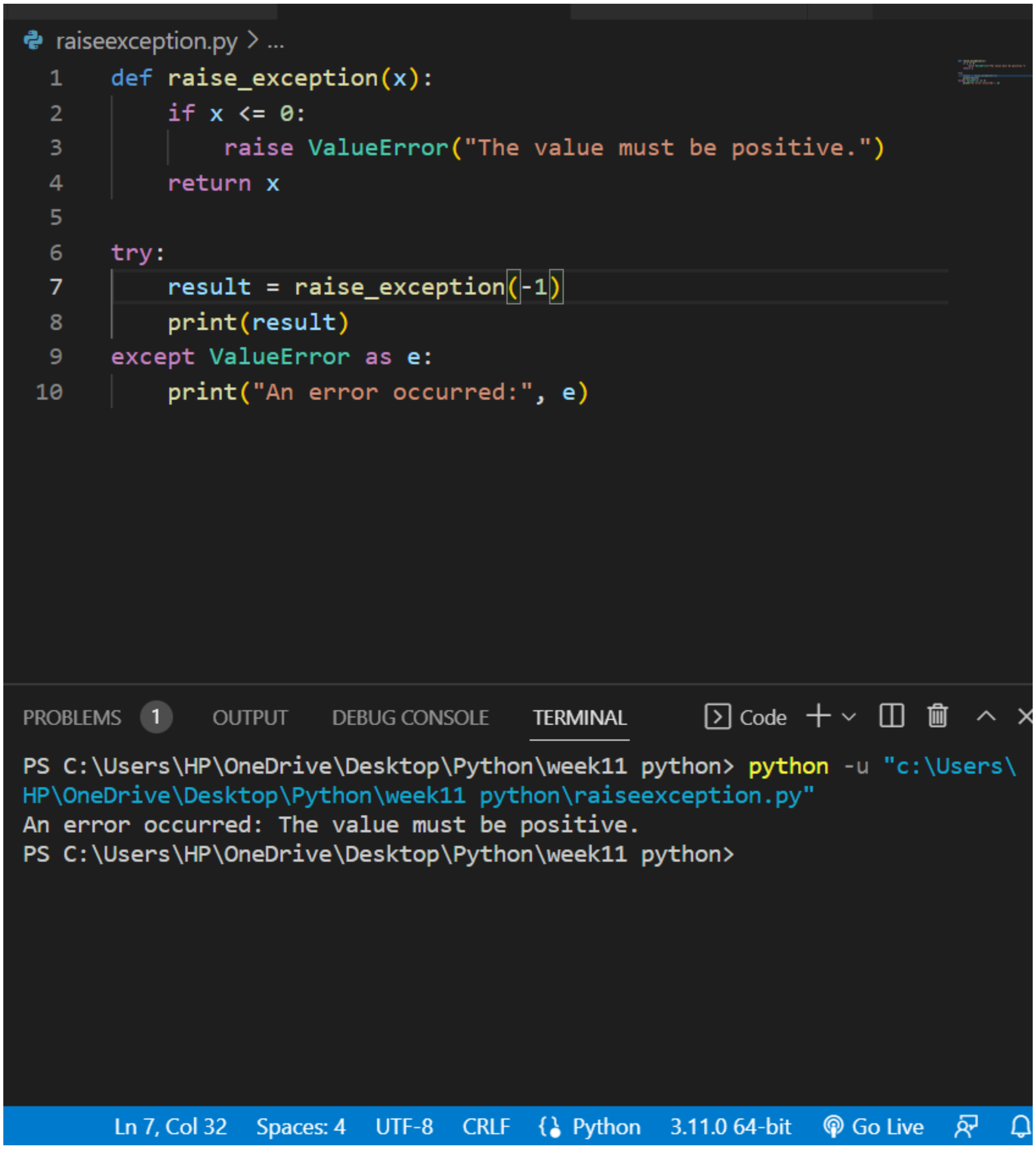
4. The ‘else’ in try…except…else statements is used to run the code on the else block if there are no exceptions in the ‘except’ block. Show an example.



```python
try:
    x = 5 / 2
except ZeroDivisionError as e:
    print("An error occurred:", e)
else:
    print("The result is:", x)
```

```
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python
> python -u "c:\Users\HP\OneDrive\Desktop\Python\wee
k11 python\elseclock.py"
The result is: 2.5
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python
>
```

5. In Python, we can choose to throw an exception if a condition occurs. To throw the exception, we use ‘raise’ keyword. Show an example.



```python
def raise_exception(x):
    if x <= 0:
        raise ValueError("The value must be positive.")
    return x

try:
    result = raise_exception(-1)
    print(result)
except ValueError as e:
    print("An error occurred:", e)
```

