**-type casting = The process of converting a value of one data type to another**

**But what if we want to know the type of the variable ?**

**Use type()**

**//Example section**

**Explicit type casting :**

**Variable-name = int(variable\_name)**

**Variable-name = float(variable\_name)**

**Variable-name = str(variable\_name)**

**Variable-name = bool(variable\_name)**

**Anything but zero will always be true,and you can use it to check if someone write his name or not as an empty string will output false**

**Implicit type casting :**

**X=2**

**Y=2.0**

**X=x/y**

**Print(x)**

**The output will be a float value even though x was an int value;**

**-How to accept user input ? :**

**Name =input(“what is your name? : )**

**Age = input(“what is your age ? : )**

**Age+=1**

**This will give you an error as you can’t concatenate an int to a string**

**Instead we need to cast it to an int data type 🡪**

**Age = int(input(“what is your age ? : ))**

**If you want to round your output 🡪**

**round(datatypeToBeRounded , decimal places)**

**-Python arithmetic operators and math module**

**Math Operators**

**From highest to lowest precedence:**

| **Operators** | **Operation** | **Example** |
| --- | --- | --- |
| **\*\*** | **Exponent** | **2 \*\* 3 = 8** |
| **%** | **Modulus/Remainder** | **22 % 8 = 6** |
| **//** | **Integer division** | **22 // 8 = 2** |
| **/** | **Division** | **22 / 8 = 2.75** |
| **\*** | **Multiplication** | **3 \* 3 = 9** |
| **-** | **Subtraction** | **5 - 2 = 3** |
| **+** | **Addition** | **2 + 2 = 4** |

| **Operator** | **Equivalent** |
| --- | --- |
| **var += 1** | **var = var + 1** |
| **var -= 1** | **var = var - 1** |
| **var \*= 1** | **var = var \* 1** |
| **var /= 1** | **var = var / 1** |
| **var //= 1** | **var = var // 1** |
| **var %= 1** | **var = var % 1** |
| **var \*\*= 1** | **var = var \*\* 1** |

-Control flow in python

if = Do some code IF condition is True

else = Do something else if above condition/s are False

if (something ) :

if true DO THIS

else :

if it’s false DO THIS

-Logical operators = used on conditional statements

and = checks two or more conditions are True

or = checks if at least one condition is True

not = True if condition is False, and vice versa

-Ternary Conditional Operator

it offers one-line code to evaluate the first expression if the condition is true, and otherwise it evaluates the second expression.

<expression1> if <condition> else <expression2>

-STRING METHODS

len(name) 🡪

Evaluates to the integer value of the number of characters in a string, list, dictionary, etc.:

stringName.find("desiredCharachter") 🡪

returns the first index encountered of that character if no found returns -1

stringName.capitalize() 🡪

returns the first letter of the string capitalized

stringName = stringName.upper() 🡪

returns the string capitalized

stringName = stringName.lower() 🡪

returns lower-cased string

stringName = stringName.isdigit() 🡪

Only returns true if string contains only digits

stringName = stringName.isalpha() 🡪

Only returns true if string contains only letters even white spaces wil be considered false

VariableName = stringName.count(" ") 🡪

returns an integer

stringName = stringName.replace("the old charachter", "the new one")