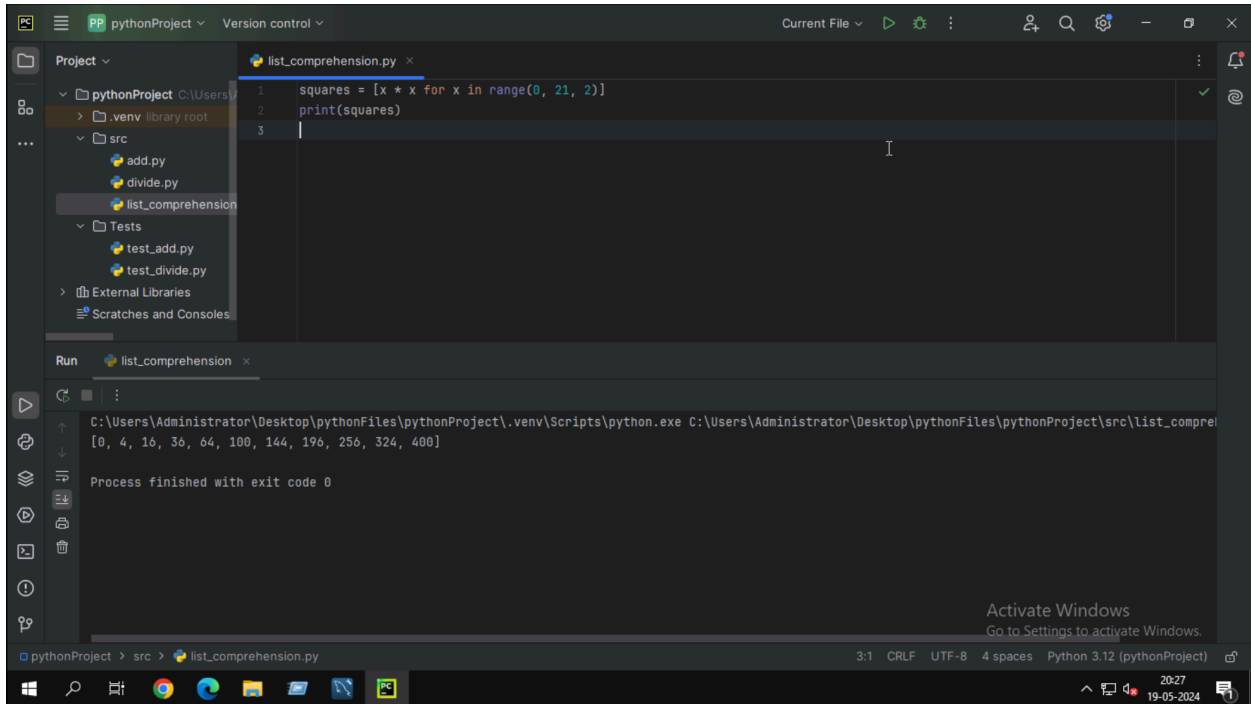


Comprehension, Iterators, Generator, Decorators , Lambda function

Task 1: Write a list comprehension that creates a list of squares for all even numbers between 0 and 20.



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 like 'add.py', 'divide.py', and 'list_comprehension.py'. The main editor window displays the following code in 'list_comprehension.py':

