***PYTHON FUNDAMENTALS***

**1. INPUT AND OUTPUT**

***PYTHON INPUT*:** Python wants to take input from the user.

***PYTHON OUTPUT*:** we use the Print () Function to allow this output data to the Standard output device.

* After all the values are printed end is Printed. If defaults to  a new line
* The file is the object where the values are printed and its default values are SYS.stdout (screen)

**NOTE:** python is case sensitive language

print("Hello world") # Hello world  
print("bibek") # bibek  
print("1st program in python") # 1st program **in** python  
print("5.9") # 5.9  
print("true") # true   
print("India","Odisha","ganjam") # India Odisha ganjam  
print("india","odisha","ganjam",**sep**="-") # india-odisha-ganjam  
print("Hello",**end**=' ')  
print("world") # hello world

**2. Comments**

Comments in Python are the lines in the code that are ignored by the interpreter during the execution of the program.

print("1st program in python") #1st program

**3. DATA TYPE**

* Every value in Python has a data type. since everything is an object in a Python program. Data type are actually classes and variables are instances (objects) of these classes.
* we can use the type () a variable or a value function Class function to know the class belongs to and the is an instance () to check the objections belonging to a particular class. no variables declaration.

**1. INTEGER**

print(4)  
print(1e309)

**2.FLOAT/DECIMAL**

print("4.9")   
**3. BOOLEAN**  
print(True)  
print(False)  
**4. complex**  
print("4+j")   
**5. string**  
print("Odisha")  
print('Odisha)  
print('"Odisha"')  
**6.list**  
print([9,0,7,7,4,7])    
**7. tuple**  
print((9,0,7,7,4,7))   
**8. Dictionary**   
print({"name":"bibek","age":"20","sex":"male"})

type():It is a built-in function that returns the type of the objects/data elements stored in any data type or returns a new type object depending on the arguments passed to the function.

type(7.7) # float

**4. VARIABLE**

* variable is a location of memory used to store some data.
* We don't need to declare the variable before using it. You can simply assign a value of a variable and its exists.
* The rules of writing variables same as the rules of writing identifiers in Python.
* We don't have need to declare the variable before using it. You can simply assign  a value of a variable and its exist.
* We don't even the declaration types of variables in handle internally according to the value assigned

**# to create a variable**   
name="Bibek"  
print(name)

a=5

b=10

print(a+b)

**NOTE:**

* **Dynamic typing**: The type of variable is determined by the run time.
* **Static typing**: The type of variable is determined in declared time.
* **Dynamic binding**: it is not possible to decide which piece of code will be executed as a result of a method call at compile-time.
* **Static binding:** Binding that takes place at compile time is known as static binding.

a = 5 # Dynamic Typing  
int a = 5 # Static Typing  
a = 5 # Dynamic Binding  
print(a)  
int a = 5 Static Binding

**5. KEYWORD**

 Keyword constant reserve words and you cannot use them as Or Variable or any other identifier name. The total number of keywords in Python is 33.

**import** keyword  
print(keyword.kwlist)  
Print ('\nTotal number of keywords:', len(keyword.kwlist))

['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlocal', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']

**6. IDENTIFIERS**

* A Python identifier is a name used to identify a variable, function class, module, or other object.
* An identifier starts with a letter (A-Z,a-z) or an underscore (\_) followed by zero or more letters.
* underscore and digits (o to 9)
* The following are naming conventions for python identifiers.
* Class name starts with an uppercase letter. All other identifiers in start with a lowercase letter.
* Starting an identifier with a single leading underscore indicates that the identifier is Private
* start an identifier with two leading underscores indicating a strongly private identifier.
* The identifier also ends with two trailing under ·Score (\_) the identifier is a language-defined special name.

name='bibek'  
print(name)  
  
\_='bibek'  
print(\_)  
  
first\_name='bibek'  
print(first\_name)

**7. TYPE CONVERSION**

#Type conversion is not a permanent operation  
first\_num= input("Enter the first number:")  
second\_num= input("Enter the second number:")  
result=int(first\_num)+int(second\_num)  
**print** (result)

Enter the first number:5

Enter the second number:6

11

#Another way to write this program   
first\_num=int(input("Enter the first number:"))  
second\_num=int(input("Enter the second number:"))  
result=first\_num+second\_num  
print(result)

Enter the first number:5

Enter the second number:6

11

9. LITERALS

a=0b1010 #binaryLITERALS  
b=100 #decimalLITERALS  
c=0.310 #ocralLITERALS  
d=0x12c #hexaLITERALS  
print(a,b,c,d)  
  
  
Float\_1=10.5  
Float\_2=1.5e2  
Float\_3=1.5e-3  
print(Float\_1,Float\_2,Float\_3)  
  
string='this is a python'  
string="this is a python"  
char="c"  
print(string)  
print(string)  
print(char)  
  
a=True+4  
b=False+10  
print("a;",a)  
print("b;",b)  
  
  
a=None  
print(a)  
  
unicode=u"\U0001f600\U0001F606\U0001F923"  
print(unicode)

