Day 2 @January 3, 2024

Topics:

- Conditions
- Comparison operators
- If, else, elif (conditional statements)
- login screen project
- While loops
- Counting with while loops

Conditions:

- Decisions
- Either true or false
- If something happens do something otherwise do something else
- Condition: if the temperature is cold in the morning
- Possibilities: Cold or Warm
- Result: Cold → Oats ; Warm → Eggs

Comparison operators:

- They allow us to compare value
- They will always give you a boolean value (True or False)
- number1= 10 and number2= 5
- number1 > number2 (greater than) → True
- number2< number1 (less than) → True
- number1 == number2 (equality) → False
- number1 = 10 and number2 = 10

- number1 ≥ number2 (greater than equal to) → True
- number2 ≤ number1 (less than or equal to) → True

```
day = "Wednesday"
day == "Wednesday"
```

Combining the two things:

- Condition is made up of comparison statements
- Conditions:
  - day == “monday”
  - number > 2
  - time < 1800
  - temperature > 20

AND, OR, NOT logic operators:

A	B	A and B	A or B	Not A
False	False	False	False	True
False	True	False	True	True
True	False	False	True	False
True	True	True	True	False

- These are used to join two conditions together
- AND: that both left and right hand conditions must be true for the whole condition to be true
- OR: Either of the two conditions to be true or both to be true for the whole condition to be true
- Not statement: invert the truth value of the condition (negation)
  - True → Not → False

- False → Not → True

IF , Else, Elif:

- Determines what to do based on truth value of a condition
- truth value: true or false

if statement:

- To run a specific code block based on a condition
- Code block: simply a group of programming statements
- indentation: space and it comes after your if statement

```
if (condition):
    #code here (code block)

if(10 > 5):
    #do something
```

else statements:

- They dont have a condition
- The run the code in the code block if the condition is false
- else require and if statement

```
if (5 > 10):
    #do something
else:
    #do something else
```

elif statements:

- Check condition after an if statement
- They require an if statement to be present

```
amount = int(input("please enter your money"))
if (amount == 0):
    print("sorry you cant buy anything")
elif (amount == 0.5):
    print("here is a candy")
elif (amount == 1):
    print("here is a bag of crisps")

else:
    print("You have a lot of money")
```

Project 2:

- Build a login screen screen for your computer
- use input to take user input
- You will take user input for password and username
- If they are correct you will log the user in
  - print(”you are logged in”)
  - f strings to say which user is logged in: Ahmed is logged in
- if statements
- AND operators
- You will check for equality

```
username = input("please give me your username")
password = input("please give me the password")

if username == "username" and password == "password123":
    print(f"You have logged in as user {username}")
else:
    print("You did not login")
```

Loops:

- Repeat the code until a certain condition is true

While loops:

- will run until a condition is true
- It checks the statement before every iteration of the loop (run)
- if the condition remains true then the while loop does not terminate (end)
- break can be used to terminate the while loop

```
while (condition):
    #Your code here
```

.

Counting with loops:

- counter: Keeps track of the number of a certain thing (Steps, calories, number of cars....)
- You need to update the counter
- counter = counter +1
- counter = 0 + 1 (1)
- counter = 1+1 (2)
- counter = 2+1(3)

counting in Python:

```
#count from 1 till 10
count = 1
while count < 11:
    print(count)
    count = count + 1

#count from 10 to 1
```

```
count = 10
while count > 0:
    print(count)
    count = count - 1
```