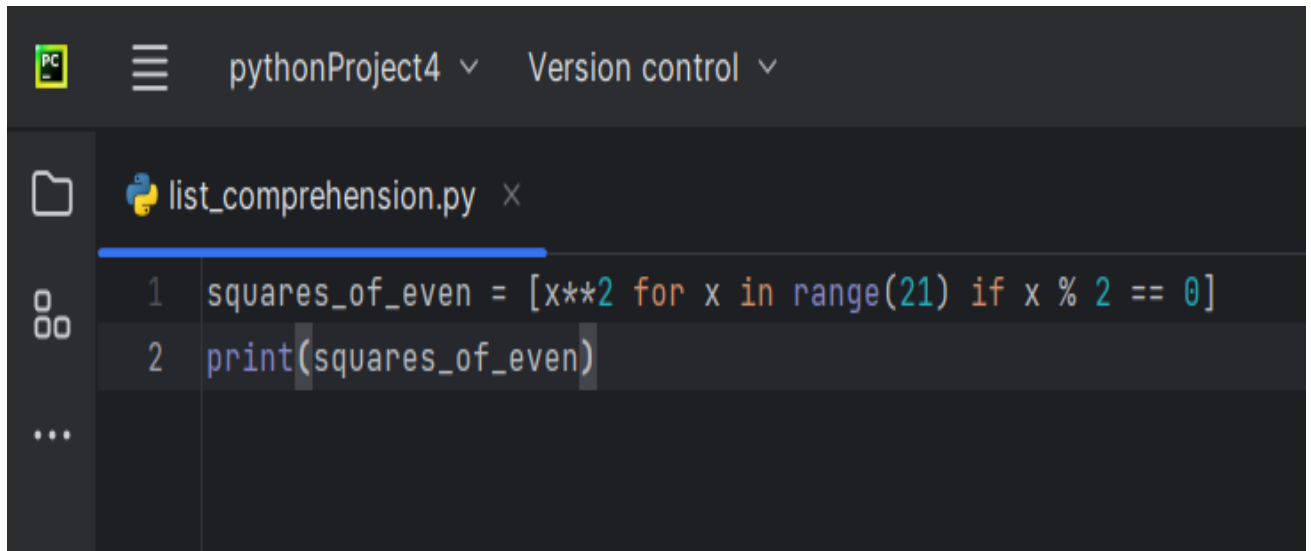


## Python Programming Day 1 Assignments

### 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.**

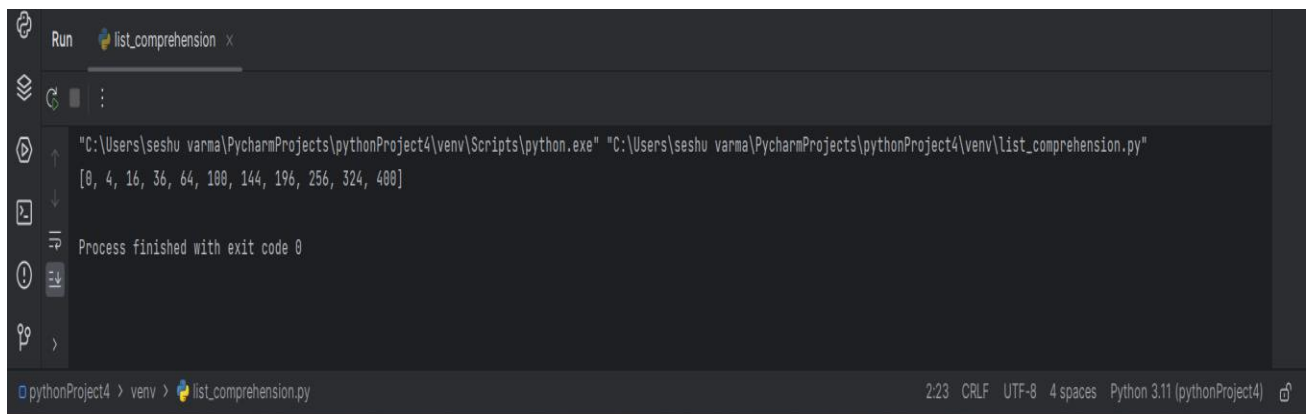
*Code Execution:*



The screenshot shows the PyCharm IDE interface. The top bar displays 'pythonProject4' and 'Version control'. The left sidebar shows a file explorer with 'list\_comprehension.py' selected. The main editor window contains the following Python code:

```
1 squares_of_even = [x**2 for x in range(21) if x % 2 == 0]
2 print(squares_of_even)
```

