Computing Fundamentals using Python

SUBJECT CODE : UQ25CA151A

Samyukta D Kumta

Computer Applications

Comprehensions in Python

- comprehension is a concise way to create new sequences (like lists, sets, or dictionaries) from existing iterables (like lists, strings, or ranges).

- It allows you to write cleaner, shorter, and more readable code compared to traditional for loops.

Comprehensions in Python

There are mainly four types of comprehensions:

1. List Comprehension
2. Set Comprehension
3. Dictionary Comprehension
4. Generator Expression

Comprehensions in Python

List Comprehension

Creates a new list from an existing iterable.

Syntax:

[expression for item in iterable if condition]

Comprehensions in Python

List Comprehension

Example:

```
# Squares of numbers 0–9
squares = [x**2 for x in range(10)]
print(squares)       #Output  [0, 1, 4, 9, 16, 25, 36, 49, 64, 81]

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

Comprehensions in Python

Set Comprehension

Similar to list comprehension, but creates a set (removes duplicates automatically).

Example:

```
nums = [1, 2, 2, 3, 4, 4, 5]
unique_squares = {x**2 for x in nums}
print(unique_squares)

# output {1, 4, 9, 16, 25}
```

Comprehensions in Python

Dictionary Comprehension

Creates a dictionary from an iterable.

Example:

```
# Map numbers to their squares
squares_dict = {x: x**2 for x in range(5)}
print(squares_dict)

# Output {0: 0, 1: 1, 2: 4, 3: 9, 4: 16}
```

Comprehensions in Python

Generator Expression

Looks like a list comprehension but uses () instead of []. It creates a generator (lazy evaluation).

Example:

```
gen = (x**2 for x in range(5))
print(next(gen))  # 0
print(next(gen))  # 1
print(list(gen))  # [4, 9, 16]
```

Advantages of Comprehension:

- More compact and readable.
- Often faster than using loops.
- Useful for filtering and transforming data.

PES UNIVERSITY
CELEBRATING 50 YEARS

Multiple Choice Questions

```
nums = [1, 2, 3, 4, 5]
result = [x*2 for x in nums if x%2==0]
print(result)
```

a) [2, 4, 6, 8, 10]
b) [4, 8]
c) [2, 6, 10]
d) [1, 3, 5]

Answer: b) [4, 8]

PES UNIVERSITY
CELEBRATING 50 YEARS

Multiple Choice Questions

Which of the following is not a valid comprehension in Python?
a) List comprehension
b) Set comprehension
c) Dictionary comprehension
d) Tuple comprehension

Answer: d) Tuple comprehension

Multiple Choice Questions

What will the following set comprehension produce?

```
s = {x % 3 for x in range(6)}
print(s)
```

a) {0, 1, 2, 3, 4, 5}
b) {0, 1, 2}
c) {0, 3}
d) {1, 2, 3}

Answer: b) {0, 1, 2}

PES UNIVERSITY
CELEBRATING 50 YEARS

Multiple Choice Questions

What is the output of the dictionary comprehension below?

```
d = {x: x**2 for x in range(3)}
print(d)
```

a) {0: 1, 1: 2, 2: 3}
b) {0: 0, 1: 1, 2: 4}
c) {1: 1, 2: 2, 3: 3}
d) {0: 0, 1: 2, 2: 3}

Answer: b) {0: 0, 1: 1, 2: 4}

PES UNIVERSITY
CELEBRATING 50 YEARS

Multiple Choice Questions

Which of the following creates a generator?

a) [x for x in range(5)]
b) (x for x in range(5))
c) {x for x in range(5)}
d) {x: x for x in range(5)}

Answer: b) (x for x in range(5))

Practice Questions

1. A teacher enters the marks of students in one line (space-separated). Write a one-liner comprehension to store only the marks **greater than or equal to 40** (pass marks).

2. You are given a single line of student names. Create a dictionary where the key is the student name and the value is the number of characters in the name.

3. Given a line of words, extract **only the vowels** from each word and return them as a list of lists.

THANK YOU

Samyukta D Kumta
Department of Computer Applications
samyuktad@pes.edu