Unclassified

Flow Control

Conditional Statements:

- Conditional statements are used to perform different actions based on certain conditions.
- The most common conditional statement in Python is the if statement.
- It allows you to execute a block of code if a specific condition is true.
- Optionally, you can use elif (short for "else if") to specify additional conditions to check, and else to provide a default action if none of the conditions are met.

```
age = 18

if age < 18:
    print("You are underage.")
elif age >= 18 and age < 65:
    print("You are an adult.")
else:
    print("You are a senior citizen.")
```

Looping:

- Loops are used to repeat a block of code multiple times.
- Python provides two types of loops: for loop and while loop.
- The for loop iterates over a sequence of elements (such as a list or a string) or a range of numbers.

```
fruits = ["apple", "banana", "cherry"]

for fruit in fruits:
    print(fruit)
```

- The while loop repeats a block of code as long as a certain condition is true.

```
count = 0

while count < 5:
    print("Count:", count)
  count += 1
```

Break and Continue:

- The break statement is used to exit a loop prematurely.

- It is commonly used within loops to terminate the loop execution based on a specific condition.
- The continue statement is used to skip the remaining code in a loop iteration and move to the next iteration.
- It is useful when you want to skip certain iterations based on specific conditions.

```
numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

for num in numbers:
    if num == 5:
        continue  # Skip number 5 and proceed to the next iteration
    elif num == 8:
        break  # Terminate the loop when reaching number 8
    else:
        print(num)
```