*Expected Output:*



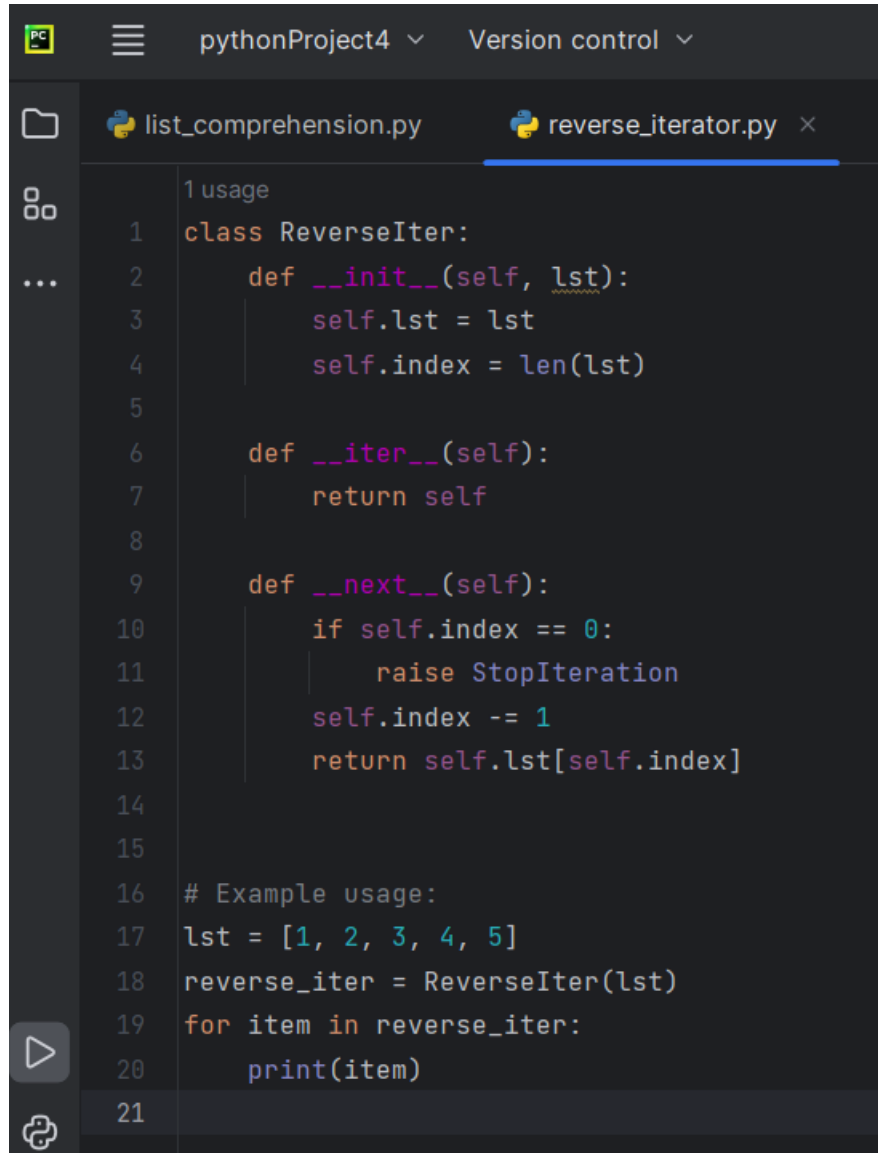
The screenshot shows the PyCharm Run console. The top bar indicates 'Run' and 'list\_comprehension'. The console output shows the command executed and the resulting list of squares:

```
"C:\Users\seshu varma\PycharmProjects\pythonProject4\venv\Scripts\python.exe" "C:\Users\seshu varma\PycharmProjects\pythonProject4\venv\list_comprehension.py"
[0, 4, 16, 36, 64, 100, 144, 196, 256, 324, 400]
```

Below the output, it states 'Process finished with exit code 0'. The bottom status bar shows 'pythonProject4 > venv > list\_comprehension.py' and '2:23 CRLF UTF-8 4 spaces Python 3.11 (pythonProject4)'.

