```
#Simple List Comprehension

list1=[10,20,30,40]

square_list=[]

for i in list1:

  square_list.append(i**2)

print(f"the sqaures of the list items is {square_list} ")
```

```
the sqaures of the list items is [100, 400, 900, 1600]
```

```
#comprehensions basically refers to the manipulations tht can be done using the list or dictionary

squares = [x**2 for x in range(10)]
print(squares)
```

```
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
```

```
#List Comprehension with an If Clause


evens = [x for x in range(10) if x % 2 == 0]
print(evens)
```

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

```
#Nested List Comprehension

# Create a 3x3 matrix using nested list comprehensions
matrix = [[j for j in range(3)] for i in range(3)]
print(matrix)
```

```
[[0, 1, 2], [0, 1, 2], [0, 1, 2]]
```

```
#Combining Multiple If Clauses

even_squares = [x**2 for x in range(10) if x % 2 == 0 and x != 0]
print(even_squares)
```

```
[4, 16, 36, 64]
```

```
#Using Functions in List Comprehensions

# Define a function to compute the cube of a number
def cube(x):
    return x**3

# Create a list of cubes of numbers from 0 to 4
cubes = [cube(x) for x in range(5)]
print(cubes)
```

```
[0, 1, 8, 27, 64]
```

```
#DICTIONARY COMPHREHENSIONS

# Basic Example: Dictionary of squares
squares = {x: x**2 for x in range(10)} #the key value pairs make the diffrence between the lists and dictionary

# Conditional Logic: Dictionary of even squares
even_squares = {x: x**2 for x in range(10) if x % 2 == 0}

# Transforming Data: Dictionary with doubled values
original_dict = {'a': 1, 'b': 2, 'c': 3}
transformed_dict = {k: v * 2 for k, v in original_dict.items()}

# Using Functions: Dictionary of squares using a function
def square(x):
    return x * x
numbers = [1, 2, 3, 4]
squared_dict = {n: square(n) for n in numbers}

# Nested Comprehensions: Matrix represented as a dictionary
matrix = {(i, j): i * j for i in range(3) for j in range(3)}

# Print all the dictionaries
print("Squares:", squares)
print("Even Squares:", even_squares)
print("Transformed Dict:", transformed_dict)
print("Squared Dict:", squared_dict)
print("Matrix:", matrix)
```

```
Squares: {0: 0, 1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81}
Even Squares: {0: 0, 2: 4, 4: 16, 6: 36, 8: 64}
Transformed Dict: {'a': 2, 'b': 4, 'c': 6}
Squared Dict: {1: 1, 2: 4, 3: 9, 4: 16}
```

```
Matrix: {(0, 0): 0, (0, 1): 0, (0, 2): 0, (1, 0): 0, (1, 1): 1, (1, 2): 2, (2, 0): 0, (2, 1): 2, (2, 2): 4}
```