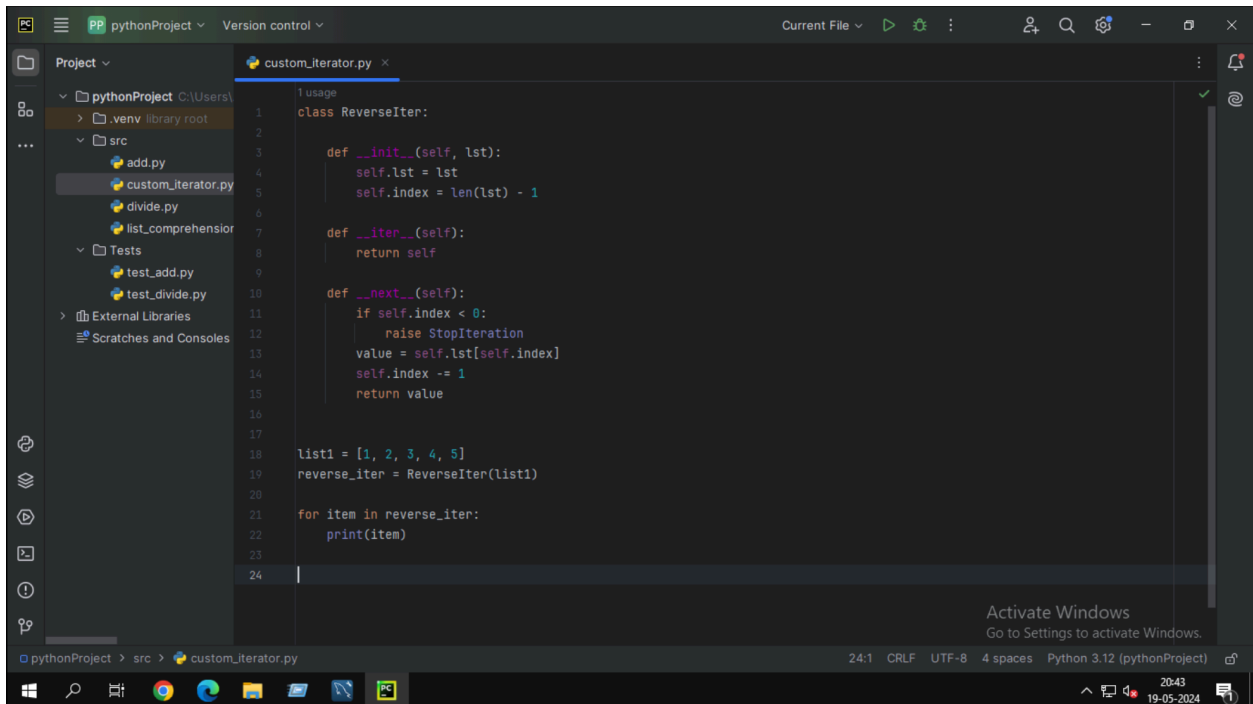
```
1 squares = [x * x for x in range(0, 21, 2)]
2 print(squares)
3
```

The 'Run' panel at the bottom shows the execution output:

```
C:\Users\Administrator\Desktop\pythonFiles\pythonProject\.venv\Scripts\python.exe C:\Users\Administrator\Desktop\pythonFiles\pythonProject\src\list_comprel
[0, 4, 16, 36, 64, 100, 144, 196, 256, 324, 400]
Process finished with exit code 0
```

The status bar at the bottom indicates the file encoding is UTF-8, the line length is 3:1, and the Python version is 3.12.

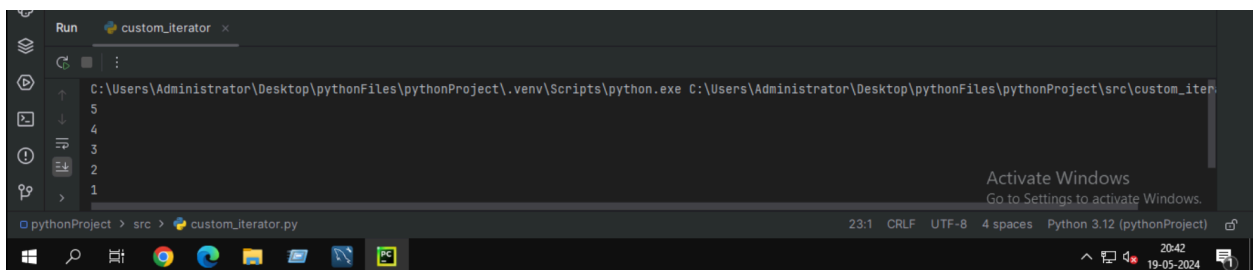
Task 2: Create a custom iterator class `ReverseIter`, which takes a list and iterates it from the reverse direction.



The screenshot shows a code editor with a project named 'pythonProject'. The file explorer on the left shows the project structure, including a 'src' directory with files like 'add.py', 'custom_iterator.py', 'divide.py', 'list_comprehension.py', 'test_add.py', and 'test_divide.py'. The main editor window displays the code for 'custom_iterator.py'.

```
1 usage
2 class ReverseIter:
3
4     def __init__(self, lst):
5         self.lst = lst
6         self.index = len(lst) - 1
7
8     def __iter__(self):
9         return self
10
11     def __next__(self):
12         if self.index < 0:
13             raise StopIteration
14         value = self.lst[self.index]
15         self.index -= 1
16         return value
17
18 list1 = [1, 2, 3, 4, 5]
19 reverse_iter = ReverseIter(list1)
20
21 for item in reverse_iter:
22     print(item)
23
24
```

The status bar at the bottom indicates the file encoding is UTF-8, the line ending is CRLF, and the Python version is 3.12.



