PYTHON

INTERVIEW QUESTIONS

with Answers to Crack

TECHNICAL INTERVIEW

Topic 3

String handling functions:

Q.1 How do you concatenate two strings in Python?

Ans: You can concatenate two strings using the **+** operator. For example:

```python
str1 = "Hello"
str2 = "World"
result = str1 + str2   # Output: "HelloWorld"
```

Q.2 How do you find the length of a string in Python?

Ans: You can use the **len()** function to find the length of a string. For example:

```python
str1 = "Hello World"
length = len(str1)   # Output: 11
```

Q.3 How do you convert a string to uppercase in Python?

Ans: You can use the **upper()** method to convert a string to uppercase. For example:

```python
str1 = "hello"
uppercase_str = str1.upper()   # Output: "HELLO"
```

Q.4 How do you split a string into a list of substrings in Python?

Ans: You can use the split() method to split a string into a list of substrings based on a delimiter. For example:

```python
str1 = "Hello,World"
substrings = str1.split(",")
# Output: ["Hello", "World"]
```

Q.5 How do you check if a string contains a specific substring in Python?

Ans: You can use the **in** keyword to check if a substring is present in a string. For example:

```python
str1 = "Hello World"
is_present = "World" in str1   # Output: True
```

Control statements, functions in Python:

Q.1 What are control statements in Python?

Ans: Control statements are used to control the flow of execution in a program. Common control statements in Python include if-else, for loops, while loops, and break/continue statements.

Q.2 How do you write an if-else statement in Python?

Ans: An if-else statement in Python is written using the following syntax:

```python
if condition:
    # Code block executed if the condition is
True
else:
    # Code block executed if the condition is
False
```

Q.3 How do you define a function in Python?

Ans: A function in Python is defined using the **def** keyword. For example:

```python
def greet():
    print("Hello, world!")
```

Q.4 How do you pass arguments to a function in Python?

Ans: You can pass arguments to a function by including them inside the parentheses when defining the function. For example:

```python
def greet(name):
    print("Hello, " + name + "!")
```

Q.5 How do you return a value from a function in Python?

Ans: You can use the return keyword to return a value from a function. For example:

```python
def add(a, b):
    return a + b
```

Topic 5

Special data types in Python:

Q.1 What is a set in Python?

Ans: A set in Python is an unordered collection of unique elements. It is defined using curly braces **{}** or the **set()** constructor. For example:

```python
my_set = {1, 2, 3}   # Output: {1, 2, 3}
```

Q.2 What is a dictionary in Python?

Ans: A dictionary in Python is an unordered collection of key-value pairs. It is defined using curly braces **{}** or the **dict()** constructor. For example:

```python
my_dict = {"name": "John", "age": 25}
# Output: {"name": "John", "age": 25}
```

Q.3 How do you access values in a dictionary in Python?

Ans: You can access values in a dictionary by using the corresponding key. For example:

```python
my_dict = {"name": "John", "age": 25}
print(my_dict["name"])   # Output: "John"
```

Q.4 What is a tuple in Python?

Ans: A tuple in Python is an ordered and immutable collection of elements. It is defined using parentheses () or the **tuple()** constructor. For example:

```python
my_tuple = (1, 2, 3)   # Output: (1, 2, 3)
```

Q.5 How do you swap the values of two variables in Python?

Ans: You can swap the values of two variables using a temporary variable or simultaneous assignment. For example:

```python
a = 5
b = 10
a, b = b, a
print(a, b)   # Output: 10, 5
```

Lambda functions, list comprehension:

Q.1 What is a lambda function in Python?

Ans: A lambda function is an anonymous function defined using the **lambda** keyword. It is typically used for short, one-line functions. For example:

```python
square = lambda x: x**2
print(square(3))   # Output: 9
```

Q.2 What is list comprehension in Python?

Ans: List comprehension is a concise way to create lists in Python based on existing lists or other iterables. It combines the creation of a new list with a loop and optional conditional statements. For example:

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

Q.3 How do you filter elements in a list using list comprehension?

Ans: You can filter elements in a list using list comprehension by adding a conditional statement. For example, to filter even numbers:

```python
numbers = [1, 2, 3, 4, 5]
even_numbers = [x for x in numbers if x % 2 == 0]
print(even_numbers)   # Output: [2, 4]
```

Q.4 Can you have multiple if conditions in list comprehension?

Ans: Yes, you can have multiple if conditions in list comprehension by chaining them using the and or or operators. For example:

```python
numbers = [1, 2, 3, 4, 5]
filtered_numbers = [x for x in numbers if x % 2 == 0 and x > 2]
print(filtered_numbers)   # Output: [4]
```

Q.5 How do you create a dictionary using list comprehension in Python?

Ans: You can create a dictionary using list comprehension by specifying key-value pairs within curly braces {}. For example:

```python
keys = ['a', 'b', 'c']
values = [1, 2, 3]
my_dict = {k: v for k, v in zip(keys, values)}
print(my_dict)  # Output: {'a': 1, 'b': 2, 'c': 3}
```