Hi everyone, I don’t know if my understanding is correct.

• There is something wrong with the arguments the function is getting; a precondition is violated.

Argument getting wrong, A mistake make at the “beginning” of the function, it happens in the step or next step when we call the function.

def Return\_Greater\_value(a,b):  
 if a > b:  
 return a  
 elif b > a:  
 return b  
 else:  
 return "a and b are equal"  
  
print(Return\_Greater\_value(1,b))

TypeError: '>' not supported between instances of 'int' and 'str'

In this example we give an invalid type of value to parameter b.

• There is something wrong with the function; a postcondition is violated.

A mistake made at the body of function.

def Return\_Greater\_value(a,b):  
 if a > b:  
 return a  
 elif b > a  
 return b  
 else:  
 return "a and b are equal"  
  
print(Return\_Greater\_value(1,2))

SyntaxError: expected ':'

In this example, we miss a colon at line 4, the error occurred at the main body of the function.

• There is something wrong with the return value or the way it is being used.

An error occurred at the end of the function, which related to return value.

def Return\_Greater\_value(a,b):  
 if a > b:  
 return a  
 elif b > a:  
 return b  
  
  
print(Return\_Greater\_value(1,1))

>>> None

In this example, I didn’t write a case when two number value equal. This bug is related to return value.

Define "precondition" and "postcondition" as part of your description.

Precondition is the statement implemented before function main body attempt to execute. (Function calling and function definition)

Post condition is from next step of function definition statement to precede step of return statement. (Function main body)

References:  
  
Downey, A. (2015). Think Python: How to think like a computer scientist. Green Tea Press

My question is, in my first example, we give an invalid value to parameter, but the error occurred at the body of the function, is this precondition violated or postcondition violated?