**Day 1**

**Printing, Commenting, Debugging, String Manipulation and Variables**

**Print()**

print(“Hello World”) # Output Hello World

print(“Hello World \n Yash \n Verma”)

Output

Hello World

Yash

Verma

print(“””Yash

Verma”””)

Output

Yash

Verma

print(“Yash” + “ “ + “Verma”)

output

Yash Verma

**Spaces are very important in Python**

\_print(“Hello”)

It is error (space before print)

**input() or input(“prompt”)**

input(“What is your name”)

print(“Hello” + input(“What is your name?”))

print(“Hello” + “ “ + input(“What is your name”))

**Variable (store data into variables)**

name=”Jack” # Here name is variable and jack is data which is stored in variable name

print(name)

output

Jack

**len() (to find length of the string)**

name = input(“What is your name?”)

length=len(name)

print(length)

Output

It will length of string data stored in variable name.

**Variable name rules**

We cannot add space in variable

Numbers cannot be at beginning of variable

Functions cannot be used as variable

**Debugging Practice**

print("Day 1 - String Manipulation")

print('String Concatenation is done with the "+" sign.')

print('e.g. print("Hello " + "world")')

print("New lines can be created with a backslash and n.")

**Input Function**

Write a program that prints the number of characters in a user's name.

print(len(input('What is your name?')))

**Variables**

Write a program that switches the values stored in the variables a and b.

For example

Input:

a: 3

b: 5

Ouptut:

a: 5

b: 3

a=input("a: ")

b=input("b: ")

print("a: "+b)

print("b: "+a)

**Project Band Name Generator :**

print("Welcome to the band name generator.")

city = input("Which city did you grow up in? \n")

pet = input("What is the name of a pet? \n")

print("Your band name could be " + city + " " + pet)

**Day 2**

**Data Types, Numbers, Operations, Type Conversion, f-string**

print(len(“hello”))

5

print(len(123))

error

**Data Types**

String

Integer

Float

Boolean

**String**

“Hello” # this is a string

**Subscript**

H E L L O ### indexing

0 1 2 3 4

print(“Hello”[4])

output

o

print(“Hello”[0]) # Output H

“Hello” is a string

“123” is a string

print(“123” + “345”)

123345

**Integer (number without decimal)**

print( 123 + 345 )

output

468

**Large numbers:**

342,654,896

In python we can replace these commas with underscore

print( 1\_2 + 3)

output 15

**Float**

3.14159

**Boolean**

True (capital T) (no inverted commas)

False (capital F) (no inverted commas)

**len function errors**

len(4837)

Error

#len function does not work with integer

**len code (Error Code)**

num\_char = len (input(“What is your name?”))

print(“Your name has” + num\_char + “characters”)

it will show you error because string cannot be added with numbers.

**type() # it will tell you type of data**

n=len(input(“What is your name?”))

print(type(n))

output

What is your name? Yash

<class ‘int’>

**len code correct**

num\_char=len(input("What is your name? "))

new\_num\_char=str(num\_char)

print("Your name has " + new\_num\_char + " characters.")

**Type Conversion**

a=123

print(type(a)) # output <class ‘int’>

a = str(123)

print(type(a))

output string

a = float(123)

print(type(a))

output float

print(70 + float(“100.5”))

170.5

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

70100

**Mathematical Operations :**

+ Addition

- Subtraction

\* Multiplication (astrix)

/ Division

\*\* Exponential (Power)

// Integer Division

% Remainder

3 + 5

7 – 3

3\*2

print(6/3)

output 2.0

print(type(6/3)) ### output float

P E M D A S

() \*\* \* / + -

But in Python PEMDAS is different

For python : PEMDASLR

Highest Priority

()

\*\*

\*/ Same Priority

+ - Same Priority

Lowest Priority

### Mathematical Calculations goes from left to right.

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

7.0

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

3

**Data Types**

Write a program that adds the digits in a 2 digit number. e.g. if the input was 35, then the output should be 3 + 5 = 8

# 🚨 Don't change the code below 👇

two\_digit\_number = input("Type a two digit number: ")

# 🚨 Don't change the code above 👆

####################################

#Write your code below this line 👇

print(int(two\_digit\_number[0])+int(two\_digit\_number[1]))

**BMI Calculator**

Write a program that calculates the Body Mass Index (BMI) from a user's weight and height.

The BMI is a measure of some's weight taking into account their height. e.g. If a tall person and a short person both weigh the same amount, the short person is usually more overweight.

The BMI is calculated by dividing a person's weight (in kg) by the square of their height (in m)

Warning you should convert the result to a whole no.

height = float(input("enter your height in m: "))

weight = int(input("enter your weight in kg: "))

BMI = int(weight/(height\*\*2))

print(BMI)

**Number Manipulation and f-string**

print(8/3)

2.66666

print(int(8/3))

2

print(round(8/3))

3

print(round(8/3,2))

2.67

print(8//3)

2

print(type(8//3))

int

print(type(4/2))

float

print(4/2)

2.0

result = 4/2

result/=2 (This means result = result/2)

print(result)

output

1

score=0

score=score+1

score+=1

score-=1

score\*=1

**f-string**

f”your score is”

f”your score is {variable}”

**Practice Program Life in Weeks**

Create a program using maths and f-Strings 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.

There are 365 days in a year, 52 weeks in a year and 12 months in a year.

# 🚨 Don't change the code below 👇

#to take input from user.

age = input("What is your current age? ")

# 🚨 Don't change the code above 👆

#Write your code below this line 👇

#convert age from string to integer type to make calculations.

age=int(age)

#to find remaining years

years=90-age

#to find remaining days

x=years\*365

#to find remaining weeks

y=years\*52

#to find remaining months

z=years\*12

#print the output using f-String property

print(f"You have {x} days, {y} weeks, and {z} months left.")

**Project Tip Calculator**

#If the bill was $150.00, split between 5 people, with 12% tip.

#Each person should pay (150.00 / 5) \* 1.12 = 33.6

#Format the result to 2 decimal places = 33.60

#Tip: There are 2 ways to round a number. You might have to do some Googling to solve this.💪

#Write your code below this line 👇

#Welcome

print("Welcome to the tip calculator!")