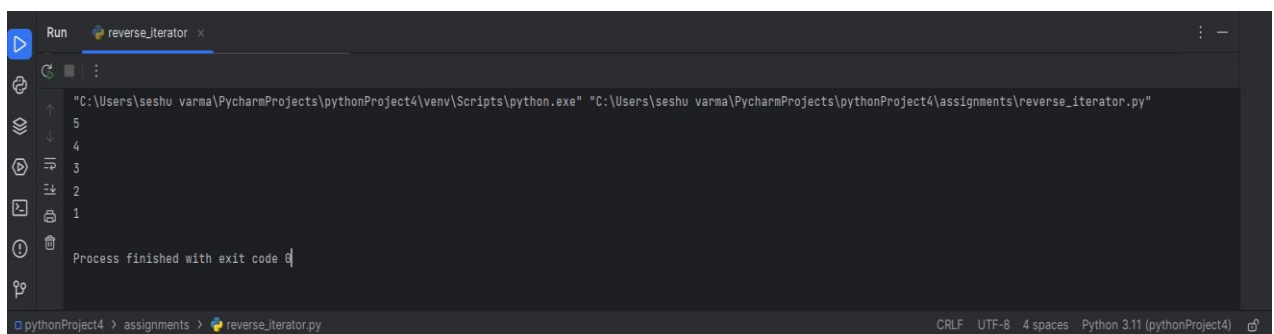
**Task 2: Create a custom iterator class ReverseIter, which takes a list and iterates it from the reverse direction.**

*Execution Code:*



```
pythonProject4  Version control
list_comprehension.py  reverse_iterator.py x
1 usage
2 class ReverseIter:
3     def __init__(self, lst):
4         self.lst = lst
5         self.index = len(lst)
6
7     def __iter__(self):
8         return self
9
10    def __next__(self):
11        if self.index == 0:
12            raise StopIteration
13        self.index -= 1
14        return self.lst[self.index]
15
16 # Example usage:
17 lst = [1, 2, 3, 4, 5]
18 reverse_iter = ReverseIter(lst)
19 for item in reverse_iter:
20     print(item)
21
```

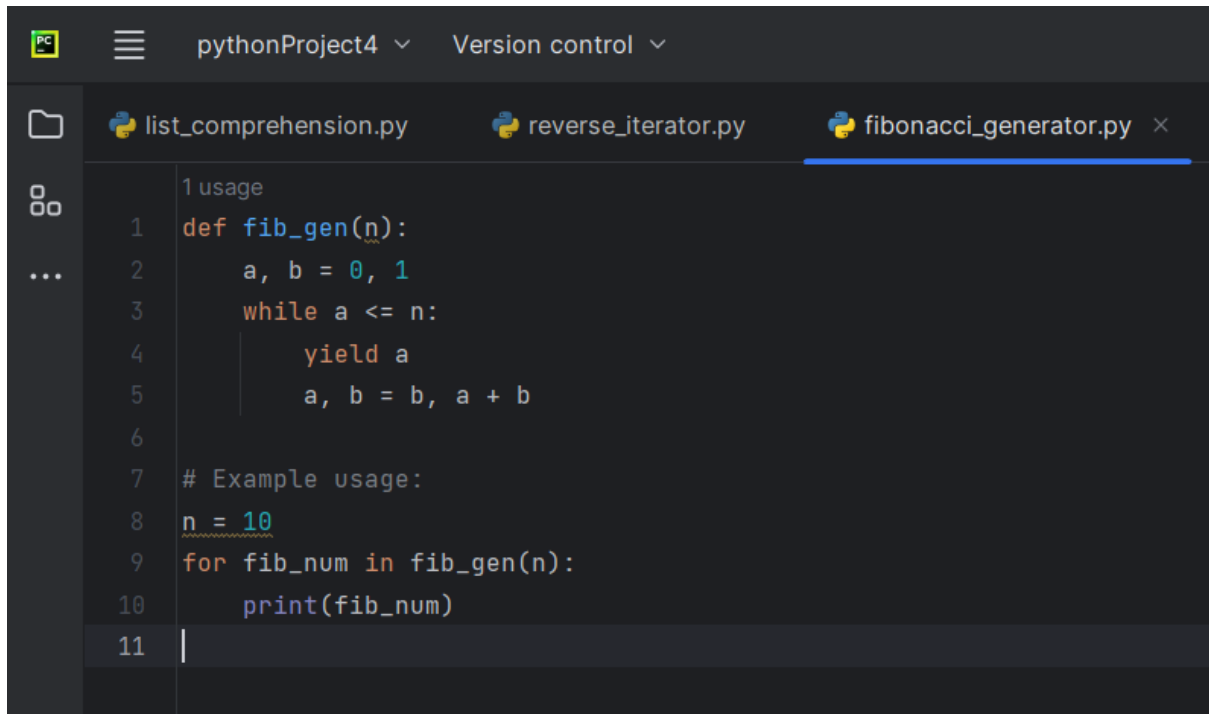
*Expected Output:*



```
Run  reverse_iterator x
"C:\Users\seshu varma\PycharmProjects\pythonProject4\venv\Scripts\python.exe" "C:\Users\seshu varma\PycharmProjects\pythonProject4\assignments\reverse_iterator.py"
5
4
3
2
1
Process finished with exit code 0
pythonProject4 > assignments > reverse_iterator.py  CRLF UTF-8 4 spaces Python 3.11 (pythonProject4)
```

