Day 2.

Data Types, Numbers, Operations, Type Conversion, f-String.

```
name = print(len('Python'))
name ↵
o/p → 6
```

Instead of strings, what about other data type like numericals?

```
print(len(12345)) ↵
o/p → Error - object 'int' has no len()
```

Data types

Basic types are,

- string
- Integer
- Float
- Boolean

Strings.

```
print('Hello'[0])
o/p → H → subscripting
          (pulling one value from the chain)
```

```
H e l l o
↓ ↓ ↓ ↓ ↓
0 1 2 3 4
```

```
print('Hello'[4])
o/p → o
```

Anything that comes inside the single/double quote, we call it has string.

```
print('123'+'456')
o/p → 123456
```

Integer:-

- numbers without any decimal places
- Print(123+456)
  o/p → 579

Large Integers => 342,654,896

342_654_896

(Instead of commas, we can use underscore to visualize the number)

Floating:-

- Numbers with decimal places
- Floating point

print (3.4152)

O/P → 3.4152

Boolean:-

- True / False

Quiz:-

which statement is incorrect?

982 is an integer ✓

"False" is a Boolean ✗

857.25 is a float ✓

"523" is a string ✓

what is the data type of the mystery variable?

mystery = 784_529.678

Integer ✗ Quotle ✗

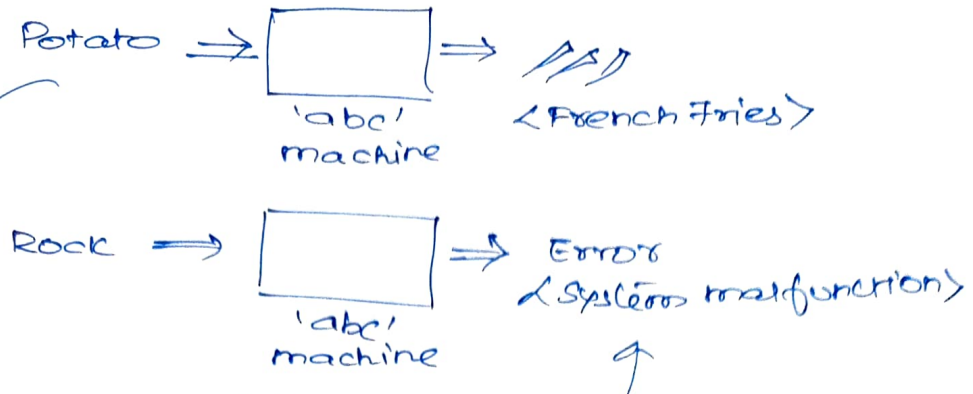
String ✗ Float ✓

I have put a spell on you. You are now a computer. If I give you the following code, what will you print out?

```
street_name = "Abbey Road"
print (street_name[4] + street_name[7])
```

```
Abbey Road
0123456789
```

yo ✓



print (len('python')) → 6

print (len (11234)) → Error <Type Error>

num_char = len (input ("what is your name?"))

print ("Your name has " + num_char + "characters.")

o/p → what is your name? Premanand

Type Error : concatenate only strings not num

type ( ) → gives the data info.

type(num_char)

o/p → int

String
int → sync

Type conversion :-

num_char = len (input("what is your name?"))

new_num_char = str (num_char)

print("your name has " + new_num_char + "characters.")

o/p :- what is your name? Premanand

your name has 9 characters

All datas are same → string (concatenation is possible)

a = 123

type(a) → int

a = str(123)

type(a) → str

a = float(123)

type(a) → float

float

Print (70 + float ("100.5"))

o/p → 170.5

print (str(70) + str(100))

o/p → 70100

Ex:-

write a program that adds the digits in a 2 digit number. Eg if the input was 35, then the o/p should be 3+5 = 8.