#input bill from user

bill = float(input("What was the total bill? $"))

#input tip from user

tip = int(input("How much tip would you like to give? 10, 12, or 15? "))

#divide bill

s = int(input("How many people to split the bill? "))

#Final Bill Calculation

Final\_Bill = ((bill+(bill\*(tip/100)))/s)

#Round Off Bill to 2 decimal places

FB = round(Final\_Bill, 2)

#print the final result

print(f"Each person should pay: {FB}")

**Round of number to 2 decimal places**

a = 3.4593022

b = “{:.2f}”.format(a)

print(f”{b}, {type(b)}”)

output 3.46 < class ‘str’ >

#if a = 3.1 then output will be 3.10

a = 3.4593022

b = round(a,2)

print(f”{b}, {type(b)}”)

3.46 < class ‘float’ >

#if a=3.1 then output will be 3.1 .

**Day 3**

**Conditional Statements, Logical Operators, Code Blocks and Scope**

**Conditional if/else**

Depending on particular condition we would do either A or B.

if condition:

do this

else:

do this

**if/else**

water\_level = 50

if water\_level>80:

print("Drain Water")

else:

print("Continue")

Output

Continue

**Comparison Operators**

>Greater Than

<Less Than

>=Greater Than or Equal to

<=Less Than or Equal to

==Equal to

!=Not Equal to

**Code Odd or Even**

Write a program that works out whether if a given number is an odd or even number.

# 🚨 Don't change the code below 👇

number = int(input("Which number do you want to check? "))

# 🚨 Don't change the code above 👆

#Write your code below this line 👇

r = number%2

if r==0:

    print("This is an even number.")

else:

    print("This is an odd number.")

**Nested if statements or elif statements**

if condition:

if another condition:

do this

else:

do this

else:

do this

**if/elif/else**

if condition 1:

do A

elif condition 2:

do B

else:

do this

**Code to Calculate Leap Year**

Write a program that works out whether if a given year is a leap year. A normal year has 365 days, leap years have 366, with an extra day in February.

This is how you work out whether if a particular year is a leap year.

on every year that is evenly divisible by 4

\*\*except\*\* every year that is evenly divisible by 100

\*\*unless\*\* the year is also evenly divisible by 400

e.g. The year 2000:

2000 ÷ 4 = 500 (Leap)

2000 ÷ 100 = 20 (Not Leap)

2000 ÷ 400 = 5 (Leap!)

So the year 2000 is a leap year.

But the year 2100 is not a leap year because:

2100 ÷ 4 = 525 (Leap)

2100 ÷ 100 = 21 (Not Leap)

2100 ÷ 400 = 5.25 (Not Leap)

# 🚨 Don't change the code below 👇

year = int(input("Which year do you want to check? "))

# 🚨 Don't change the code above 👆

#Write your code below this line 👇

if year%4==0:

    if year%100==0:

        if year%400==0:

            print("Leap year.")

        else:

            print("Not leap year.")

    else:

        print("Leap year.")

else:

    print("Not leap year.")

**BMI 2.0**

Write a program that interprets the Body Mass Index (BMI) based on a user's weight and height.

It should tell them the interpretation of their BMI based on the BMI value.

* Under 18.5 they are underweight
* Over 18.5 but below 25 they have a normal weight
* Over 25 but below 30 they are slightly overweight
* Over 30 but below 35 they are obese
* Above 35 they are clinically obese.

# 🚨 Don't change the code below 👇

height = float(input("enter your height in m: "))

weight = float(input("enter your weight in kg: "))

# 🚨 Don't change the code above 👆

#Write your code below this line 👇

BMI = round((weight/(height\*\*2)))

if BMI <= 18.5:

    print(f"Your BMI is {BMI}, you are underweight.")

elif BMI <= 25:

    print(f"Your BMI is {BMI}, you have a normal weight.")

elif BMI <= 30:

    print(f"Your BMI is {BMI}, you are slightly overweight.")

elif BMI <= 35:

    print(f"Your BMI is {BMI}, you are obese.")

else:

    print(f"Your BMI is {BMI}, you are clinically obese.")

**Multiple if**

if condition 1:

do A

if condition 2:

do B

if condition 3:

do C

**Code Roller Coaster**

print("Welcome to the rollercoaster.")

height = int(input("What is your height in cm? "))

bill = 0

if height >= 120:

print("You can ride the Roller Coaster !")

age = int(input("What is your age? "))

if age < 12:

bill = 5

print("Child tickets are $5.")

elif age <= 18:

bill = 7

print("Youth tickets are $7.")

else:

bill = 12

print("Adult tickets are $12")

wants\_photo = input("Do you want a photo taken? Y or N. ")

if wants\_photo == "Y":

bill+=3

print(f"Your final bill is {bill}")

else:

print("Sorry, you have to grow taller before you can ride.")

**Pizza Ordering Code**

Congratulations, you've got a job at Python Pizza. Your first job is to build an automatic pizza order program.

Based on a user's order, work out their final bill.

Small Pizza: $15

Medium Pizza: $20

Large Pizza: $25

Pepperoni for Small Pizza: +$2

Pepperoni for Medium or Large Pizza: +$3

Extra cheese for any size pizza: + $1

# 🚨 Don't change the code below 👇

print("Welcome to Python Pizza Deliveries!")

size = input("What size pizza do you want? S, M, or L ")

add\_pepperoni = input("Do you want pepperoni? Y or N ")

extra\_cheese = input("Do you want extra cheese? Y or N ")

# 🚨 Don't change the code above 👆

#Write your code below this line 👇

bill = 0

if size == "S":

    bill+=15

    if add\_pepperoni == "Y":

        bill+=2

    if extra\_cheese == "Y":

        bill+=1

elif size == "M":

    bill+=20

    if add\_pepperoni == "Y":

        bill+=3

    if extra\_cheese == "Y":

        bill+=1

elif size == "L":

    bill+=25

    if add\_pepperoni == "Y":

        bill+=3

    if extra\_cheese == "Y":

        bill+=1

print(f"Your final bill is: ${bill}")

**Logical Operators**

**A and B**

**C or D**

**not E**

**And (if you need both of the conditions as true)**

a=12

a>10 and a<13

T T

True

But,

a = 12

a>15 and a<13

F T

False

**Or (if you need only one of the condition as true)**

**(or if both are true then everything becomes true)**

**(only if both condition is false then output will be false)**

a=12

a>10 or a<13

T T

True

a = 12

a>15 or a<13

F T

False

a=12

a>13 or a<10

F F

False

**not (if True it will return False)**

**(if False it will return True)**

a=12

not a>15

output

True

**Love Calculator**

You are going to write a program that tests the compatibility between two people.

