1. what is a tuple?

Sol: Tuple is a immutable object it will not do any modifications.

Coma separated elements are called tuple.

For example a = 10, 20, 30

```
>>> a
(10, 20, 30)
```

```
          >>> type(a)
         <class 'tuple'>
For eg: >>> a = "john",20,"delhi"
                  >>> a
           ('john', 20, 'delhi')
           >>> type(a)
           <class 'tuple'>
```

2. Write a program swap a variable?

Program:

```
                >>> a = input("input of
first number:")
                 input of first number:12
                >>> b = input("input of
second number:")
                input of second
number:14
                >>> a,b = b,a
```

output:

```
         >>> b
         '12'
        >>> a
        '14'
```

3.what is parallel assignment in python?

Sol: Tuple to tuple assignment is called parallel assignment. We assign more than one variable. For eg:

```
            a = 10
             >>> b = 20
              >>> a,b = b,a------ right
```

hand side is tuple of objects and

left hand side is the tuple of variables.

```
             >>> b
                 10
             >>> a
            20
           >>> a
          20
          >>> b
```

10

4.What is meant by introspective language?

Sol: For checking the capabilities of an objects, variables.

5.what is the built in function to find output type?

Sol: The python interpreter has many number of built in functions. For eg: print() method is used to print the output value.

6.what is meaning of identifies in python?

Sol: Identifier is used to identify the names,functions,classes,variables.

7. what is literals in python?

Sol: It is defined as data can be given in a variable.
For eg: string,unicode,tuple these are literals

8.what is an object in python?

Sol: python is the object- oriented language. Object is a collection of Variables and functions.

9.what is the compound statement?

Sol: It is also called conditional statements which are if,if-else,else if statements.

10.how to assign the function to variable? and how to call the function by using variable?

Sol: n1 = 4

```
n2 = 6
def sum(n1,n2):
    total = n1+n2
    return total
```

Output:

```
    sum
      <function sum at
0x03D1B468>
```

11. how to find out the identity of the object?

Sol: a = "Hello"--------> This for string

```
          >>> id(a)
         66163040
          >>> a = "25" ---------> This is for integer
          >>> id(a)
```

```
57507040
```

12. create a variable and assign value i.e a = 100, b = 100 find out the identity of a and   b and explain what is happening?

```
Sol:       a = 100
          >>> b = 100
           >>> id(a)
```

```
 1740401296
 >>> id(b)
1740401296
```

Here Both id’s are same why because we take small integers to find the identities.

```
For eg: a = 12345
>>> b = 12345
```

```
 >>> id(a)
66137728
>>> id(b)
66137776
```

Here, Both variables are same but the identities are difficult, It will change for longer values.

13. what is the output of the below code

```
        a = 10;
        a() ?     what is the output and explain ?
```

Sol: a = 10

```
>>> a()
Traceback (most recent call last):
  File "<pyshell#70>", line 1, in <module>
    a()
TypeError: 'int' object is not callable
```

When i am trying to call that variable it is already available with the same name.

```
a = 10
```

```
>>> a
10------------------------> This is the Modified version.
```

14.explain the output of the below code?

```
 def greet():
     print("Hi good Morning")
a = greet()
```

a() what is the output and explain ?

```
Sol: def greet():
    print("Hi good Morning")
>>> greet()
Hi good Morning
>>> a = greet
>>> a()
Hi good Morning
```

```
>>> def greet():
    print("Hi good Morning")
    a = greet
    a()
>>> a()
Hi good Morning
```

Here I can define a function as greet ,it will print the given string when that time i create a one variable and print that variable.

types, object, identifies

2. data types?

integers, long

how to represent the long for the integers

Float?

Sol: a = 19

```
>>> type (a)
<class 'int'>------------------->
```

This is for integer.

```
a=9765345687892155681233333
3333577777779874561232145698
7789542368745963147963324789
6321478852223336987411222222
55889632147896258741236
          >>> type(a)
         <class 'int'>---------------->
```

Here int and long are same.

```
  a = 45.6
```

```
>>> type(a)
<class 'float'>--------------->
```

This is a float value.

1.what is the size of the integer?

Sol: The size of integer is 32 -bits

2. what is the size of the long?

Sol: The size of the long is also same as int but long will take larger values.

3. what is the size of float?

Sol: The size of the float is 32 bit or 64 bits.

4. how to represent float?

Sol: a = 3.2

>>> type(a)

<class 'float'>--------------------------> It will represent the decimal points.

5. how to represent decimal representation?

Sol: bin()

6. how to represent hexadecimal representation?

Sol: a = 0x10-------> Representation of hexadecimal

```
>>> a
16
```

7. how to represent octal representation?

Sol: oct()

For eg: a = 0o01

```
            >>> a
              1
```

```
>>> a = 0o10
>>> a
8
```

8. give few example for all above?

Sol: For eg: binary number

```
a = 0b00
>>> a
  0
```

For eg: octal number

```
a = 0o10
>>> a
 8
```

For eg:Hexa decimal value

```
a = 0x10
>>> a
16
```

9. write a program which tells what type of data entered by user?

for example if user enters 10
==> type is integer
if user enter jhhhhff
==>type is string.