The screenshot shows the 'Run' console window for the 'custom_iterator' script. The command executed is:

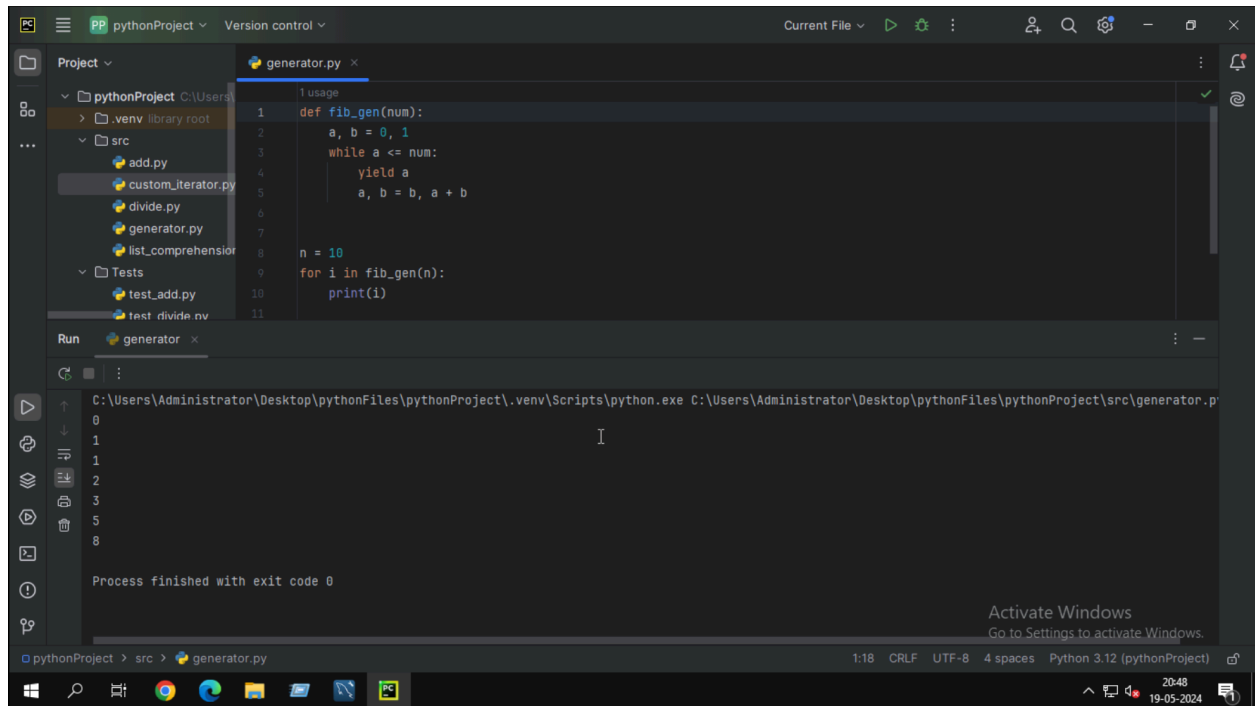
```
C:\Users\Administrator\Desktop\pythonFiles\pythonProject\.venv\Scripts\python.exe C:\Users\Administrator\Desktop\pythonFiles\pythonProject\src\custom_iter
```

The output shows the list elements in reverse order:

```
5
4
3
2
1
```

The status bar at the bottom indicates the file encoding is UTF-8, the line ending is CRLF, and the Python version is 3.12.

Task 3: Write a generator function `fib_gen` that yields the Fibonacci sequence up to a given number `n`.