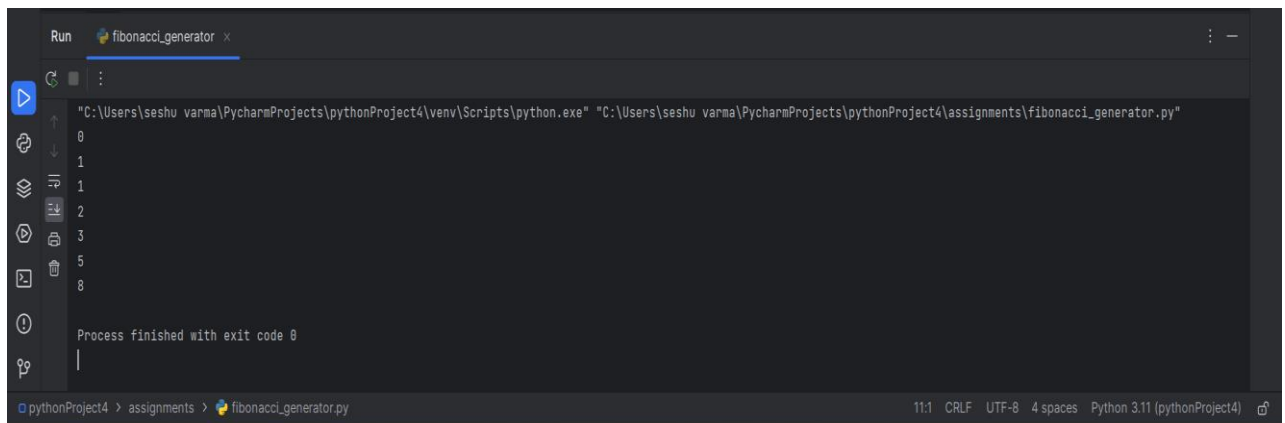
**Task 3: Write a generator function `fib_gen` that yields the Fibonacci sequence up to a given number `n`.**

*Execution Code:*



```
pythonProject4  Version control  list_comprehension.py  reverse_iterator.py  fibonacci_generator.py x
1 usage
2 def fib_gen(n):
3     a, b = 0, 1
4     while a <= n:
5         yield a
6         a, b = b, a + b
7
8 # Example usage:
9 n = 10
10 for fib_num in fib_gen(n):
11     print(fib_num)
```

