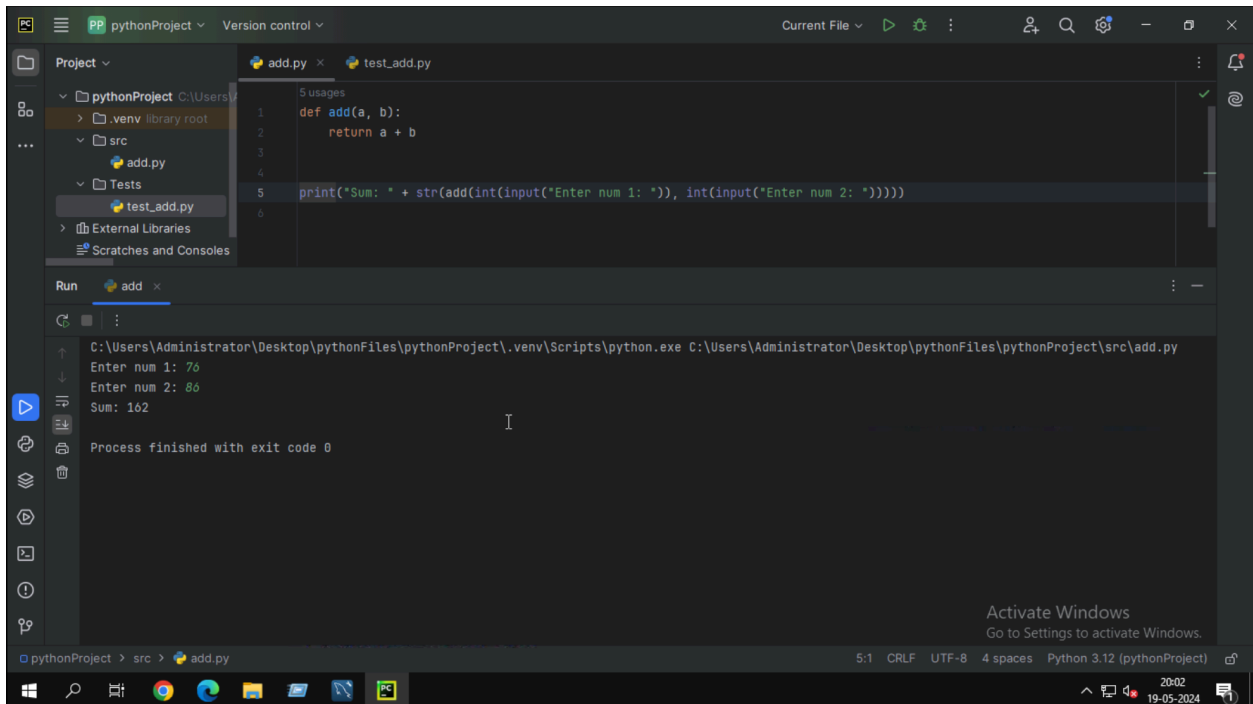


**Task 1:** Write a simple Python function to add two numbers and then write a pytest test case to test this function.

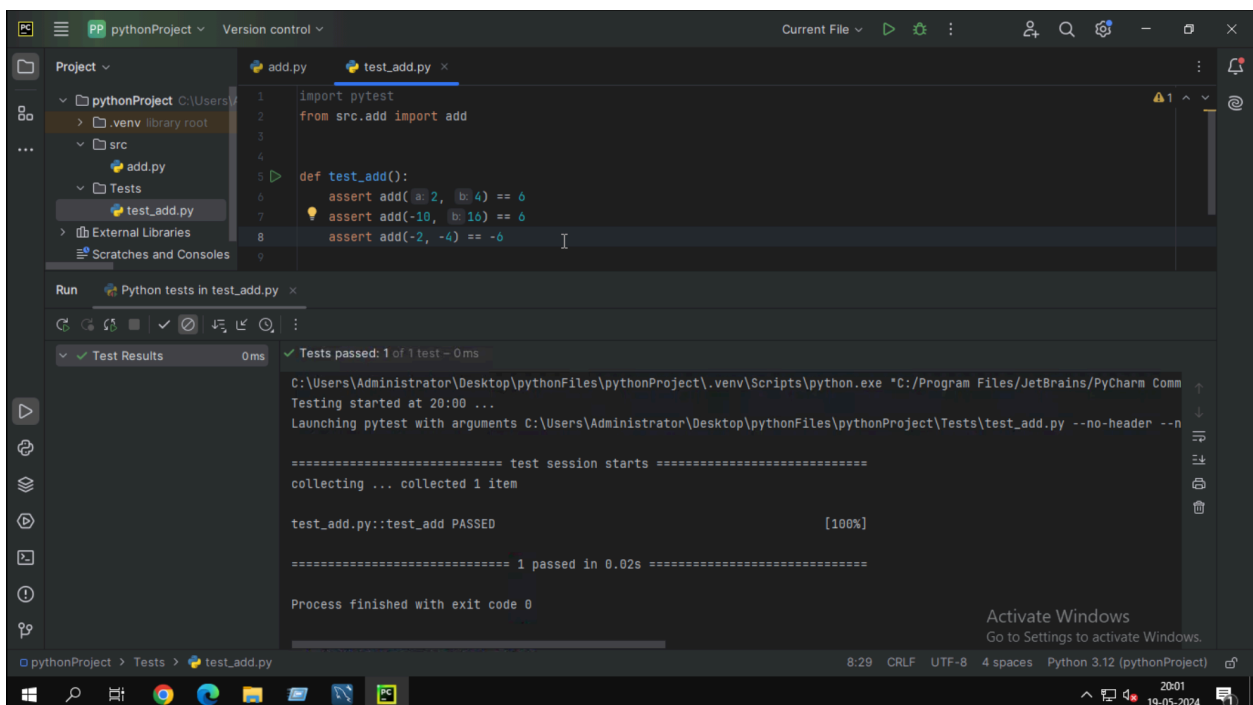


The screenshot shows the PyCharm IDE with a project named 'pythonProject'. The file explorer on the left shows the project structure: 'pythonProject' (C:\Users\Administrator\pythonFiles\pythonProject) containing a '.venv' directory, a 'src' directory, and a 'Tests' directory. The 'src' directory contains 'add.py' and 'test\_add.py'. The 'Tests' directory contains 'test\_add.py'. The 'Run' window shows the execution of 'add.py'.

```
1 def add(a, b):  
2     return a + b  
3  
4  