The screenshot shows a Python IDE with a project named 'pythonProject'. The file explorer on the left shows a directory structure with files like `add.py`, `custom_iterator.py`, `divide.py`, `generator.py`, `list_comprehension.py`, and test files. The main editor displays the `generator.py` file with the following code:

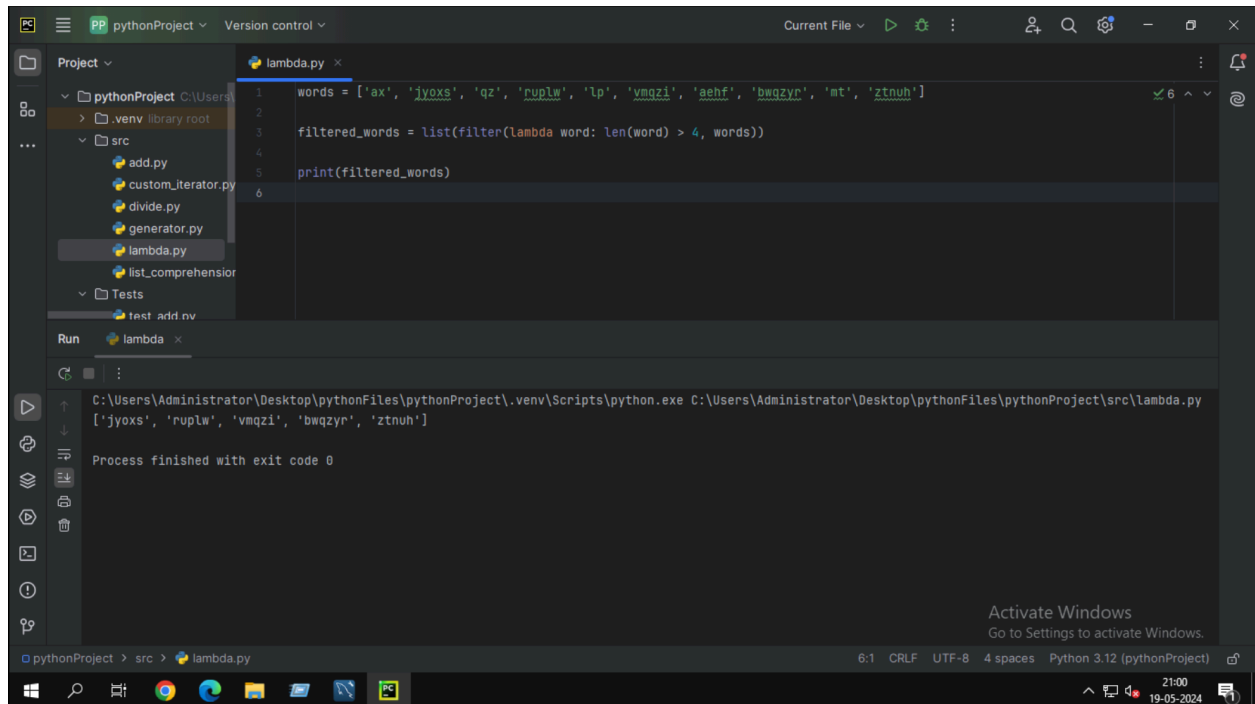
```
1 def fib_gen(num):
2     a, b = 0, 1
3     while a <= num:
4         yield a
5         a, b = b, a + b
6
7 n = 10
8 for i in fib_gen(n):
9     print(i)
```

The Run console at the bottom shows the command used to execute the script and the output of the program:

```
C:\Users\Administrator\Desktop\pythonFiles\pythonProject\.venv\Scripts\python.exe C:\Users\Administrator\Desktop\pythonFiles\pythonProject\src\generator.py
0
1
1
2
3
5
8
Process finished with exit code 0
```

The status bar at the bottom indicates the file encoding is UTF-8, the line length is 118, and the Python version is 3.12.

Task 4: Create a filter using a lambda function that extracts all words from a list that have more than 4 characters.



The screenshot shows a Python IDE with a project named 'pythonProject'. The file explorer on the left shows a directory structure with a 'src' folder containing several Python files, including 'lambda.py'. The main editor window displays the code in 'lambda.py':

```
1 words = ['ax', 'jyoxs', 'qz', 'ruplw', 'lp', 'vmqzi', 'aehf', 'bwqzyn', 'mt', 'ztnuh']
2
3 filtered_words = list(filter(lambda word: len(word) > 4, words))
4
5 print(filtered_words)
6
```

The Run window at the bottom shows the execution of the code, displaying the output: ['jyoxs', 'ruplw', 'vmqzi', 'bwqzyn', 'ztnuh']. The status bar at the bottom indicates the file encoding is UTF-8, the line ending is CRLF, and the Python version is 3.12.