



```
3) assert squareIt(2)==4,"The square of 2 should be 4"
4) assert squareIt(3)==9,"The square of 3 should be 9"
5) assert squareIt(4)==16,"The square of 4 should be 16"
6) print(squareIt(2))
7) print(squareIt(3))
8) print(squareIt(4))
9)
10) D:\Python_classes>py test.py
11) Traceback (most recent call last):
12) File "test.py", line 4, in <module>
13)   assert squareIt(3)==9,"The square of 3 should be 9"
14) AssertionError: The square of 3 should be 9
15)
16) def squareIt(x):
17)     return x*x
18) assert squareIt(2)==4,"The square of 2 should be 4"
19) assert squareIt(3)==9,"The square of 3 should be 9"
20) assert squareIt(4)==16,"The square of 4 should be 16"
21) print(squareIt(2))
22) print(squareIt(3))
23) print(squareIt(4))
24)
25) Output
26) 4
27) 9
28) 16
```

## Exception Handling vs Assertions:

Assertions concept can be used to alert programmer to resolve development time errors.

Exception Handling can be used to handle runtime errors.