

Python Fundamentals

By Menna Jaheen

String Concatenation :

```
age = 21  
print("you are " + str(age) + "years old")
```

Separate arguments :

```
print ("You are " , age , " years old " )
```

F-string:

```
print (f "you are {age} years old")
```

Tips for variables :

Instead of declaring multiple variables on multiple lines , you can do this :

```
x , y , z = 1,2,3  
x=y=z =1
```

Type Casting :

- Type() → To print the datatype of a variable
- Explicit Casting → `age = float(age)`
- Implicit Casting → when dividing an integer by a float the output is float

Accepting user input :

Using the Input() function :

```
name = Input("Enter your name ")
```

NOTE :

The user input is a string so if you want to use it in any mathematical expression you need to convert it to a numerical datatype first .

Arithmetic Operators:

```
import math
```

```
x = 0
```

```
x+=1
```

```
x-=1
```

```
x*=2
```

```
x/=2
```

```
x**=2 ( x power 2 )
```

```
x%=2
```

IF Statement:

```
age = int (Input("Enter your age "))
```

```
if age>=100:
```

```
    print("you are too old to sign up " )
```

```
elif age >= 18 :
```

```
    print( " you are +18 ")
```

```
else:
```

```
    print("You must be +18 to sign up ")
```

Logical Operators :

Used in Conditional Statements [and , or , not]

and : Checks if two or more conditions are true

or : Checks if at least one condition is true

not : True If condition is false and Vice Versa

```
if age >=18 and student == true :
```

```
    print("you can Sign up ")
```

Conditional Expression:

A one-line shortcut for the if else statement (ternary Operator)

X if condition else y

```
result = "EVEN" if num%2==0 else "ODD"
```

```
max_num = a if a >b else b
```

Format Specifiers :

{value : flags} format a value based on what flags are inserted

- (number)f = round to that many decimal places
- (number) = allocate that many spaces
- 0(number) = allocate and zero pad that many spaces
- < = left justify
- > = right justify
- ^ = center align
- + = use a plus sign to indicate positive value
- = = place sign to leftmost position
- : = insert a space before positive numbers
- , = comma separator
- % = percentage format

```
price1 = 3.14159
price2 = -987.65
price3 = 12.34
print(f"price1 is: ${price1:.2f}") # 3.14
print(f"price2 is: ${price2:.3f}") # -987.650
print(f"price3 is: ${price3:.2f}") # 12.34
```

While Loop :

perform some code WHILE some condition remains true

```
age = int(input("Enter your age: "))
while age < 0:
    print("Age can't be negative")
    age = int(input("Enter your age: "))
print(f"You are {age} years old")
```

For Loop :

- execute a block of code a fixed number of times.
- You can iterate over a range, string, sequence, etc.

Example :

```
credit_card = "1234-5678-9012-3456"

for x in credit_card:
    print(x)
```

Continue :

- When the program execution reaches a `continue` statement, the program execution immediately jumps back to the start of the loop.

```
for x in range(1, 21):
    if x == 13:
        continue
    else:
        print(x)
```

Break:

- If the execution reaches a `break` statement, it immediately exits the `while` loop's clause:

```
for x in range(1, 21):
    if x == 13:
        break
    else:
        print(x)
```

Nested Loop :

A loop within another loop (outer, inner)

outer loop:

inner loop:

Example:

```
rows = int(input("Enter the # of rows: "))
columns = int(input("Enter the # of columns: "))
symbol = input("Enter a symbol to use: ")

for x in range(rows):
    for y in range(columns):
        print(symbol, end="")
```