To work out the love score between two people:

Take both people's names and check for the number of times the letters in the word TRUE occurs.

Then check for the number of times the letters in the word LOVE occurs.

Then combine these numbers to make a 2 digit number.

# 🚨 Don't change the code below 👇

print("Welcome to the Love Calculator!")

name1 = input("What is your name? \n")

name2 = input("What is their name? \n")

# 🚨 Don't change the code above 👆

#Write your code below this line 👇

name = name1 + name2

name = name.lower()

t = name.count("t")

r = name.count("r")

u = name.count("u")

e = name.count("e")

true = t + r + u + e

l = name.count("l")

o = name.count("o")

v = name.count("v")

e = name.count("e")

love = l + o + v + e

score = str(true) + str(love)

score = int(score)

if score<10 or score>90:

    print(f"Your score is {score}, you go together like coke and mentos.")

elif score>40 and score<50:

    print(f"Your score is {score}, you are alright together.")

else:

    print(f"Your score is {score}.")

# The End

**Treasure Hunt Game**

print('''

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

| | | |

\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.=""\_;=.\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_

| | ,-"\_,="" `"=.| |

|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_"=.\_o`"-.\_ `"=.\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| `"=.\_o`"=.\_ \_`"=.\_ |

\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_:=.\_o "=.\_."\_.-="'"=.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_

| | \_\_.--" , ; `"=.\_o." ,-"""-.\_ ". |

|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_.\_" ,. .` ` `` , `"-.\_"-.\_ ". '\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

| |o`"=.\_` , "` `; .". , "-.\_"-.\_; ; |

\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_\_\_\_\_| ;`-.o`"=.\_; ." ` '`."\` . "-.\_ /\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_\_\_\_\_\_\_

| | |o; `"-.o`"=.\_`` '` " ,\_\_.--o; |

|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\_| ; (#) `-.o `"=.`\_.--"\_o.-; ;\_\_\_|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_|o;.\_ " `".o|o\_.--" ;o;\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_

/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_"=.\_o--.\_ ; | ; ; ;/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_

\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_"=.\_o--.\_ ;o|o; \_.\_;o;\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_

/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_"=.\_o.\_; | ;\_.--"o.--"\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_

\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_"=.o|o\_.--""\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_

/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_\_/\_\_\_\_\_ /

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

''')

print("Welcome to Treasure Island.")

print("Your mission is to find the treasure.")

#Write your code below this line 👇

choice1 = input('You\'re at a cross road. Where do you want to go? Type "left" or "right" \n').lower()

if choice1 == "left":

choice2 = input('You\'ve come to a lake. There is an island in the middle of the lake. Type "wait" to wait for a boat. Type "swim" to swim across. \n').lower()

if choice2 == "wait":

choice3 = input("You arrive at the island unharmed. There is a house with 3 doors. One red, one yellow and one blue. Which colour do you choose? \n").lower()

if choice3 == "red":

print("It's a room full of fire. Game Over.")

elif choice3 == "yellow":

print("You found the treasure! You Win!")

elif choice3 == "blue":

print("You enter a room of beasts. Game Over.")

else:

print("You chose a door that doesn't exist. Game Over.")

else:

print("You get attacked by an angry trout. Game Over.")

else:

print("You fell into a hole. Game Over.")

**Day 4**

**Randomisation and Python Lists**

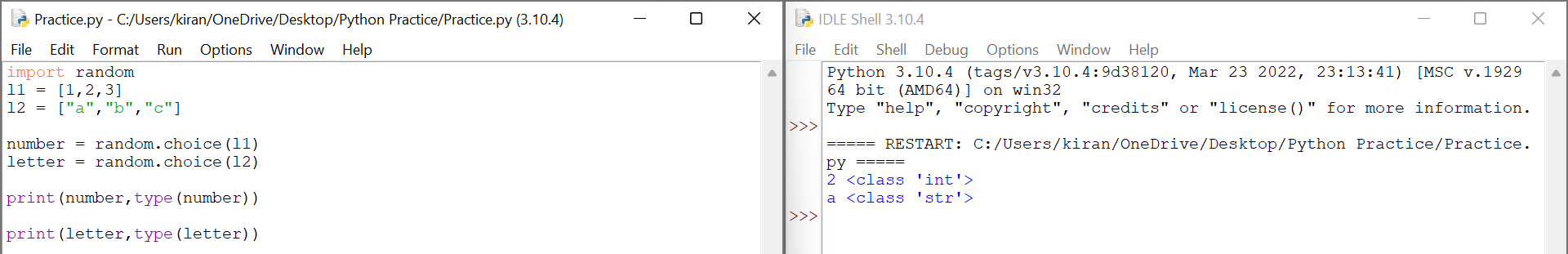
Randomisation is very important in a program if we want to make a game or to do something different each time, then we need to introduce a little bit of randomness, right?

By the end of the day we will create Rock, Paper and Scissors game…

**Random Module**

Randomisation : unpredictable ( random )

**Best Code for Random Module**



This code displays random item from the list no matter it is integer or string. random.choice(list)

import random

l1 = [1,2,3]

l2 = ["a","b","c"]

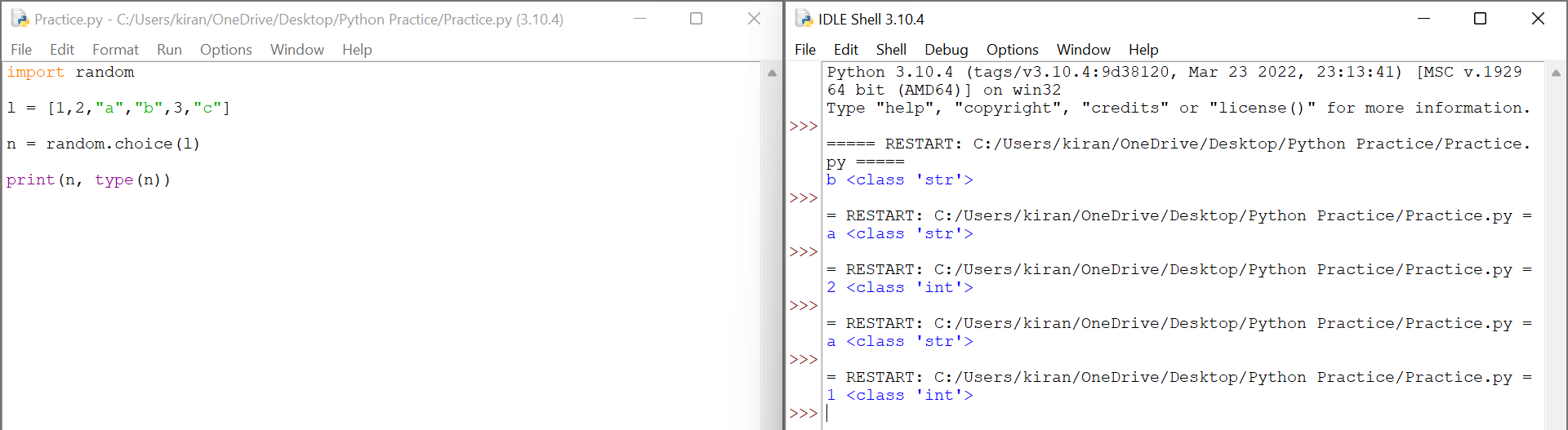
number = random.choice(l1)

letter = random.choice(l2)

print(number,type(number))

print(letter,type(letter))

