DATA TYPES

In [ ]:
```
#string

print("hello"[0])
print("hello"[-1])

# '[x]' ==> gives the character present in 'x th' place as like the above exampl

# in programming always start counting from ' 0 '

# any datatype inside quotation is always treated as string
```

```
h
o
```

In [ ]:
```
# integer

# always the number must be not in the quotation.

print("123"+"456") # if we try to add two strings, it will concatenate

# if the numbers are in the quotation, then it is always considered as a string

print(123+456) # if we try to add two integers, it will result in summation
```

```
123456
579
```

In [ ]:
```
# float

# float is a decimal point number, it too need not to be inside the quotation

print("16.6" + "446.6") # if we try to add two strings, it will concatenate

# if the floats are in the quotation, then it is always considered as a string

print(123.123+456.456) # if we try to add two floats, it will result in summatic
```

```
16.6446.6
579.5790000000001
```

In [ ]:
```
# boolean

# it has only two possible values called "True" or "False"

# always T and F must be capital in these words

print(1<2) # returns the boolean answer True/False

print(1==2)
```

```
True
False
```

type() and type conversion

In [ ]:
```
# type() ==> used to get the data type of something
```

```
name = "aswin"
age = 18
print(type(name))
print(type(age),'\n')

# instead of using variables here, we can directly enter the data which we need

print(type("aswin"))
print(type(18))
```

```
<class 'str'>
<class 'int'>

<class 'str'>
<class 'int'>
```

In [ ]:
```
# type error

# example, if we try to check the the length of a string using len(), let check
# because len() gives the length of the string data type only

print("string ==>",len("aswin")) # let check it with string

print("integer ==>",len(123)) # let check it with integer

# error will be produced in 2nd code

# string ==> 5
# ---------------------------------------------------------------------------
# TypeError                                 Traceback (most recent call last)
# <ipython-input-12-3c3a1166e154> in <cell line: 8>()
#       6 print("string ==>",len("aswin")) # let check it with string
#       7
# ----> 8 print("integer ==>",len(123)) # let check it with integer

# TypeError: object of type 'int' has no len()
```

```
string ==> 5
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[7], line 8
      1 # type error
      2
      3 # example, if we try to check the the length of a string using len(), let
check it will produce type error
      4 # because len() gives the length of the string data type only
      6 print("string ==>",len("aswin")) # let check it with string
----> 8 print("integer ==>",len(123)) # let check it with integer
     10 # error will be produced in 2nd code
     11
     12 # string ==> 5
   (...)
     19
     20 # TypeError: object of type 'int' has no len()

TypeError: object of type 'int' has no len()
```

In [ ]:
```
# what is the alternate for checking the length of the string ?

# we are just going to convert the integer or other datatype into string, so tha
```

```
a = 123
s = str(a)
print(type(a))
print(type(s))
print(len(s))
# this is known as type conversion or type casting
```

```
<class 'int'>
<class 'str'>
3
```

In [ ]:
```
# type conversion

# changing on data type into another

# example

age = 18  # here the given data type is an integer, and we are simply going to c

print("data type of age ==>",type(age))

s_age = str(age) # just convert the datatype by giving the datatype into the dat

print("data type of s_age ==>",type(s_age))

# therefor we had converted the integer datatype into string data type
```

```
data type of age ==> <class 'int'>
data type of s_age ==> <class 'str'>
```

exercise 1

write a program that adds the digits in a two digit number. e.g. if the input was 35, then the output should be 3 + 5 = 8

NOTE: input() are always identified as strings

warning: write your code at the line which is allotted to you. your program should work for different inputs. e.g. any two - digit - number

sample input: 25

sample output: 2 + 5 = 7

In [ ]:
```
# don't change the code below

two_digit_number = input("Type a two digit number")

# don't change the code above
a = two_digit_number[0]
b = two_digit_number[1]
A = int(a)
B = int(b)
print(A + B)

# write your code below this line

# write your code above this line
```

```
Type a two digit number 25
7
```

mathematical operations in python

In [ ]:
```
# + ==> addition

# - ==> subtraction

# * ==> multiplication

# / ==> division

# ** ==> exponential (or) to the power of

# // ==> division, which gives the rounded value, and not in float datatype

# when we use / operation, results in float value only even it is rounded.

# example

print(type(5/5))
```

```
<class 'float'>
```