5 print("Sum: " + str(add(int(input("Enter num 1: ")), int(input("Enter num 2: "))))  
6
```

Run: add

```
C:\Users\Administrator\Desktop\pythonFiles\pythonProject\.venv\Scripts\python.exe C:\Users\Administrator\Desktop\pythonFiles\pythonProject\src\add.py  
Enter num 1: 76  
Enter num 2: 86  
Sum: 162  
Process finished with exit code 0
```



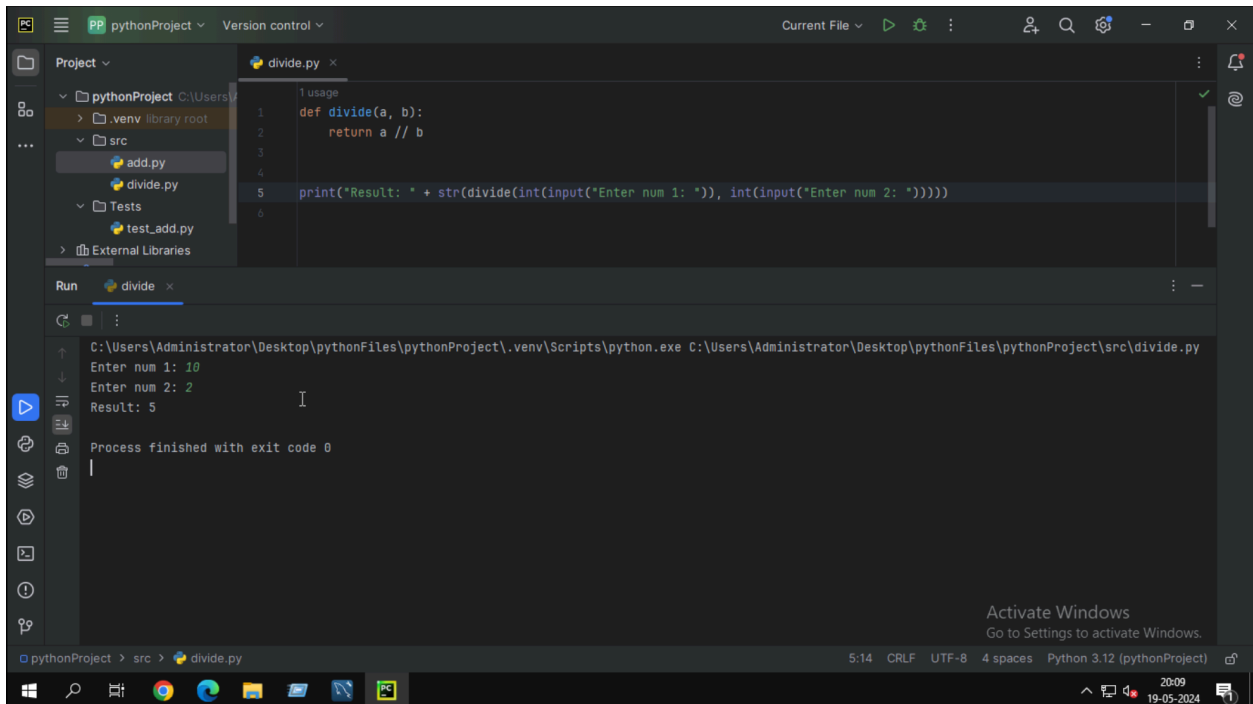
The screenshot shows the PyCharm IDE with the same project structure. The 'test\_add.py' file is open in the editor. The 'Run' window shows the execution of 'test\_add.py'.

```
1 import pytest  
2 from src.add import add  
3  
4  
5 def test_add():  
6     assert add(2, 4) == 6  
7     assert add(-10, 16) == 6  
8     assert add(-2, -4) == -6  
9
```