**Another code for random.choice()**



import random

l = [1,2,"a","b",3,"c"]

n = random.choice(l)

print(n, type(n))

output of this code is shown in figure.

**Random module code**

**Code to print random integer**

import random

random\_integer = random.randint(1,10)

print(random\_integer)

# It always gives random number between 1 and 10, including 1 and 10.

**Module**

So what exactly is a module? Well,

Each module is responsible for a different bit of functionality of your program.

If we have lots of modules in a project so we can collaborate and divide the work between people and where each person can work on different things…

But how can we create our own modules and how do modules work anyways?

Create my\_mofule.py and add content to it and save anywhere on your pc

Content : pi=3.14159246

Now go back to main.py module and import my\_module module into that file ……………

Type following code to understand working of module:

import my\_module

print(my\_module.pi)

Snapshot Reference:





**Code to print random float:**

import random

random\_float=random.random()

print(random\_float)

# it prints random float numbers between 0 to 1 but it do not include 1.

# it goes from 0 to 0.99999999 , but it do not include 1.

**Code to print random float between 0 to any number for example 2, 3, 4 ,5 ……**

**But it will not include last number**

**If you want to print random float number between 0 to 5 …**

**It will include 0 but it will not include 5**

**It will be upto 4.99999999**

import random

randomFloat = random.random()\*5

print(randomFloat)

or

import random

randomFloat = random.random()

randomFloat = randomFloat\*5

print(randomFloat)

or

import random

randomFloat = random.random()

randomFloat \*= 5

print(randomFloat)

**Code coin toss head or tails :**

**## Heads or Tails**

**# UPDATE**

**We've moved away from repl.it for coding exercises.**

**Check out the new exercises on Coding Rooms with automated submissions.**

**Login to your Udemy course and head over to the link below to get the sign up link:**

**[Click here](https://www.udemy.com/course/100-days-of-code/learn/lecture/17825914#questions)**

**# Instructions**

**You are going to write a virtual coin toss program. It will randomly tell the user "Heads" or "Tails".**

**\*\*Important\*\*, the first letter should be capitalised and spelt exactly like in the example e.g. Heads, not heads.**

**There are many ways of doing this. But to practice what we learnt in the last lesson, you should generate a random number, either 0 or 1. Then use that number to print out Heads or Tails.**

**e.g.**

**1 means Heads**

**0 means Tails**

**# Example Output**

**```**

**Heads**

**```**

**or**

**```**

**Tails**

**```**

**# Solution**

**[https://repl.it/@appbrewery/day-4-1-solution](**[**https://repl.it/@appbrewery/day-4-1-solution**](https://repl.it/@appbrewery/day-4-1-solution)**)**

#Write your code below this line 👇

#Hint: Remember to import the random module first. 🎲

import random

random\_side = random.randint(0, 1)

if random\_side == 1:

print("Heads")

elif random\_side == 0:

print("Tails")

**Understanding the Offset and Appending Items to Lists**

**Python List (Data Structure)**

**Organise data in python using list**

Large group of data is stored …

You need order of data

We should have good data structure

**List**

Fruits = [item1, item2]

fruits = [“Cherry”, “Apple”, “Pear”]

**Order of the list**

states\_of\_america = ["Delaware", "Pennsylvania", “New York”]

# Order of List 0 1 3

-3 -2 -1

#List

states\_of\_america = ["Delaware", "Pennsylvania", "New York"]

print(states\_of\_america[-1])

print(states\_of\_america[-2])

print(states\_of\_america[-3])