In [ ]:
```
# THE PEMDAS RULE

# p ==> paranthesis
# e ==> exponential
# m ==> multipication
# d ==> division
# a ==> addition
# s ==> subtraction

# python interpreter always use pemdas rule to prioritize any math problem

print(3*(3+3)/3-3)
```

```
3.0
```

EXERCISE 2

write a program to calculate Body Mass Index (BMI) from the user's height and weight

CLUE: BMI = weight(kg)/height**2(m**2)

example input

weight = 80

height = 1.75

example output

80 / (1.75*1.75) = 26.122448979591837

26

In [ ]:
```
# don't change the code below
height = float(input("enter the height in m: "))
weight = float(input("enter the weight in kg: "))
# don't change the code above

# write your code belos this line
bmi=float(weight/(height**2))
BMI = int(bmi)
print(f'Your BMI is {BMI}')
# write your code above this line
```

```
enter the height in m:  1.75
enter the weight in kg:  80
Your BMI is 26
```

number manipulation

In [ ]:
```
# if we divide 8/3, then the answer will be ==> 2.666666666666...., it is a floa

# if we convert into integer, then the answer will bw ==> 2

print(8/3)

print(int(8/3))

# this is what we called as rounding the number

# we can also use "round()" function to get a rounded value

# we can also get upto 'x' decimal point which is nearer by using round function

# syntax ===> round(operation, x); where x is the number which indicates digits

print(round(8/3))

print(round(8/3 ,3))

print(8 // 3)  # returns the same divided value, but not the decimal value
```

```
2.6666666666666665
2
3
2.667
2
```

In [ ]:
```
# increment or decrement

# let take score = 0 initially, we need to add 10 to that score, what can we do.

# yes ==> score = score + 10

# another way is to increment it with 10

# i.e., ==> score += 10

# it is also called as manipulating value with the previous value

score = 0
print(score)
```

```
score += 10
print(score)
```

```
0
10
```

F-String

In [ ]:
```
# f - string is nothing but the optimized version of printing something

# syntax ==> print(f'user prompt/display{x}'); where x = any variable which neea

# example, let take score = 50, and let do it with both normal and optimized met

score =  50

# normal method

print("your score ==>"+str(score))

# we need to convert score into string and then we need to print that, quite tim

# optimized method

print(f'your score ==> {score}')

# no need to convert into string here, works in any data type
```

```
your score ==>50
your score ==> 50
```

EXERCISE 3

create a program using maths and f-string that tells us how many days, weeks, months, we have left if we live until 90 years old.

it will take your current age as the input and output a message with our time left in this format

"you have x days, y weeks, and z months left"

where x,y,z are replaced with actual calculated numbers

warning: your output should match the words in the example output precisely. you should round the results to the nearest whole number.

example input

56

example output

you have 12410 days, 1768 weeks, and 408 months left

In [ ]:
```
# don't change the code below

age = input('what is your current age ?')
```

```
# don't change the code above

# write the code below this line
Age = int(age)
years = 90 - Age
month = years * 12
weeks = years * 52
days = years * 365
print(f'Your age is {Age}')
print(f"You have {days} days, {weeks} weeks, and {month} months left.")
# write the code above this line
```

```
what is your current age ? 19
Your age is 19
You have 25915 days, 3692 weeks, and 852 months left.
```

PROJECT OF THE DAY: TIP CALCULATOR

write a program for tip calcuator using the user's bill amount.

==> ask the user to enter the bill amount

==> ask the user to enter the perentage which he likes to give as tip

==> ask the user to enter the number of persons to split the bill

example:

welcome to the tip calculator

what was the total bill ? $100

what percentage tip would you like to give ? 10, 12, or 15 ?12

how many people to split the bill ? 3

each person should pay: $37.33

In [ ]:
```
print("welcome to the tip calculator")
# ask the user to enter the bill amount
bill = float(input('What was the total bill? '))
# ask the user to enter the perentage which  he likes to give as tip
tip = int(input('Whats percentage tip would you like to give? 10, 12, or 15? '))
# ask the user to enter the number of persons to split the bill
people = int(input('How many people to split the bill? '))
# write your calculation code here
# To Calculate percentage multiply the bill with tip percentage and divide by 10
percentage = bill * (tip / 100)

total_bill = (percentage + bill) / people
Total = round(total_bill, 2)
print(f'Each person should pay {Total}')
```

```
welcome to the tip calculator
Each person should pay 37.33
```