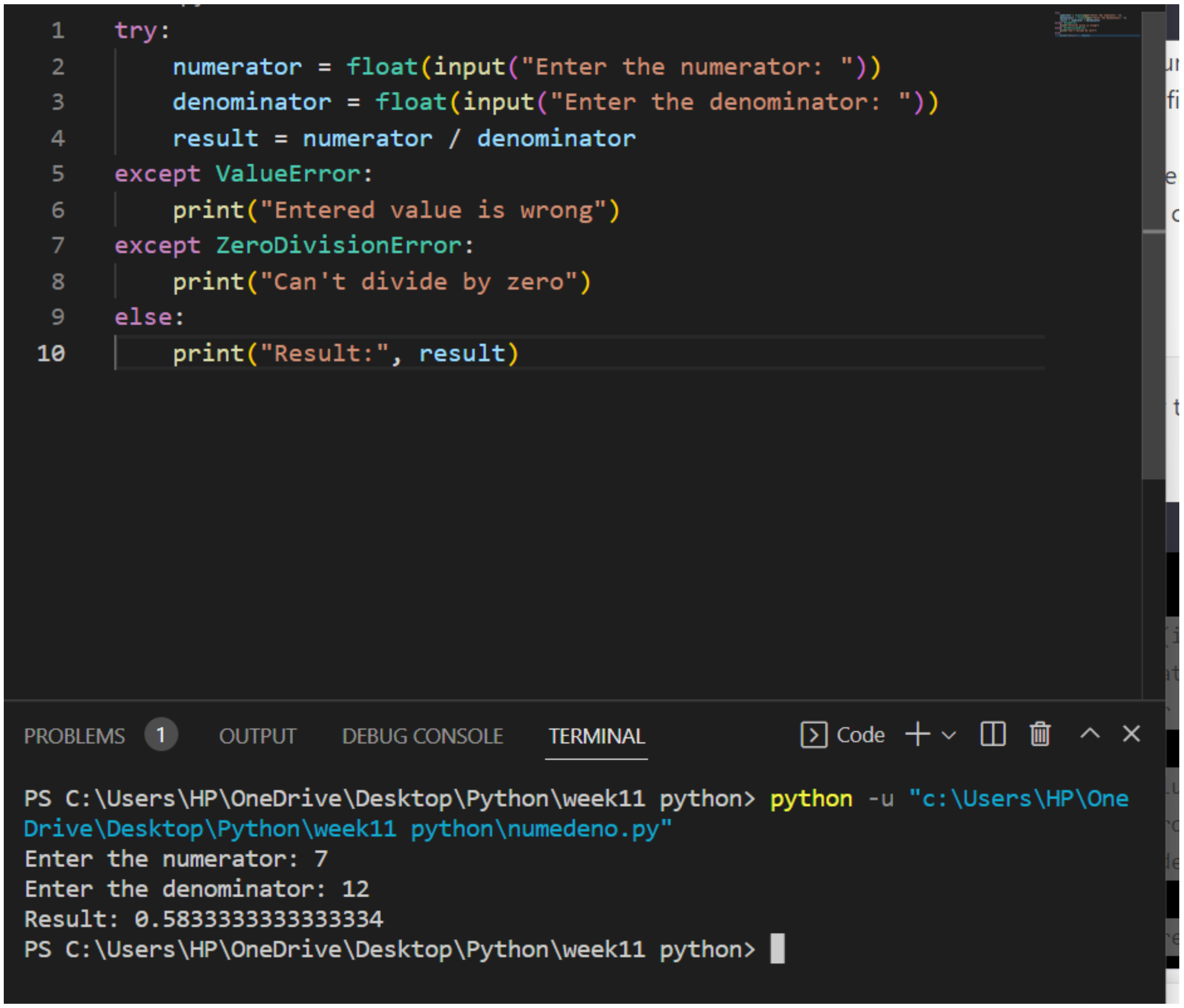
```
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python> python -u "c:\Users\
HP\OneDrive\Desktop\Python\week11 python\raiseexception.py"
An error occurred: The value must be positive.
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python>
```

Part 3

1. Ask the user for the numerator and denominator value; and perform division. If the user enters a number, the program will evaluate and produce the result.

   If the user enters a non-numeric value then, the try block will throw a `ValueError` exception, and we can catch that using a first catch block 'except ValueError' by printing the message 'Entered value is wrong'.