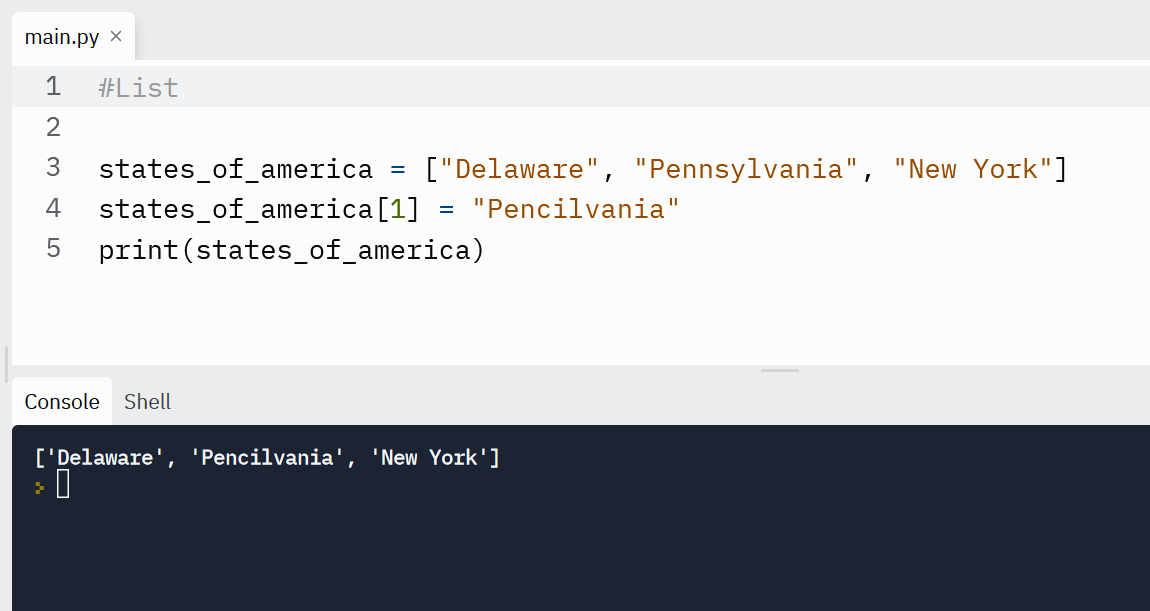
Output

New York

Pennsylvania

Delaware

**Change the items in the list**



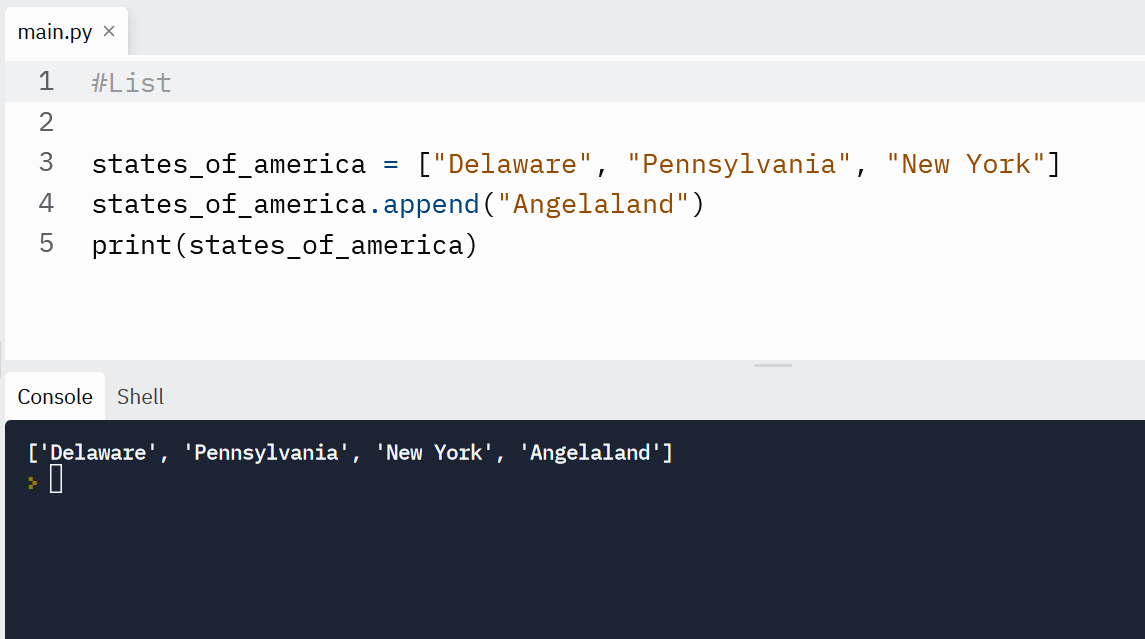
#List

states\_of\_america = ["Delaware", "Pennsylvania", "New York"]

states\_of\_america[1] = "Pencilvania"

print(states\_of\_america)

**add items to the end of the list .append()**



#List

states\_of\_america = ["Delaware", "Pennsylvania", "New York"]

states\_of\_america.append("Angelaland")

print(states\_of\_america)

**use back slash \ in printing strings…**

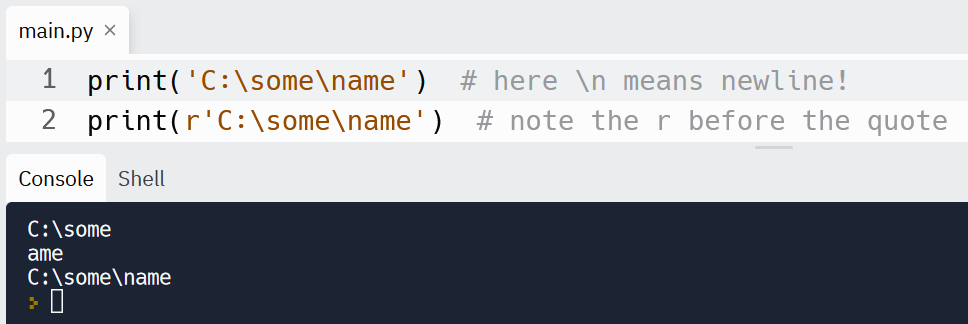
print('"Isn\'t," they said.')

**Raw Strings: print(r”Hello my \name is Yash Verma”)**

If you don’t want characters prefaced by \ to be interpreted as special characters, you can use raw strings by adding an r before the first quote:

print('C:\some\name') # here \n means newline!

print(r'C:\some\name') # note the r before the quote



**We can multiply string using \***

print(3\*"Yash ")

Output

Yash Yash Yash

print(3\*"Yash \n")

output

Yash

Yash

Yash

word = "Python"

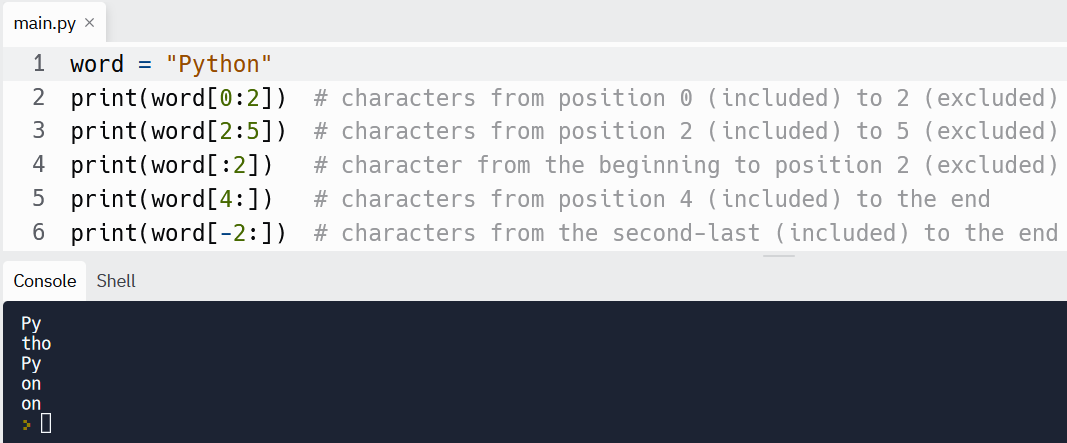
print(word[0:2]) # characters from position 0 (included) to 2 (excluded)

print(word[2:5]) # characters from position 2 (included) to 5 (excluded)

print(word[:2]) # character from the beginning to position 2 (excluded)

print(word[4:]) # characters from position 4 (included) to the end

print(word[-2:]) # characters from the second-last (included) to the end



**Note!!!**

**If you want to learn More About Python you can visit to Python Website**

**Python**

**Code Banker Roulette**

import random

# 🚨 Don't change the code below 👇

test\_seed = int(input("Create a seed number: "))

random.seed(test\_seed)

# Split string method

names\_string = input("Give me everybody's names, separated by a comma. ")

names = names\_string.split(", ")

# 🚨 Don't change the code above 👆

#Write your code below this line

l = len(names)

rname = random.randint(0, l - 1)

print(f"{names[rname]} is going to buy the meal today!")

