Python basics

- **Indentation matters**
- **Case sensitive**
- **dynamically typed variable**
- **No need for semicolon**
- **Comment using #**

What is google colab?

- Colaboratory, or “Colab” for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education. More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing free access to computing resources including GPUs.
- Google Colab Setup

Variable

- dynamically typed (C variables are statically typed)
- do not need explicit declaration
- declaration happens automatically when you assign a value to a variable

- Formatted string
- Typecast
- Take input

If-else statement

- **if**
- **elif**
- **else**

Operators

< <= > >= = != (similar to C)

For loop

```
for i in range(0, 9, 2):
    print(i)
```

While loop

```
i = 1
while i < 6:
    print(i)
    i += 2
```

List

```
arr = ['apple', 'grape', 100, 3.4, [1, 2] , (2, 3)]
print(arr)
```

- append('bird') - inserts bird at the last
- insert(1, 'bird') - inserts bird at index 1
- pop(1) - delete from index 1
- remove('bird') - remove the first occurrence of bird
- iterate through a list
- Slicing

Practice Problems

1. Write a python program that takes N numbers as input from the user and prints all possible pairs of numbers. Assume the N numbers are different from each other.

Sample Input	**Sample Output**
5 // N = 5 4 8 1 5 0	4 8 4 1 4 5 4 0 8 1 8 5 8 0 1 5 1 0 5 0

2. Write a program that prints the following series upto Nth term.
   1*2, 3*4, 5*8, 7*16, …

Sample Input	Sample Output
4	1*2, 3*4, 5*8, 7*16
6	1*2, 3*4, 5*8, 7*16, 9*32, 11*64

Dictionary

- "Key" : "value"

```
dictionary = {
    "name" : "fariha",
    "email" : "fariha@cse.uiu.ac.bd",
    "joined" : 2019
}
```

- keys() - returns all keys
- values() - returns all values
- get("name") - returns fariha
- dictionary["name"] - returns fariha
- dictionary.pop("email") - returns "fariha@cse.uiu.ac.bd" and deletes this key from dictionary
- "name" in dictionary - returns True
- Iterate

```
for key in dictionary:
    print(key, ':', dictionary[key])
```

Function

```
def minmax(arr):
  return min(arr), max(arr)
```

Practice Problems

3. Write a code that counts the frequency of each letter in a sentence.

Sample Input	Sample Output
I study in UIU	I: 2 : 3 s: 1 t: 1 u: 1 d: 1 y: 1 i: 1 n: 1 U: 2

4. Write a Python script to generate and print a dictionary that contains a number (between 1 and n) in the form (x, x*x)
5. Write a function that calculates the distance between two 2D points.

Sample Input	Sample Output
1 1 4 5	Distance 5

Class

```
class Employee:
  emp_num = 0 # works like static variable common to all objects of a class

  def __init__(self, first, last, pay): # constructor
    self.first = first # member variable
    self.last = last # member variable
    self.email = first + '.' + last + '@email.com' # member variable
    self.pay = pay # member variable
    apple = 1 # not member variable
    Employee.emp_num += 1 # static variable

  def fullname(self): # self is comparable to this keyword of Java
    return f'{self.first} {self.last}'# "%s %s", self.first, self.last

  def applyRaise(self, raise_):
    self.pay = int(self.pay * raise_)

  def __str__(self): # toString()
    return f'{self.fullname()} - {self.email}: {self.pay}'

  @staticmethod # same working procedure for all objects of a class
  def is_holiday(day):
    holidays = [1, 30, 49, 350]
    if day in holidays:
        return True
    else:
      return False

class Developer(Employee):
    def __init__(self, first, last, pay, prog_lang):
        super().__init__(first, last, pay) # call parent class constructor
        self.prog_lang = prog_lang
```

Practice Problems

6. Write a class Point to store the x,y coords of 2D points and write a distance function to calculate the Manhattan distance between two points.
7. Write a class Point to store the x,y coords of 2D points. The class should have the four methods move_left, move_right, move_up, move_down. move_left moves the point 1 cell to the left, move_right moves the point 1 cell to the right, move_up moves the point 1 cell up, and move_down moves the point 1 cell down. Also write a static distance function to calculate the euclidean distance between two points.
8. Write a class Student that stores a student's id, name and cgpa. Initially cgpa is 0. The class should have a function update_cgpa.
   Take the id and name of N students and store them in a dictionary using student id as key.
   Take the id and cgpa of M students and update their cgpa if the student is already in the dictionary, otherwise do nothing.
9. Write a program to take as input the name, height in cm and age of n employees, and sorts it according to height in descending order. If two employees have the same height then sorts according to age in descending order. Print the employees in the sorted order.

Numpy Array