PYTHON NOTES

* first kuch bhi print karane ke liye use print ("")

{c ki Tarah h bas printf ki jagah print aur koi semicolon ki zaroorat nahi hoti}

* name = "Aashi"
* age = 18
* print(name)
* print("age")
* yaha, we declared 2 variables one which stores a string that’s why it's in double quotes, and one stores an integer. {print mei “” inke beech kuch bhi likho ge to vo print hoga.: the output of this program is:



name = input ("What is your name?")

print ("oh hello! " + name)

* here we used the input function to take input from the user.

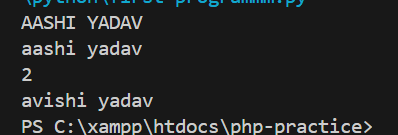
this input function generally stores data in string form.

* yob= input ("enter your year of birth:")
* age = 2024 - int(yob)
* print(age)

since the input function stores data in the string to seedha hum string ko number ke saath concatenate nahi kar sakte uske liye we use int function... 🡪 int(yob)

* some functions we can use in a string:

1. upper () – converts all the characters in uppercase
2. lower () – converts all the characters in lowercase
3. find () – helps to find a character in the string
4. replace () – Replace a particular character with another character.
5. name="aashi yadav"
6. print(name.upper())
7. print(name.lower())
8. print(name.find('s'))
9. print(name.replace("aashi","avishi"))

 the output of the following function.

name = “aashi yadav”

print ('y' in name)

print ('was' in name)

* “in” is used as a Boolean function for string

 the output of the function 🡨

* In arithmetic operators when we divide a number, we use the ‘/’ symbol this may give us answers in float form but what if we only want answers in integer format we use ‘//’ in place of ‘/’.
* In arithmetic operators when we need to find the remainder we use ‘%’ and for power we use ‘\*\*’ (e.g.: 3\*\*2 = 9)

print (5+2)

print (5-2)

print (5\*2)

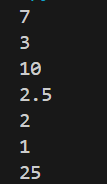
print (5/2)

print (5//2)

print (5%2)

print (5\*\*2)

output 🡪



* Basic c wali cheez –

I = 5

I = I + 2 also equal to I += 2 (similarly for subtraction and multiplication)

(point to be noted: jaise c mei hum i++ likhte the yaha nahi likh sakte invalid format h vo)

* Conditions:

1. Or: even if one of the statements is true it will be true.
2. And: when all the statements are true then only it will return true else false.
3. Not: when written in front of the statement then returns false for true statement and true for false statement.
4. print (3>2 or 5>7)
5. print (3>2 and 5>7)
6. print (not 3>4)

output 🡪



* IF ELSE STATEMENTS:

So basically, we used to use {} these brackets to write the code for if and else conditions here we don’t use brackets instead we use intendation means we give 4 space bar spaces before writing statements for respective conditions.

We also use ‘:’ for if and else. “elif” is used as an elseif condition.

Code 🡪

age = input ("enter your age:")

if int(age)>=18:

    print ("you are an adult")

    print ("you can vote")

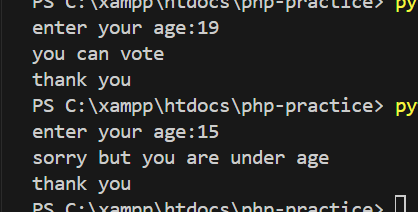
else:

    print ("Sorry but you are underage")

print ("thank you")

since print (“thank you”) is not written with 4 space bar spaces therefore it will print for every condition.

Output of the following code 🡪



EXERCISE: MINI CALCULATOR

Code:

first = input ("Enter the first number: ")

operator = input ("enter operator to be used (+, -, \*, /, %, \*\*, //): ")

second = input ("Enter the second number: ")

if operator == "+":

    print(int(first) + int(second))

elif operator == "-":

    print(int(first) - int(second))

elif operator == "\*":

    print(int(first) \* int(second))

elif operator == "/":

    print(int(first) / int(second))

elif operator == "%":

    print(int(first) % int(second))

elif operator == "\*\*":

    print(int(first) \*\* int(second))

elif operator == "//":

    print(int(first) // int(second))

else:

    print("Invalid operation")

* LOOPS

1. While Loop

Code:

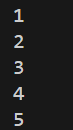
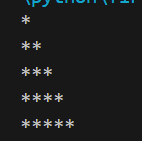
i = 1

while i<=5:

    print(i) ## star printing ke liye just write print (i \* "\*")

    i += 1

output:

   
(Similarly, we will have the syntax for For loop)

* Lists:

Basically, It's Like an array that does not depend on the type of character (can be int, str, float)

marks = [95,97,98] ##marks is a list

print(marks)

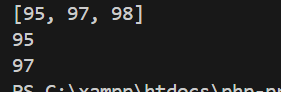
print (marks [0])

print (marks [-2])

here when you notice you see that marks [0] is written 🡪 it will display the marks at the 0th position similarly for marks [-2] 🡪 it will start counting from the last till the 2nd position.

If we only need 2 outputs then we will use print (marks [1:2]) to display 1st and 2nd numbers.

Output🡪



Operations we can attempt on the list:

Append: used to insert a value at the end of the list

Insert: used to insert the value at any position you want

Length: used to determine the length of the list

Clear: used to clear a list and returns an empty list

e.g.:

marks = [95,97,98] ##marks is a list

print(marks)

marks.insert(1, 99)

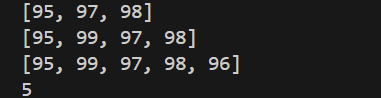
print(marks)

marks.append(96)

print(marks)

print(len(marks))

output 🡪



Using while loop:

marks = [95,97,98] ##marks is a list

i=0

while i< len(marks):

    print(marks[i])

    i +=1

output:



* FOR LOOP:

Conditions used in for loop:

* + 1. Break: jis par break kiya vo aur uske baad wala kuch bhi print nahi hoga
    2. Continue: jis par continue lagaya uske alawa sab print hoga use chodd kar

e.g.:

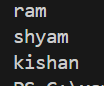
students = ["Ram","Shyam","Kishan","radha","Radhika"]

for student in students:

    if student == "Radha":

       break

    print(student)

output🡪 

code 2:

students = ["Ram","Shyam","Kishan","radha","Radhika"]

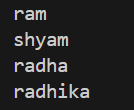
for student in students:

    if student == "Kishan":

       continue

    print(student)

output🡪



In an array kind of function, different brackets have different meanings like

[] means a list

() means tupple (its normal nothing new)

{} means set (it stores unique value, cannot access value from index{whereas can access in () and[])