## Instructions

You are going to write a program that will select a random name from a list of names. The person selected will have to pay for everybody's food bill.

**Important**: You are not allowed to use the choice() function.

**Line 8** splits the string names\_string into individual names and puts them inside a **List** called names. For this to work, you must enter all the names as names followed by comma then space. e.g. name, name, name

When you run the code, just use a random number as the seed. e.g. 67346 It doesn't matter what you chose, it's only for our testing code to check your work.

## Example Input

Angela, Ben, Jenny, Michael, Chloe

Note: notice that there is a space between the comma and the next name.

## Example Output

Michael is going to buy the meal today!

## Hint

1. You might need the help of the len() function.

<https://stackoverflow.com/questions/1712227/how-do-i-get-the-number-of-elements-in-a-list>

1. Remember that Lists start at index 0!

**Or**

import random

namestring = input("Enter Names :")

names = namestring.split(", ")

randn = random.choice(names)

print(f"{randn} will pay the bill!")

**IndexErrors and Working with Nested Lists :**

**List inside lists or Nested Lists**



fruits = ["Strawberries", "Nectarines", "Apples", "Grapes", "Peaches", "Cherries", "Pears"]

vegetables = ["Spinach", "Kale", "Tomatoes", "Celery", "Potatoes"]

dirty\_dozen = [fruits, vegetables]

print("dirty\_dozen = ", dirty\_dozen)





**Code Treasure Map**

# 🚨 Don't change the code below 👇

row1 = ["⬜️","⬜️","⬜️"]

row2 = ["⬜️","⬜️","⬜️"]

row3 = ["⬜️","⬜️","⬜️"]

map = [row1, row2, row3]

print(f"{row1}\n{row2}\n{row3}")

position = input("Where do you want to put the treasure? ")

# 🚨 Don't change the code above 👆

#Write your code below this row 👇

horizontal = int(position[0])

vertical = int(position[1])

map[vertical - 1][horizontal - 1] = "X"

#Write your code above this row 👆

# 🚨 Don't change the code below 👇

print(f"{row1}\n{row2}\n{row3}")

**Final Project Rock Paper Scissors Game**

**rock = '''**

**\_\_\_\_\_\_\_**

**---' \_\_\_\_)**

**(\_\_\_\_\_)**

**(\_\_\_\_\_)**

**(\_\_\_\_)**

**---.\_\_(\_\_\_)**

**'''**

**paper = '''**

**\_\_\_\_\_\_\_**

**---' \_\_\_\_)\_\_\_\_**

**\_\_\_\_\_\_)**

**\_\_\_\_\_\_\_)**

**\_\_\_\_\_\_\_)**

**---.\_\_\_\_\_\_\_\_\_\_)**

**'''**

**scissors = '''**

**\_\_\_\_\_\_\_**

**---' \_\_\_\_)\_\_\_\_**

**\_\_\_\_\_\_)**

**\_\_\_\_\_\_\_\_\_\_)**

**(\_\_\_\_)**

**---.\_\_(\_\_\_)**

**'''**

**#Write your code below this line 👇**

**import random**

**choice = [rock,paper,scissors]**

**user = int(input("What do you choose? Type 0 for Rock, 1 for Paper or 2 for Scissors. "))**

**if user<=-1 or user>=3:**

**print("You typed an invalid number... You Lose !")**

**else:**

**player = choice[user]**

**rn = random.randint(0,2)**

**bot = choice[rn]**

**if player == rock and bot == paper:**

**print(f"player \n {player} \n bot \n {bot} \n You Lose !")**

**elif player == rock and bot == scissors:**

**print(f"player \n {player} \n bot \n {bot} \n You win !")**

**elif player ==rock and bot == rock:**

**print(f"player \n {player} \n bot \n {bot} \n draw !")**

**elif player == paper and bot == scissors:**

**print(f"player \n {player} \n bot \n {bot} \n You Lose !")**

**elif player == paper and bot == rock:**

**print(f"player \n {player} \n bot \n {bot} \n You Win !")**

**elif player == paper and bot == paper:**

**print(f"player \n {player} \n bot \n {bot} \n draw !")**

**elif player == scissors and bot == rock:**

**print(f"player \n {player} \n bot \n {bot} \n You Lose !")**

**elif player == scissors and bot == paper:**

**print(f"player \n {player} \n bot \n {bot} \n You Win !")**

**elif player == scissors and bot == scissors:**

**print(f"player \n {player} \n bot \n {bot} \n draw !")**

**Day 5:**

**For Loops, Range and Code Blocks**

**Using the for loop with Python Lists**

for item in list\_of\_items:

#Do something to each item

fruits = ["Apple", "Peach", "Pear"]

for fruit in fruits:

print(fruit)

output

Apple

Peach

Pear

fruits = ["Apple", "Peach", "Pear"]

for fruit in fruits:

print(fruit)

print(fruit + "Pie")

output

Apple

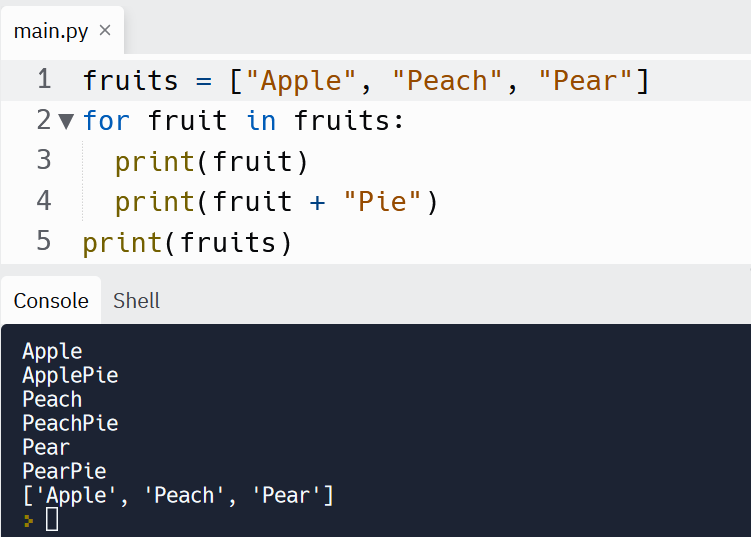
ApplePie

Peach

PeachPie

Pear

PearPie



fruits = ["Apple", "Peach", "Pear"]

for fruit in fruits: #it will assign variable to every item in list of fruits

print(fruit)

print(fruit + "Pie")

print(fruits) #this will be printed after loop is over…

**Code for Average Height**

**3 codes … Last one is best…**

# 🚨 Don't change the code below 👇

student\_heights = input("Input a list of student heights ").split()

for n in range(0, len(student\_heights)):

student\_heights[n] = int(student\_heights[n])

