30 Python-related questions that are commonly asked for roles like Data Scientist, Python Developer, and Machine Learning Engineer, particularly focusing on Python control flow (like loops, if-else statements, etc.) at companies such as TCS:

1. What is the difference between == and is in Python?

- == checks if two objects have the same value, whereas is checks if two references point to the same object in memory.

2. How would you handle errors or exceptions in Python?

- Use try-except blocks to catch and handle exceptions.

3. What are Python's different loop structures?

- Python has for and while loops for iteration.

4. Explain the difference between for and while loops.

- for is used for iterating over a sequence (like a list, tuple, or range), while while runs until a condition is false.

5. How does the break statement work in Python?

- break is used to exit the loop prematurely.

6. What is the use of the continue statement in Python?

- continue skips the current iteration of the loop and proceeds with the next iteration.

7. What are the different types of conditional statements in Python?

- if, elif, and else are used for decision-making in Python.

8. What is a nested loop in Python?

- A nested loop is when one loop is placed inside another loop.

9. Explain the difference between if and elif in a conditional statement.

- if is used for the first condition, and elif allows for checking additional conditions if the if condition is false.

10. How do you write a Python program to check if a number is prime?

- Using a for loop to check if the number is divisible by any integer other than 1 and itself.

11. How do you handle multiple conditions in a Python if statement?

- You can use logical operators (and, or, not) to combine conditions.

12. What is the output of the following code?

```
x = 5

if x > 3:

  print("x is greater than 3")

elif x > 2:

  print("x is greater than 2")

else:

  print("x is 2 or less")
```

- **Answer:** x is greater than 3

13. What is a list comprehension?

- A concise way to create lists in Python using a single line of code, often with a for loop inside it.

14. How do you implement an infinite loop in Python?

- Using a while True: loop.

15. What will happen if you run this code?

```
for i in range(5):

  print(i)

  if i == 2:

    break
```

- **Answer:** It will print 0, 1, 2 and then stop because of the break statement.

16. Explain how the else block works in a loop.

- The else block in a loop executes after the loop finishes, unless the loop is exited using a break statement.

17. What is a try-except block?

- It is used to handle exceptions. Code inside try is executed, and if an exception occurs, control is passed to except.

18. What is the purpose of the pass statement in Python?

- It is a placeholder used when a statement is syntactically required but you don't want to execute any code.

19. What will the following code output?

```
i = 1
while i < 5:
    print(i)
    i += 1
```

- **Answer:** 1 2 3 4

20. How do you write a Python program to reverse a string using a loop?

- You can use a for loop to iterate backward through the string.

21. What is the output of the following code?

```
x = 4
if x < 5:
    print("x is less than 5")
else:
    print("x is greater than or equal to 5")
```

- **Answer:** x is less than 5

22. What is the use of range() function in Python?

- It generates a sequence of numbers, which is commonly used in loops.

23. Can we use else with a for loop?

- Yes, the else block executes if the loop completes normally (i.e., not through a break statement).

24. How do you handle multiple exceptions in one except block?

- You can specify multiple exceptions by separating them with a comma.

```
try:
    # Code that may raise exceptions
except (TypeError, ValueError) as e:
    # Handling multiple exceptions
```

25. What is the output of the following code?

```
x = 10
if x < 5:
```

```
    print("x is less than 5")
elif x == 10:
    print("x is 10")
else:
    print("x is greater than 5")
```

- **Answer:** x is 10

26. What will be the output of the following code?

```
i = 0
while i < 5:
    print(i)
    i += 1
else:
    print("Done")
```

- **Answer:** 0 1 2 3 4 Done

27. What is the result of this code?

```
for i in range(1, 5):
    if i == 3:
        continue
    print(i)
```

- **Answer:** 1 2 4

28. Can you write a Python program that uses if-else to check whether a number is even or odd?

```
num = int(input("Enter a number: "))
if num % 2 == 0:
    print("Even")
else:
    print("Odd")
```

29. How would you write a Python function that sums all even numbers in a list using a loop?

```
def sum_even_numbers(lst):
    total = 0
    for num in lst:
        if num % 2 == 0:
            total += num
    return total
```

30. What is a generator in Python, and how is it different from a normal function?

- A generator is a function that yields items one at a time using yield instead of returning them all at once. This allows for lazy evaluation, meaning values are produced on demand.

These questions test your understanding of Python's control flow, loops, conditional statements, and exception handling, which are crucial for roles like Data Scientist, Python Developer, and Machine Learning Engineer.