DAY 1

DAY 2 ,date - MARCH/11/2025

Loops and Control flow tools

✅ Loops

Loops are used to repeatedly execute a block of code as long as a condition is met. Python supports two types of loops:

1. **for loop – Used to iterate over a sequence (like a list, tuple, string, etc.).**
   Example:

python
CopyEdit
```
for i in range(5):
    print(i)
```

2. **while loop – Repeats a block of code while a condition is `True`.**
   Example:

python

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

✅ Loop Control Statements

- **break – Stops the loop immediately.**
- **continue – Skips the current iteration and moves to the next one.**
- **else – Executes after the loop finishes normally (without `break`).**

Example with `break`, `continue`, and `else`:

python
CopyEdit
```
for i in range(5):
    if i == 3:
        continue
    print(i)
else:
    print("Loop finished!")
```

Output:

```
0
1
2
4
Loop finished!
```

Functions and Modules

🌟 Types of Arguments in Python – A Simple Story 🌟

Imagine you're at a pizza shop ordering a pizza. The process of ordering reflects how arguments work in Python. Let's see how each type of argument fits into this scenario:

🍕 1. Positional Arguments – "First come, first served"

When you order a pizza, the order in which you say the ingredients matters. If you say:
👉 "Cheese, Pepperoni" – you'll get a pizza with cheese and pepperoni.

But if you say:
👉 **"Pepperoni, Cheese" – you'll get a pizza with pepperoni first, possibly changing the outcome!**

Python Example:

python
CopyEdit
```
def order_pizza(topping1, topping2):
    print(f"Pizza with {topping1} and
{topping2}")

order_pizza("Cheese", "Pepperoni")   #
Order matters!
order_pizza("Pepperoni", "Cheese")   #
Outcome changes!
```

Output:

csharp
CopyEdit
```
Pizza with Cheese and Pepperoni
Pizza with Pepperoni and Cheese
```

🍕 **2. Keyword Arguments – "Specify what you want!"**

Instead of listing the toppings in order, you can tell the chef exactly what you want, no matter the order:
👉 **"I want cheese first and pepperoni second."**
👉 **"I want pepperoni first and cheese second."**

Python Example:

python
CopyEdit

```python
def order_pizza(topping1, topping2):
    print(f"Pizza with {topping1} and 
{topping2}")

order_pizza(topping1="Cheese", 
topping2="Pepperoni")
order_pizza(topping2="Cheese", 
topping1="Pepperoni")  # Order doesn’t 
matter
```

Output:

csharp
CopyEdit

```
Pizza with Cheese and Pepperoni
Pizza with Pepperoni and Cheese
```

🍕 3. Default Arguments – "The house special"

If you don't mention any toppings, the chef will give you the house special with cheese:
👉 "I'll have a pizza." – You get a plain cheese pizza!
👉 "I'll have a pizza with pepperoni." – Now you get a pepperoni pizza!

Python Example:

python
CopyEdit

```
def order_pizza(topping="Cheese"):
    print(f"Pizza with {topping}")

order_pizza()                    # Default
value used
order_pizza("Pepperoni")         # Custom
value used
```

Output:

csharp
CopyEdit

```
Pizza with Cheese
Pizza with Pepperoni
```

🍕 4. Variable-Length Arguments (`*args`) – "Add as many toppings as you want!"

The chef says you can add as many toppings as you like:
👉 "I want cheese, pepperoni, mushrooms, and olives!"
👉 "I'll just take cheese."

Python Example:

python
CopyEdit

```
def order_pizza(*toppings):
    print(f"Pizza with {',
'.join(toppings)}")

order_pizza("Cheese", "Pepperoni",
"Mushrooms", "Olives")
order_pizza("Cheese")
```

Output:

csharp
CopyEdit

```
Pizza with Cheese, Pepperoni, Mushrooms,
Olives
Pizza with Cheese
```

🍕 5. Keyword Variable-Length Arguments (`**kwargs`) – "Custom Order Instructions"

The chef lets you specify additional custom instructions:
👉 "I want a pizza with cheese and pepperoni, but make it extra crispy and cut into squares."