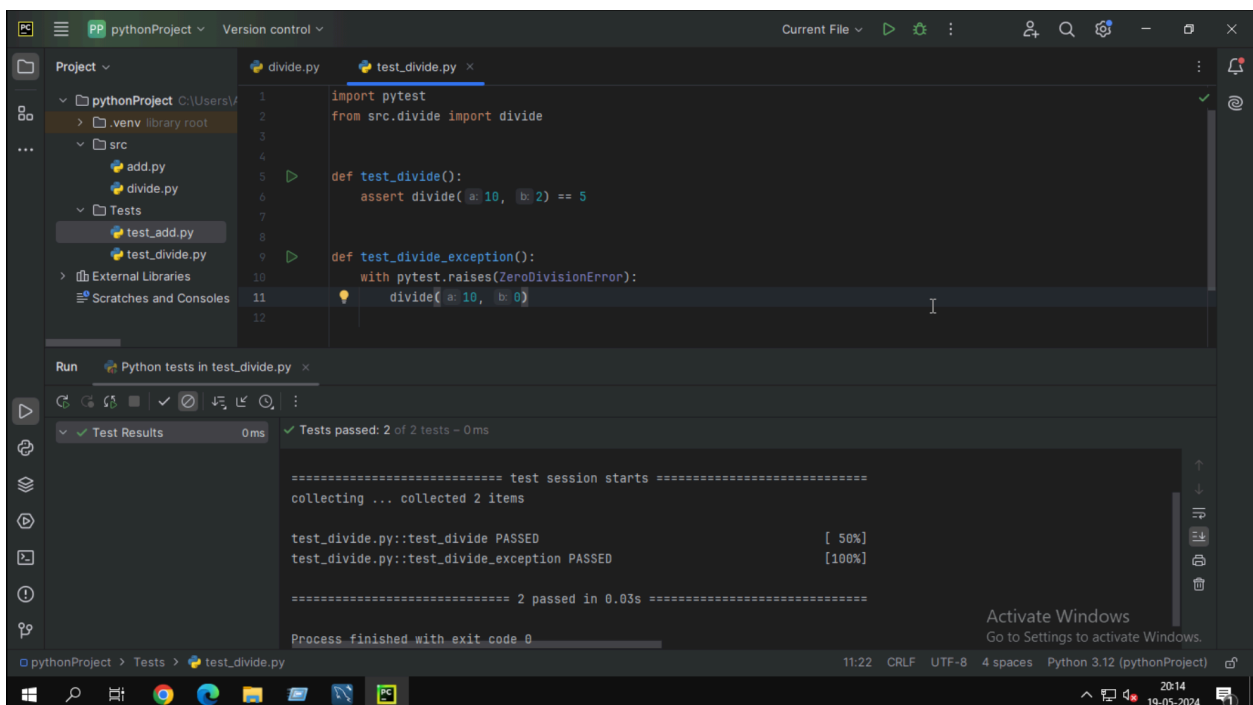
Run: Python tests in test\_add.py

```
C:\Users\Administrator\Desktop\pythonFiles\pythonProject\.venv\Scripts\python.exe "C:/Program Files/JetBrains/PyCharm Comm  
Testing started at 20:00 ...  
Launching pytest with arguments C:\Users\Administrator\Desktop\pythonFiles\pythonProject\Tests\test_add.py --no-header --n  
===== test session starts =====  
collecting ... collected 1 item  
  
test_add.py::test_add PASSED [100%]  
  
===== 1 passed in 0.02s =====  
Process finished with exit code 0
```

