VIDHI ROHIRA

S.Y B.TECH

SEM III

COMPUTER ENGINEERING

PD LAB 7

231071052

BATCH – C

LABORATORY 7

AIM:- To write programs in list comprehension

THEORY:-

Q] WHAT ARE LIST COMPREHENSIONS ?

List comprehensions in Python provide a concise way to create lists. They are a more readable and compact alternative to using loops and `append()` for generating lists. List comprehensions allow you to generate a new list by applying an expression to each element of an existing iterable (like a list, tuple, or string) and can include optional filtering logic.

EXAMPLES:-

1] Creating a list of squares:

```
numbers = [1, 2, 3, 4, 5]
squares = [x**2 for x in numbers]
print(squares)  # Output: [1, 4, 9, 16, 25]
```

Output

```
PS F:\VIDHI ROHIRA SY BTECH CE\SEMESTER 3\PD_LAB_7> & C:/Users/DELL/AppData/Local/Microsoft/WindowsApps/python3.11.exe "f:/VIDHI ROHIRA SY BTECH CE/SEMESTER 3/PD_LAB_
7/listcomprehensions.py"
[1, 4, 9, 16, 25]
```

2] Creating a list of even numbers:

```
numbers = range(10)
evens = [x for x in numbers if x % 2 == 0]
print(evens)  # Output: [0, 2, 4, 6, 8]
```

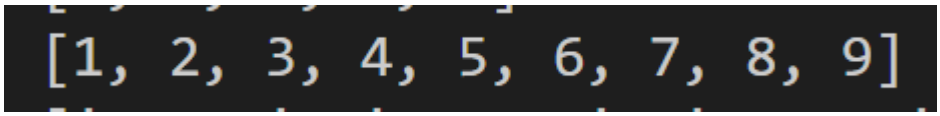
Output

```
[0, 2, 4, 6, 8]
```

3] Nested comprehensions:

```
matrix = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
flat_list = [item for sublist in matrix for item in sublist]
print(flat_list)
```

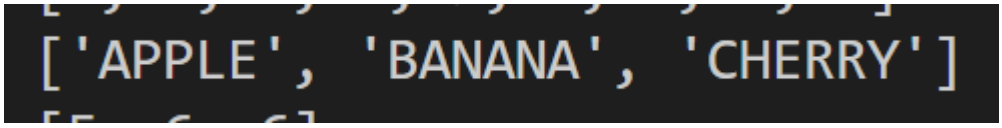
Output



4] Simple Transformations:

```
words = ['apple', 'banana', 'cherry']
upper_words = [word.upper() for word in words]
print(upper_words)
```

Output



5] Using a function in the comprehensions:

```
words = ['apple', 'banana', 'cherry']
word_lengths = [len(word) for word in words]
print(word_lengths)
```

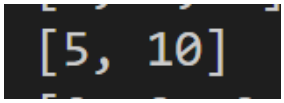
Output

```
[5, 6, 6]
```

6] Filtering elements:

```
numbers = [-10, -5, 0, 5, 10]
positives = [num for num in numbers if num > 0]
print(positives)
```

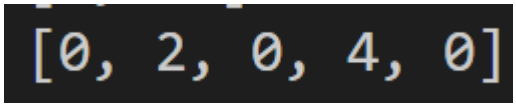
Output



7] Using if and else:

```
numbers = [-1, 2, -3, 4, -5]
replaced = [num if num > 0 else 0 for num in numbers]
print(replaced)
```

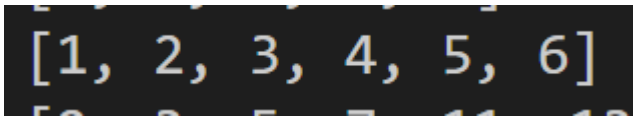
Output



8] Flatten a list:

```
lists = [[1, 2], [3, 4], [5, 6]]
flat_list = [num for sublist in lists for num in sublist]
print(flat_list)  # Output: [1, 2, 3, 4, 5, 6]
```

Output



9] Prime Numbers in a list:

```
primes = [x for x in range(2, 51) if all(x % i != 0 for i in range(2,
int(x**0.5) + 1))]
print(primes)  # Output: [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43,
47]
```

Output

```
[2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47]
```

10] Nested List 2D matrix creation:

```
matrix = [[i + j for j in range(3)] for i in range(3)]
print(matrix)
```

Output

```
 [[0, 1, 2], [1, 2, 3], [2, 3, 4]]
 PS F:\VIDHI ROHIRA SY BTECH CE\SEMESTER 3\PD_LAB_7>
```

CONCLUSION:- Hence in this lab we have studied about list comprehensions and implemented some of its examples.