# 🚨 Don't change the code above 👆

#Write your code below this row 👇

items = 0

sum\_of\_height = 0

for i in student\_heights:

sum\_of\_height += i

items += 1

average\_height = sum\_of\_height/items

print(round(average\_height))

# 🚨 Don't change the code below 👇

student\_heights = input("Input a list of student heights ").split()

for n in range(0, len(student\_heights)):

student\_heights[n] = int(student\_heights[n])

# 🚨 Don't change the code above 👆

#Write your code below this row 👇

TotalHeight = sum(student\_heights)

NumberOfStudents = len(student\_heights)

AverageHeight = TotalHeight / NumberOfStudents

print(round(AverageHeight))

#End

#Understanding of map function

#Code for Average Height

StudentHeights = list(map(int,input("Enter Student Heights : ").split()))

print(f"List of Student Heights = {StudentHeights}")

TotalHeight = sum(StudentHeights)

print(f"Sum of Heights in list = {TotalHeight}")

NumberOfStudents = len(StudentHeights)

print(f"Total number of Students in list = {NumberOfStudents}")

AverageHeight = round(TotalHeight/NumberOfStudents)

print(f"Average Height = {AverageHeight}")

#END

**Code for Highest Score**

**2 codes… Last one is best…**

# 🚨 Don't change the code below 👇

student\_scores = input("Input a list of student scores ").split()

for n in range(0, len(student\_scores)):

student\_scores[n] = int(student\_scores[n])

print(student\_scores)

# 🚨 Don't change the code above 👆

#Write your code below this row 👇

HighestScore = 0

for score in student\_scores:

if score > HighestScore:

HighestScore = score

print(f"The highest score in the class is: {HighestScore}")

StudentScores = list(map(int,input("Enter Student Scores : ").split()))

print(f"List of Student Scores = {StudentScores}")

HighestScore = max(StudentScores)

print(f"Highest Score = {HighestScore}")

#END

**Using for loops with range function**

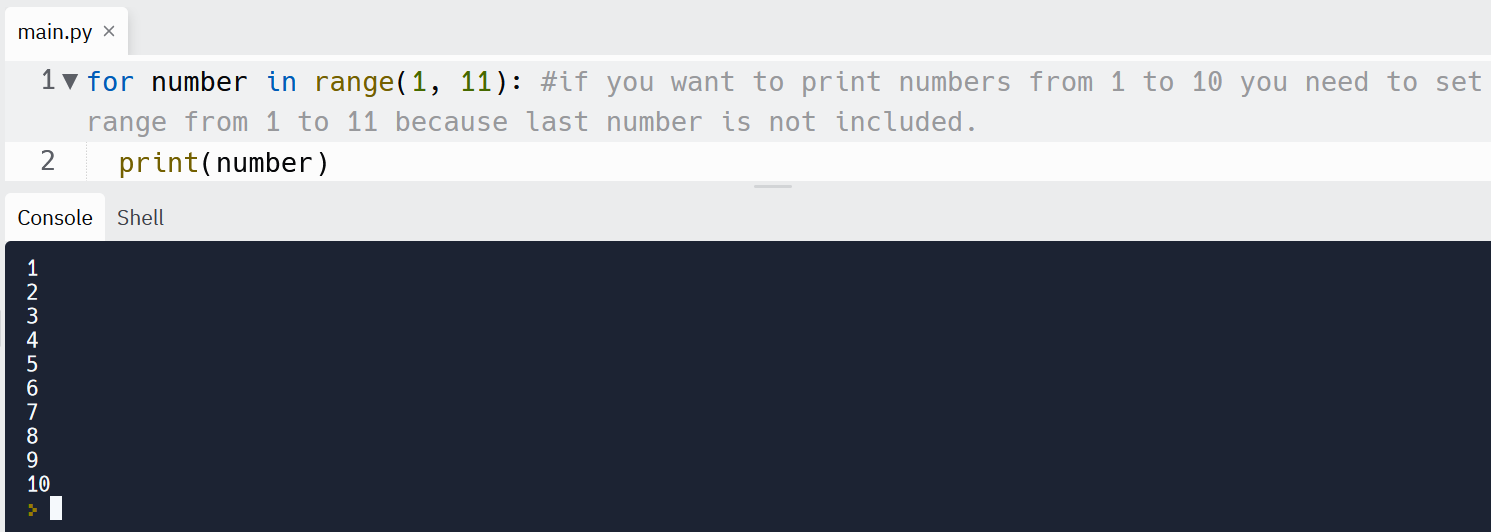
For number in range(a, b):

Print(number)



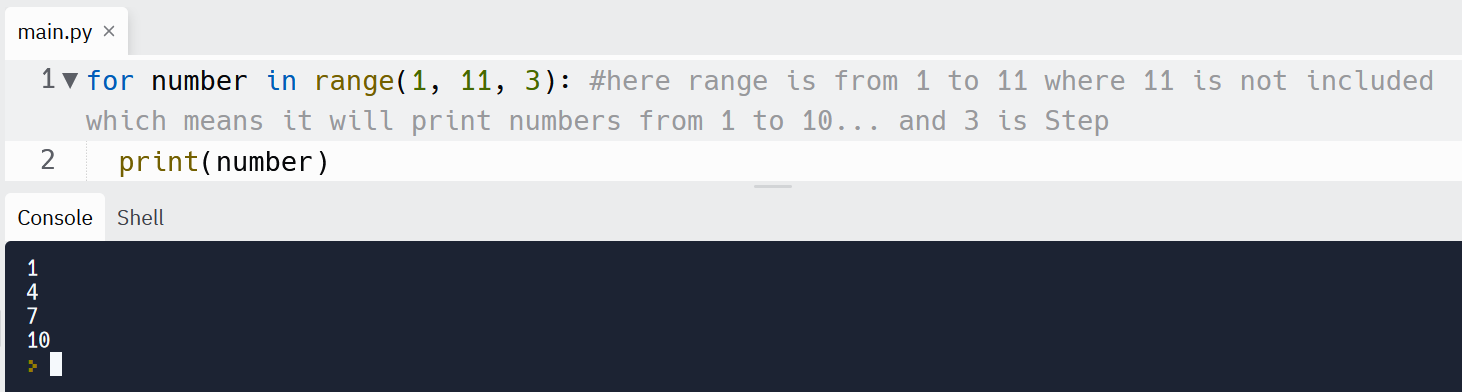
for number in range(1, 10): #range is from 1 to 10 ,but 10 is not included