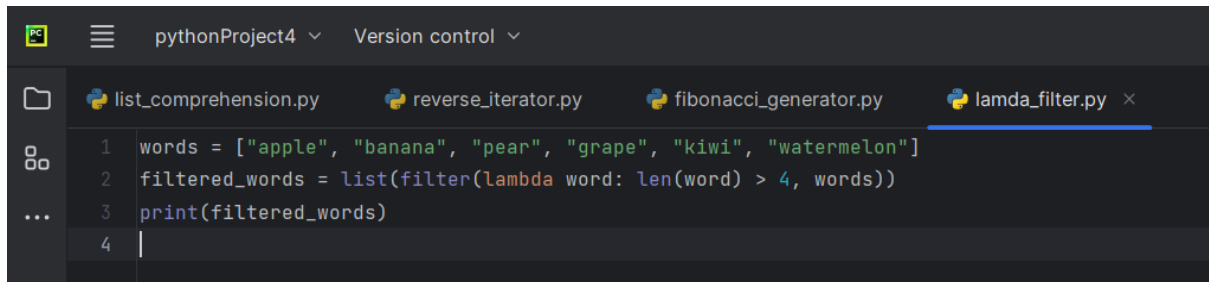
*Expected Output:*



```
Run  fibonacci_generator x
"C:\Users\seshu varma\PycharmProjects\pythonProject4\venv\Scripts\python.exe" "C:\Users\seshu varma\PycharmProjects\pythonProject4\assignments\fibonacci_generator.py"
0
1
1
2
3
5
8
Process finished with exit code 0
pythonProject4 > assignments > fibonacci_generator.py  11:1 CRLF UTF-8 4 spaces Python 3.11 (pythonProject4)
```

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

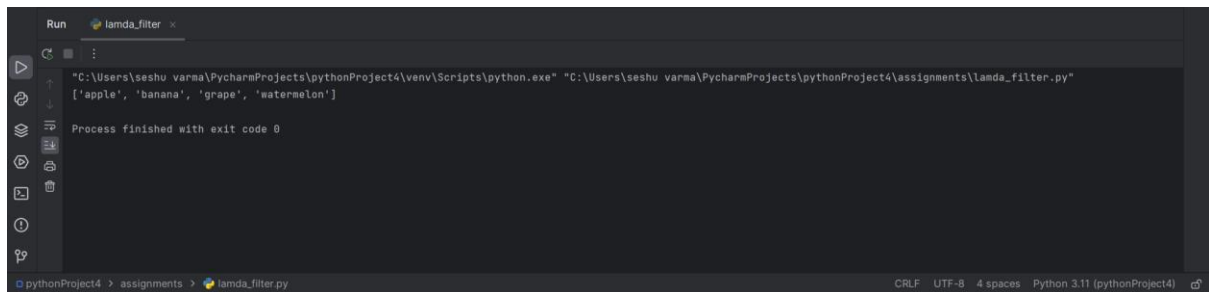
*Execution Code:*



The screenshot shows a code editor with a dark theme. At the top, there's a header bar with a Python icon, a hamburger menu, and two dropdown menus labeled 'pythonProject4' and 'Version control'. Below the header, there's a tab bar with four tabs: 'list\_comprehension.py', 'reverse\_iterator.py', 'fibonacci\_generator.py', and 'lamda\_filter.py'. The 'lamda\_filter.py' tab is active. The code in the editor is as follows:

```
1 words = ["apple", "banana", "pear", "grape", "kiwi", "watermelon"]
2 filtered_words = list(filter(lambda word: len(word) > 4, words))
3 print(filtered_words)
4
```

*Expected Output:*



The screenshot shows a terminal window with a dark theme. The title bar says 'Run' and 'lamda\_filter'. The terminal output is as follows:

```
"C:\\Users\\seshu varma\\PycharmProjects\\pythonProject4\\venv\\Scripts\\python.exe" "C:\\Users\\seshu varma\\PycharmProjects\\pythonProject4\\assignments\\lamda_filter.py"
['apple', 'banana', 'grape', 'watermelon']
Process finished with exit code 0
```

At the bottom of the terminal, there's a status bar with the following text: 'pythonProject4 > assignments > lamda\_filter.py', 'CRLF', 'UTF-8', '4 spaces', 'Python 3.11 (pythonProject4)', and a small icon.