**Task 2:** Write a pytest test case to check if an exception is raised for a function that divides two numbers.

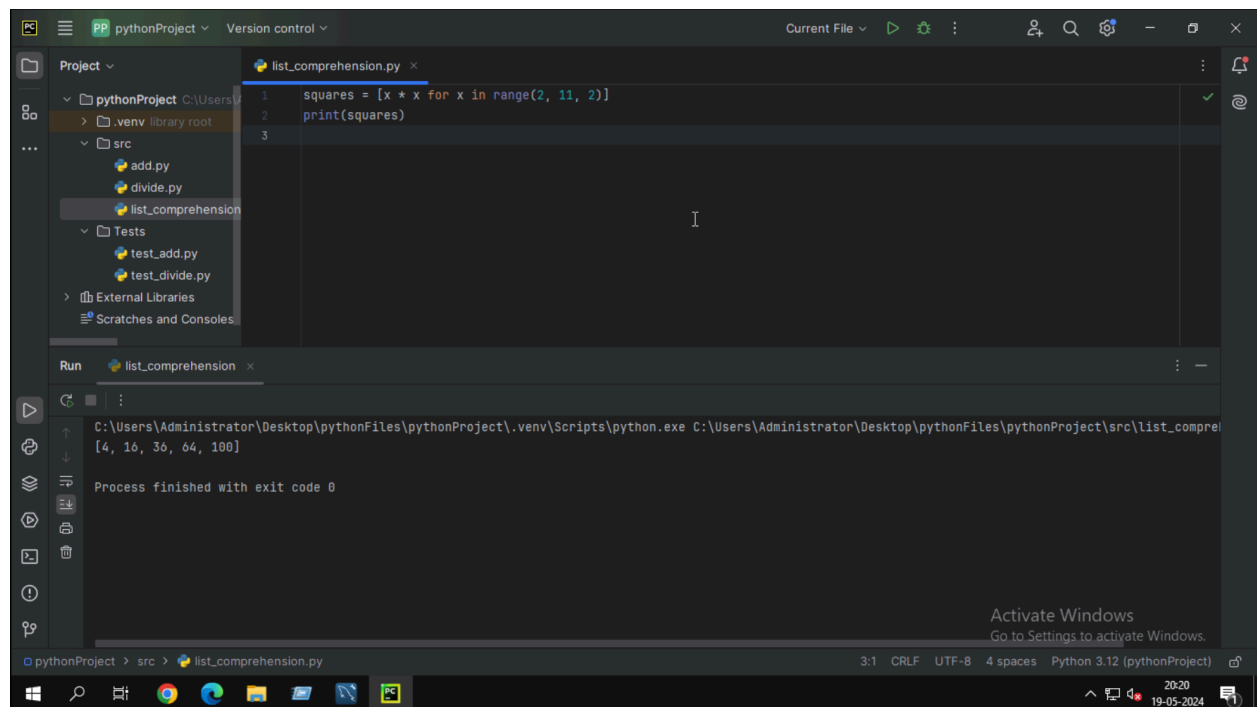


```
pythonProject Version control
Project
  pythonProject C:\Users\...
  .venv library root
  src
    add.py
    divide.py
    Tests
    test_add.py
    test_divide.py
  External Libraries
Run divide
C:\Users\Administrator\Desktop\pythonFiles\pythonProject\.venv\Scripts\python.exe C:\Users\Administrator\Desktop\pythonFiles\pythonProject\src\divide.py
Enter num 1: 10
Enter num 2: 2
Result: 5
Process finished with exit code 0
pythonProject > src > divide.py 5:14 CRLF UTF-8 4 spaces Python 3.12 (pythonProject) 20:09 19-05-2024
```



```
pythonProject Version control
Project
  pythonProject C:\Users\...
  .venv library root
  src
    add.py
    divide.py
    Tests
    test_add.py
    test_divide.py
    test_divide_exception.py
  External Libraries
Run Python tests in test_divide.py
Test Results 0ms Tests passed: 2 of 2 tests - 0ms
===== test session starts =====
collecting ... collected 2 items
test_divide.py::test_divide PASSED [ 50%]
test_divide.py::test_divide_exception PASSED [100%]
===== 2 passed in 0.03s =====
Process finished with exit code 0
pythonProject > Tests > test_divide.py 11:22 CRLF UTF-8 4 spaces Python 3.12 (pythonProject) 20:14 19-05-2024
```

**Task 3:** Create a list comprehension in Python to generate squares of even numbers between 1 to 10.



The screenshot shows a Python IDE with a project named 'pythonProject'. The file explorer on the left shows the project structure, including a 'src' directory with files 'add.py', 'divide.py', 'list\_comprehension.py', and 'Tests' containing 'test\_add.py' and 'test\_divide.py'. The main editor window displays the file 'list\_comprehension.py' with the following code:

```
1 squares = [x * x for x in range(2, 11, 2)]  
2 print(squares)  
3
```

The 'Run' panel at the bottom shows the execution output for 'list\_comprehension.py':

```
C:\Users\Administrator\Desktop\pythonFiles\pythonProject\.venv\Scripts\python.exe C:\Users\Administrator\Desktop\pythonFiles\pythonProject\src\list_compre!  
[4, 16, 36, 64, 100]  
Process finished with exit code 0
```

The status bar at the bottom indicates the file encoding is UTF-8, line endings are CRLF, and it uses 4 spaces for indentation. The Python version is 3.12.