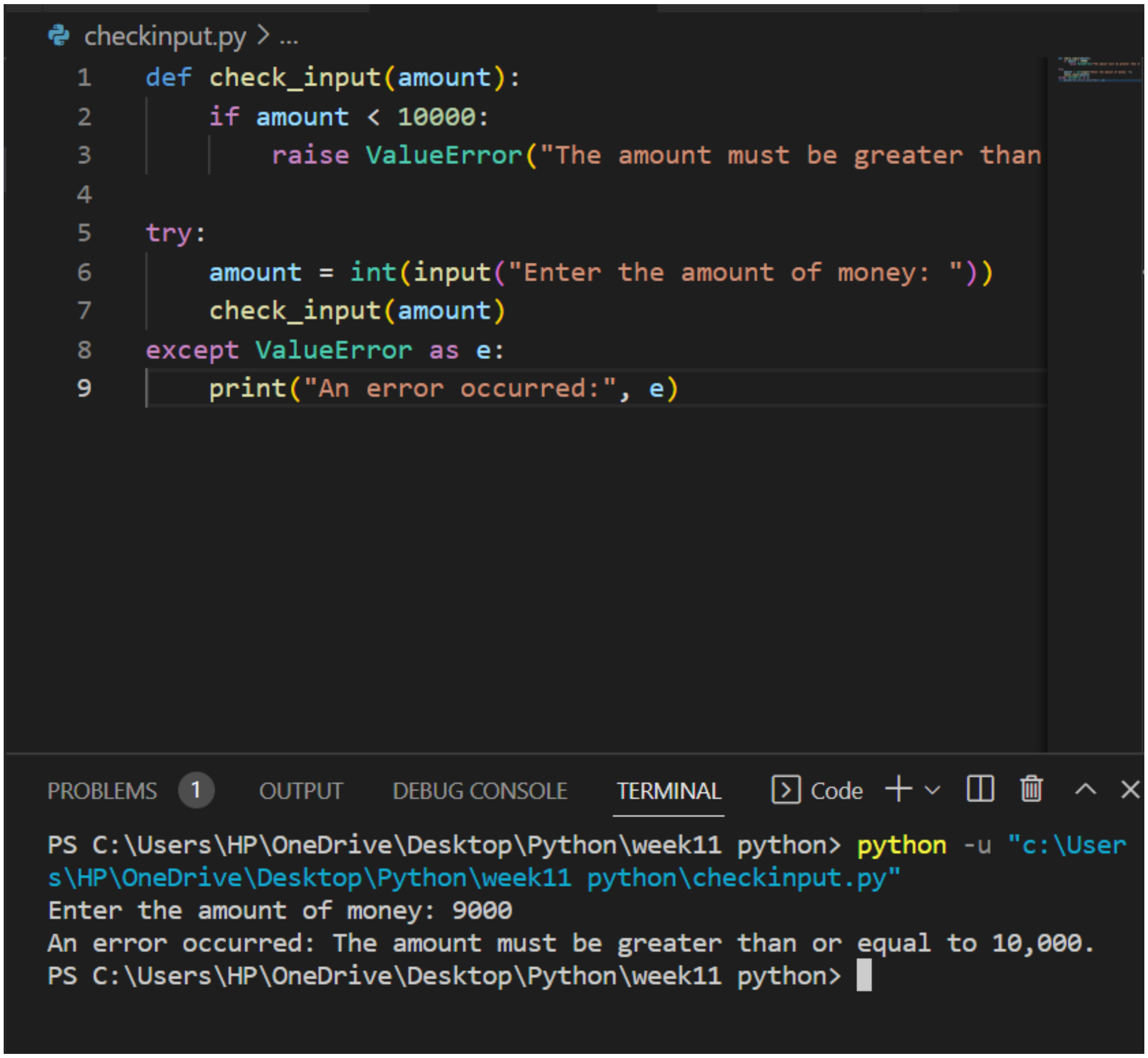
   And suppose the user enters the denominator as zero. In that case, the try block will throw a `ZeroDivisionError`, and we can catch that using a second catch block by printing the message 'Can't divide by zero'.



```python
try:
    numerator = float(input("Enter the numerator: "))
    denominator = float(input("Enter the denominator: "))
    result = numerator / denominator
except ValueError:
    print("Entered value is wrong")
except ZeroDivisionError:
    print("Can't divide by zero")
else:
    print("Result:", result)
```

```
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python> python -u "c:\Users\HP\One
Drive\Desktop\Python\week11 python\numedeno.py"
Enter the numerator: 7
Enter the denominator: 12
Result: 0.5833333333333334
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python>
```

2. Ask the user to enter an amount of money. In the try block, run a condition to check if the input value is less than 10 thousand; in which case raise a ValueError and print your message inside it. In the except block, catch the ValueError we previously raised and print the message inside it.



checkinput.py > ...