Sol:-

```
two_digit_number = input()
print(type(two_digit_number))
first_digit = int(two_digit_number[0])
second_digit = int(two_digit_number[1])
two_digit_number = first_digit + second_digit
print(two_digit_number)
```

Mathematical operation

3+5 → Add

7-4 → Sub

3*2 → Mul

print(6/3) → Div

↳ o/p of division is always in float ⇒ o/p = 2.0

2**2 → exponential power

PEMDAS

- Parenthesis
- Exponential
- Multiplication
- Division
- Addition
- Subtraction

Left ↔ Right (Direction)

Ex:-

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

9 + 3/3 - 3

9 + 1.0 - 3

10.0 - 3

7.0

for visualization understanding

⇓

Thonny IDE

for the same mathematic equation, how can I get the o/p as 3.0

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

o/p → 3.0

Parenthesis importance

Ex:-

write a program that calculates the Body Mass Index (BMI) from a users weight & height.

$$BMI = \frac{weight\ (kg)}{height^2\ (m^2)}$$

Sol:-

```
height = input( )
weight = input( )

weight_as_int = int(weight)
height_as_float = float(height)
bmi = weight_as_int / height_as_float ** 2
```

(or)

```
bmi = weight_as_int/(height_as_float *
                     height_as_float)
bmi_as_int = int(bmi)
print(bmi_as_int)
```

Number manipulation & F-strings:-

```
print(8/3)          o/p → 2.666666666
print(int(8/3))     o/p → 2
print(round(8/3))   o/p → 3 (>2.6)
print(round(8/3,2)) o/p → 2.67 (>2.666)
                      ↑
                  no. of places as decimal
Print(round(2.6666666,2)) o/p → 2.67
print(8//3)  o/p → 2
```

↓↓

float we will get it in division, inorder to get it as int use floor division

```
print (type (8//3))      o/p → int
print (type (8/3))       o/p → float
print (type (4/4))       o/p → float

result = 4/2 ⇒ 2.0
result /= 2 ⇒ result = result/2
print (result)  → o/p → 1.0

score = 0
score = score + 1
    ↳ score += 1
      score -= 1
      score *= 1
      score /= 1
print (score)

score = 0
print ("your score is" + score)  → o/p ⇒ Error
print ("your score is " + str(score)  → o/p ⇒ ✓

score = 0                    f-string
height = 1.8
isWinning = True
print (f'your score is {score}')  → your score is 0
print (f'your score is {score} and height is {height}
        and winning is always {isWinning}')
```

Ex:-

Your life in weeks and realized just how little time we actually have - Tim Urban

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

Sol:-

```
age = input()
age = int(age)
no_of_weeks_per_year = 52
no_of_years_from_90 = 90 - age
no_of_weeks_we_live = no_of_years_from_90 *
                      no_of_weeks_per_year
print(f'you have {no_weeks_we_live} weeks left.')
```

Quiz:-

You are a computer. what will this line of code print?

```
print(6+4/2-(1*2))
```

6 + 2.0 - 2
(2) (1)
8.0 - 2 ⇒ 6.0
(3)

3 ✗
6.0 ✓
8.0 ✗
5 ✗

what is the data type of the result of the variable a in the following line of code,

```
a = int('5')/ int(2.7)
```

5/2 ⇒ 2.5

int ✗
float ✓
str ✗
bool ✗

which of the following lines of code gives error?

```
age = 12
print('you are' + age + 'years old')
```

o/p ⇒ TypeError

```
age = 12
Print(f'you are {age} years old')

name= input('what is your name?')
Print('Hello,' +name)

name= input('what is your name?')
Print(f'Hello, {name}')
```

Project:-

```
print ('Welcome to TIP calculator!')
bill = input ('What was the total bill? Rs:')
bill = float (bill)
tip = input ('How much tip would you like
      to give? 10, 12 or 15?')
tip = int (tip)
people = int ('How many people to split the bill?')
bill_with_tip = tip/100 * bill + bill
print (bill_with_tip)
bill_person = bill_with_tip / people
final = round (bill_person, 2)
print (f'Each person should pay {final}/-')
```

→ final = "{:.2f}".format (bill_person)

⇓

If we need to format the display like 31.6, but I need 2 decimal point