TASK AND SOLUTION -1

**01. Print the given strings as per the stated format. Given**

input: "Data" "Science" "Mentorship" "Program"

"By" "CampusX"

output: Data-Science-Mentorship-Program-started-By-CampusX

print("Data","Science","Mentorship","Program",sep='-',end='-started-')  
print("By","CampusX",sep='-')

**OUTPUT:Data-Science-Mentorship-Program-started-By-CampusX**

02. Write a program that will convert the Celsius value to Fahrenheit.

celcius=float(input('Enter the temp in celsius:'))  
faren=celcius\*(9/5)+32 Enter the temp in celcius:43  
print(faren,'F') 109.4 F

**03. Take 2 numbers as input from the user. Write a program to swap the numbers without using any special python syntax.**

a=int(input("Enter the first number:"))  
b=int(input("Enter the second number:"))  
temp=a Enter the first number:  
a=b Enter the second number:  
b=temp value of a:  
print('value of a:',a) value of b:  
print('value of b:',b)

**04.  Write a program to find the Euclidean distance between two coordinates. Take both the coordinates from the user as input.**

p1x=int(input('Enter x cood of 1st point:'))  
p1y=int(input('Enter y cood of 1st point:'))  
p2x=int(input('Enter x cood of 2nd point:'))  
p2y=int(input('Enter y cood of 2nd point:'))  
distance=((p2x-p1x)\*\*2+(p2y-p1y)\*\*2)\*\*0.5  
print(round(distance,2))

**05. Write a program to find the simple interest when the value of principle,rate of interest and time period is provided by the user. Hint - si = (p \* t \* r)/100**

p=int(input('Enter amount:'))  
t=int(input('Enter time period:'))  
r=float(input('Enter rate:'))  
interest=(p\*t\*r)/100  
print('the interest is',interest)

**06. Write a program that will tell the number of dogs and chickens that are there when the user provides the value of total heads and legs.**

total\_heads=int(input("Enter the number of heads:"))  
total\_legs=int(input("Enter the number of legs:"))  
if total\_legs<total\_heads:  
 print("invalid input!")  
else:  
 dogs=(total\_legs-(2\*total\_heads))/2  
 chicken=total\_heads-dogs  
 print("Number of dogs in farm are:"+str(dogs))  
 print("Number of chicken farm are:"+str(dogs))

**07.  Write a program to find the sum of squares of first n natural numbers where n will be provided by the user. Hint - Thus, the sum of the squares of first n natural numbers = n(n+1)(2n+1)/6**

n=int(input("Enter the number:")) **output**:   
i=1 Enter the number:5  
sum=0 55.0  
while(i<=n):  
 sum=sum+(i\*i)  
 i=i+1  
print("sum= ",sum)  
#OR  
n=int(input('Enter the number:'))  
result=(n\*(n+1)\*(2\*n+1))/6  
print(result)

**8. Given the first 2 terms of an Arithmetic Series. Find the Nth term of the series. Assume all inputs are provided by the user.**

first\_term=int(input('Enter 1st term:')) Enter 1st term:3  
second\_term=int(input('Enter 2nd term:')) Enter 2nd term:6  
n=int(input('Enter the value of n:')) Enter the value of n:5  
d=second\_term-first\_term 15  
an=first\_term+(n-1)\*d  
print(an)

**09. Given 2 fractions, find the sum of those 2 fractions. Take the numerator and denominator values of the fractions from the user.**

n1=int(input('Enter the Num1:')) **output**:  
d1=int(input('Enter the Den1:')) Enter the Num1:13  
n2=int(input('Enter the Num2:')) Enter the Den1:17  
d2=int(input('Enter the Den2:')) Enter the Num2:22  
rn=n1\*d2+n2\*d1 Enter the Den2:211  
rd=d1\*d2 47/77  
print('{}/{}'.format(rn,rd))

**10. Given the height, width, and breadth of a milk tank, you have to find out how many glasses of milk can be obtained. Assume all the inputs are provided by the user.**

**import** math **output:**   
 Enter the height: 10  
h\_t = float(input('Enter the Height:')) Enter the breadth:19

Enter the length: 10  
b\_t = float(input('Enter the Breadth:')) The height of glass: 5  
l\_t = float(input('Enter the Length')) Radius of the glass: 2  
h\_g = float(input('The Height of glass')) no of glasses :30  
r\_g = float(input('Radius of the glass'))  
  
vol\_tank = h\_t\*b\_t\*l\_t  
vol\_glass = 3.14\*r\_g\*r\_g\*h\_g  
print('no of glasses',math.floor(vol\_tank/vol\_glass))