print(number)



for number in range(1, 11): #if you want to print numbers from 1 to 10 you need to set range from 1 to 11 because last number is not included.

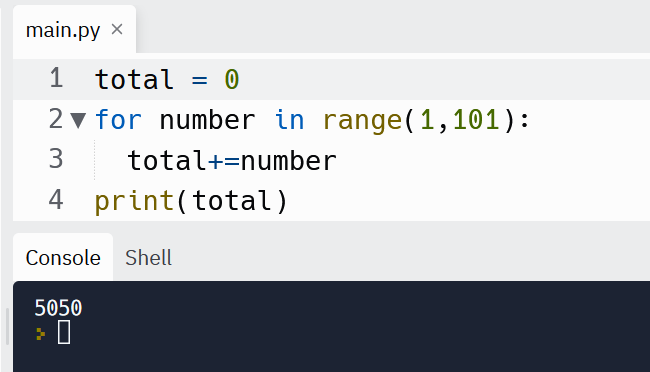
print(number)



for number in range(1, 11, 3): #here range is from 1 to 11 where 11 is not included which means it will print numbers from 1 to 10... and 3 is Step

print(number)

**Code to add numbers 1 to 100**



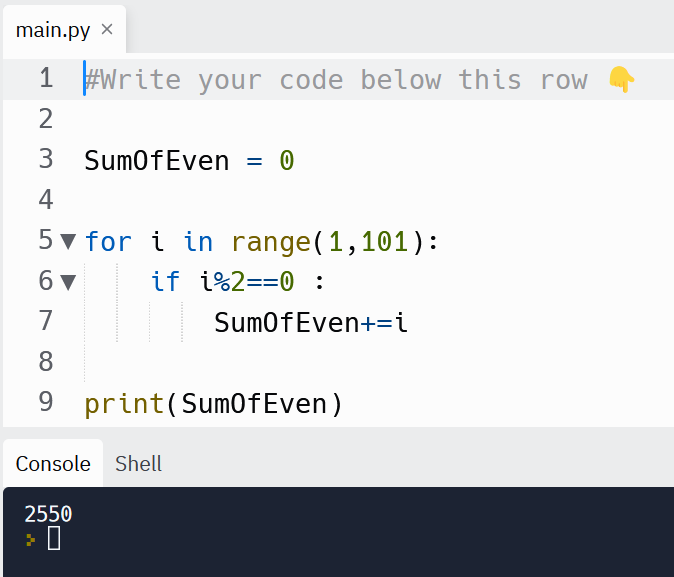
total = 0

for number in range(1,101):

total+=number

print(total)

**Code to print sum of even numbers in the range 1 to 100**



#Write your code below this row 👇

SumOfEven = 0

for i in range(1,101):

if i%2==0 :

SumOfEven+=i

print(SumOfEven)

**Another code for adding even numbers:**

total = 0

for i in range (2,101,2):

total+=i

print(total)

output

2550

**Code for Fizz Buzz Game**

You are going to write a program that automatically prints the solution to the FizzBuzz game.

Your program should print each number from 1 to 100 in turn.

When the number is divisible by 3 then instead of printing the number it should print "Fizz".

When the number is divisible by 5, then instead of printing the number it should print "Buzz".`

  And if the number is divisible by both 3 and 5 e.g. 15 then instead of the number it should print "FizzBuzz"

#Write your code below this row 👇

for n in range(1,101):

    if n % 3 == 0 and n % 5 == 0:

        print("FizzBuzz")

    elif n % 3 == 0:

        print("Fizz")

    elif n % 5 == 0:

        print("Buzz")

    else:

        print(n)

#END

**Project: Password Generator**

**Easy Password Generator:**

#Password Generator Project

import random

letters = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z', 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z']

numbers = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']

symbols = ['!', '#', '$', '%', '&', '(', ')', '\*', '+']

print("Welcome to the PyPassword Generator!")

nr\_letters= int(input("How many letters would you like in your password?\n"))

nr\_symbols = int(input(f"How many symbols would you like?\n"))

nr\_numbers = int(input(f"How many numbers would you like?\n"))

#Easy Level

password = ""

for i in range(0,nr\_letters):

letter = random.choice(letters)

password += letter

for i in range(0,nr\_symbols):

symbol = random.choice(symbols)

password += symbol

for i in range(0,nr\_numbers):

number = random.choice(numbers)

password += number

print(password)

#Eazy Level - Order not randomised:

#e.g. 4 letter, 2 symbol, 2 number = JduE&!91

#Hard Level - Order of characters randomised:

#e.g. 4 letter, 2 symbol, 2 number = g^2jk8&P

**HARD PASSWORD GENERATOR:**

#Password Generator Project

import random

letters = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z', 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z']

numbers = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']

symbols = ['!', '#', '$', '%', '&', '(', ')', '\*', '+']

print("Welcome to the PyPassword Generator!")

nr\_letters= int(input("How many letters would you like in your password?\n"))

nr\_symbols = int(input(f"How many symbols would you like?\n"))

nr\_numbers = int(input(f"How many numbers would you like?\n"))

#Hard Level

l = [letters, symbols, numbers]

total = nr\_letters + nr\_symbols + nr\_numbers

password = ""

for i in range(0, total):

character = random.choice(l)

password += random.choice(character)

print(password)

#Eazy Level - Order not randomised:

#e.g. 4 letter, 2 symbol, 2 number = JduE&!91

#Hard Level - Order of characters randomised:

#e.g. 4 letter, 2 symbol, 2 number = g^2jk8&P

#END