Python Example:

python
CopyEdit

```python
def order_pizza(**instructions):
    for key, value in instructions.items():
        print(f"{key}: {value}")

order_pizza(topping1="Cheese",
topping2="Pepperoni", style="Extra crispy",
cut="Squares")
```

Output:

makefile

CopyEdit

```
topping1: Cheese
topping2: Pepperoni
style: Extra crispy
cut: Squares
```

🌟 Types of Arguments in Python – A Simple Story 🌟

Imagine you're at a pizza shop ordering a pizza. The process of ordering reflects how arguments work in Python. Let’s see how each type of argument fits into this scenario:

🍕 1. Positional Arguments – "First come, first served"

When you order a pizza, the order in which you say the ingredients matters. If you say:
👉 "Cheese, Pepperoni" – you'll get a pizza with cheese and pepperoni.
But if you say:
👉 "Pepperoni, Cheese" – you'll get a pizza with pepperoni first, possibly changing the outcome!

Python Example:

python
CopyEdit

```
def order_pizza(topping1, topping2):
    print(f"Pizza with {topping1} and
{topping2}")

order_pizza("Cheese", "Pepperoni")   #
Order matters!
order_pizza("Pepperoni", "Cheese")   #
Outcome changes!
```

Output:

csharp
CopyEdit

```
Pizza with Cheese and Pepperoni
Pizza with Pepperoni and Cheese
```

🍕 2. Keyword Arguments – "Specify what you want!"

Instead of listing the toppings in order, you can tell the chef exactly what you want, no matter the order:
👉 "I want cheese first and pepperoni second."
👉 "I want pepperoni first and cheese second."

Python Example:

python

CopyEdit
```
def order_pizza(topping1, topping2):
    print(f"Pizza with {topping1} and
{topping2}")

order_pizza(topping1="Cheese",
topping2="Pepperoni")
order_pizza(topping2="Cheese",
topping1="Pepperoni")  # Order doesn't
matter
```

Output:

csharp
CopyEdit
```
Pizza with Cheese and Pepperoni
Pizza with Pepperoni and Cheese
```

🍕 3. Default Arguments – "The house special"

If you don't mention any toppings, the chef will give you the house special with cheese:
👉 "I'll have a pizza." – You get a plain cheese pizza!
👉 "I'll have a pizza with pepperoni." – Now you get a pepperoni pizza!

Python Example:

python
CopyEdit
```
def order_pizza(topping="Cheese"):
    print(f"Pizza with {topping}")

order_pizza()                   # Default
value used
order_pizza("Pepperoni")        # Custom
value used
```

Output:

csharp
CopyEdit
```
Pizza with Cheese
Pizza with Pepperoni
```

🍕 4. Variable-Length Arguments (`*args`) – "Add as many toppings as you want!"

The chef says you can add as many toppings as you like:
👉 **"I want cheese, pepperoni, mushrooms, and**

olives!"
👉 **"I'll just take cheese."**

Python Example:

python
CopyEdit
```
def order_pizza(*toppings):
    print(f"Pizza with {',
'.join(toppings)}")

order_pizza("Cheese", "Pepperoni",
"Mushrooms", "Olives")
order_pizza("Cheese")
```

Output:

csharp
CopyEdit
```
Pizza with Cheese, Pepperoni, Mushrooms,
Olives
Pizza with Cheese
```

🍕 **5. Keyword Variable-Length Arguments (`**kwargs`) – "Custom Order Instructions"**

The chef lets you specify additional custom instructions:
👉 **"I want a pizza with cheese and pepperoni, but make it extra crispy and cut into squares."**

Python Example:

python
CopyEdit
```
def order_pizza(**instructions):
    for key, value in instructions.items():
        print(f"{key}: {value}")

order_pizza(topping1="Cheese",
topping2="Pepperoni", style="Extra crispy",
cut="Squares")
```

Output:

makefile
CopyEdit
```
topping1: Cheese
topping2: Pepperoni
style: Extra crispy
cut: Squares
```