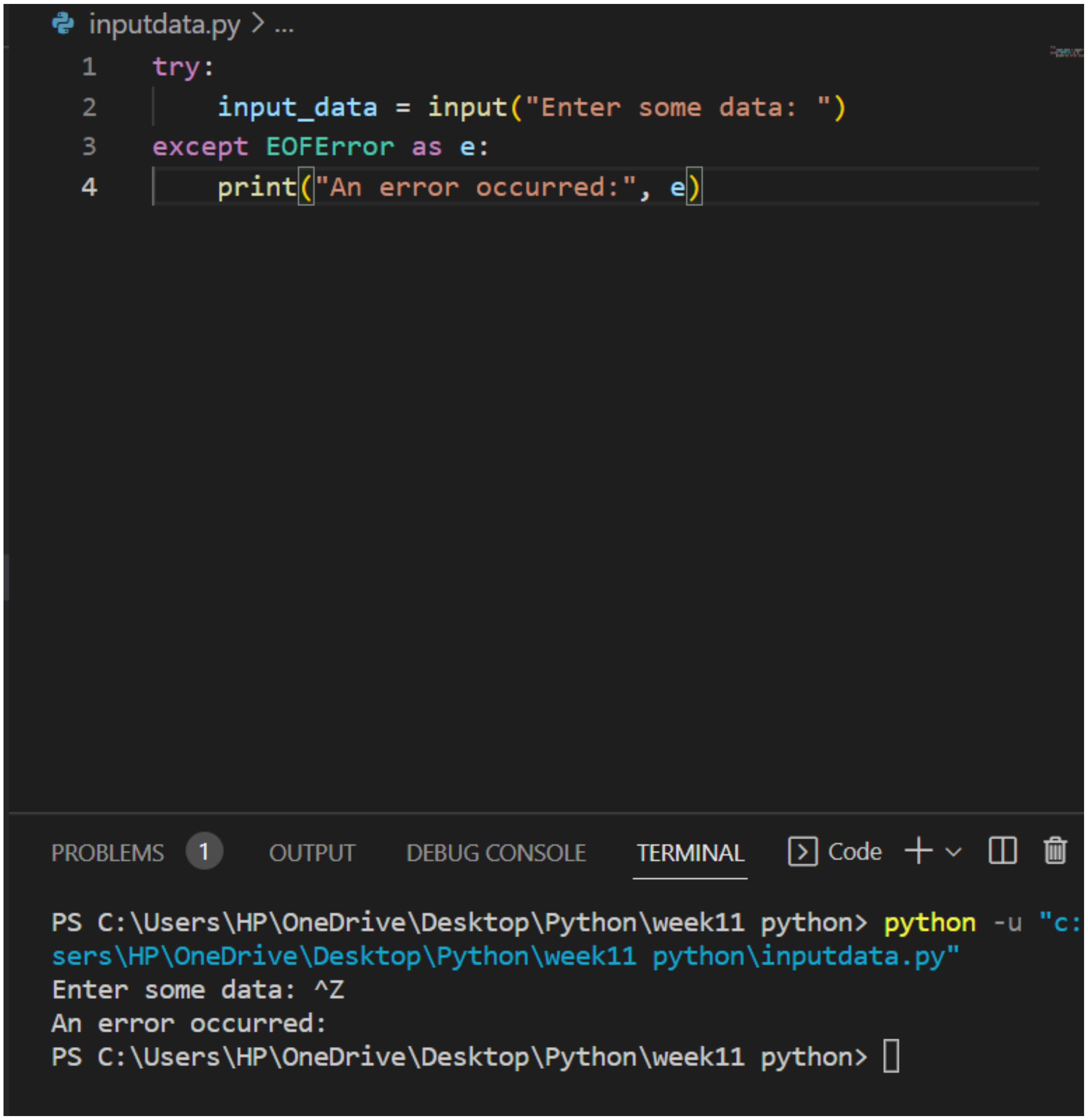
```python
def check_input(amount):
    if amount < 10000:
        raise ValueError("The amount must be greater than

try:
    amount = int(input("Enter the amount of money: "))
    check_input(amount)
except ValueError as e:
    print("An error occurred:", e)
```

PROBLEMS 1   OUTPUT   DEBUG CONSOLE   TERMINAL   Code

```
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python> python -u "c:\User
s\HP\OneDrive\Desktop\Python\week11 python\checkinput.py"
Enter the amount of money: 9000
An error occurred: The amount must be greater than or equal to 10,000.
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python> 
```

3. An EOFError is raised when built-in functions like input() hits an end-of-file condition (EOF) without reading any data. The file methods like readline() return an empty string when they hit EOF. Show an example.



```python
try:
    input_data = input("Enter some data: ")
except EOFError as e:
    print("An error occurred:", e)
```

```
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python> python -u "c:
sers\HP\OneDrive\Desktop\Python\week11 python\inputdata.py"
Enter some data: ^Z
An error occurred:
PS C:\Users\HP\OneDrive\Desktop\Python\week11 python>
```