Sol: Code:

```
a = input("Enter the user: ")
print (type(a))
```

Output:

```
Enter the user: 45
<class 'str'>
>>>
 RESTART:
C:/Users/Sandya-Samara/AppData
/Local/Programs/Python/Python37-
32/user12.py
Enter the user: Sandhya
```

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

Date: 7/19

==========

Arithamatic operators:

+

-

*

/

%

** powerof

// floor dev operator.

Bitwise operations

&

|

~

Augmented assignment?

A = a+ 10   a+=10

Assignments:

1.Do the addition of two integers

```
Sol: a = 24
b = 32
c = a + b
print("The value is",c)
```

Output:

```
The value is 56
```

2.Do the addition of two strings.

```
Sol: a = "Hello"
b = " Welcome"
c = a + b
print("The string is",c)
```

Output:

```
The string is Hello Welcome
```

3.Do the addition of characters.

```
Sol: a = "A"
b = " B"
c  = a + b
print("the character is",c)
```

Output:

```
the character is A B
```

4.Do the subtraction of two integers?

```
Sol: a = 23
b = 21
c  = a - b
print("The subtraction value is",c)
```

Output: The subtraction value is 2

5.Do the subtraction of two characters?

Sol:

6.Do the subtraction of two strings

Sol:

7.Division of two integers?

```
Sol: a = 24
b = 32
c = a / b
print("The value is",c)
```

Output:

```
The value is 0.75
```

8.Division of two strings?

Sol:

Comment line:

Single line comments

Sol: print(“Hello World”) #This is a line comment

Multiline comments:

"""

shjfhhkjkfsrtytty

```
dfjjkgl;jkjlnfertytlkjnbvcx
zbnmxchgjkfxnvzgchdgkjnbm
afbdhgsajbv
sajhgruehgjdbvdj
"""
'\nshjfhhkjkfsrtytty\ndfjjkgl;jkjlnfertyt
lkjnbvcx\nzbnmxchgjkfxnvzgchdgkj
nbm\nafbdhgsajbv\nsajhgruehgjdb
vdj\n'
```

Date: 7/24/2019

=============

1. **Ask user to enter two integers and print the bitwise AND of these two integers.**

```
Sol: a = int(input("User enter first number: "))
       b = int(input("User enter second number:"))
       c = a&b
       print("The result is:",c)
```

Output:

```
         User enter first number: 12
         User enter second  number:21
The result is: 4
>>>
 RESTART: C:/Users/Sandya-Samara/AppData/Local/Programs/Python/Python37-32/AND.py
        User enter first number: 010
        User enter second  number:100
        The result is: 0
```

2 Ask user to enter two integers and print the bitwise
OR of these two integers.

Sol: #To write a program for bitwise OR.

```
             x = int(input("user enter first number: "))
             y =int(input("user enter second number: "))
              z = x | y
              print("The result is:",z)
```

Output:

```
          Eg:   user enter first number: 12
                user enter second number: 10
                The result is: 14
          Eg:     user enter first number: 0001
                 user enter second number: 0010
                 The result is: 11
```

3 Ask user to enter two integers and print the Exclusive OR of these
two integers.

Sol: #to write a program for bitwise Ex-OR

```
          x = int(input("user enter first number: "))
          y =int(input("user enter second number: "))
          z = x ^ y
          print("The result is:",z)
```

Output:

```
      Eg:     user enter first number: 24
              user enter second number: 12
              The result is: 20
      Eg:    user enter first number: 0111
             user enter second number: 0110
             The result is: 1
```

4. Ask user to enter a single integer and prints its bitwise complement.

Sol:

```
a = int(input("User enter single integer: "))
a = ~a #~a = -a-1
print("The result is :",a)
```

Output:

```
=== RESTART: C:/Users/Sandya-Samara/Desktop/sandhya programs/COMPLEMENT.py ===
User enter single integer: 0010
The result is : -11
>>>
=== RESTART: C:/Users/Sandya-Samara/Desktop/sandhya programs/COMPLEMENT.py ===
User enter single integer: 2
The result is : -3
>>>
=== RESTART: C:/Users/Sandya-Samara/Desktop/sandhya programs/COMPLEMENT.py ===
User enter single integer: 20
The result is : -21
```

5. What are the assignment statements?

Sol: An assignment statement is a way of assigning a value to the variable

```
Variable = expression # here = is the assignment
```

For eg:

```
#assignment statement
a = 2 #operand1
```

```
            b = 4 #operand2
            c = a*b # two operands are multiply expression is assigned to variable c
            print("The value is:",c)
```

Output: The value is: 8

6. What is assignment operator

Sol: For eg: a = 5

Assignment operator that assigns the value 5 on the right to the variable a on the left side

Here “ = ”,is the simple assignment operator.

7.. What is equality operator?

Sol: Equality operator means to verify the given values are True or False.

Symbol is “==”

```
For eg: a=2
            b=2
           c = ( a==b)
           print(“the result is:”,c)
```

Output:

```
          ===== RESTART: C:/Users/Sandya-Samara/Desktop/sandhya programs/Equal.py =====
            the result is : True
```

8. When you assign the result of a conditional expression to an variable, what could be the possible content of variable?

Sol:

9. Write a area_of_square() function which ask user to enter the size of a side of a square in centimeters. This function should read that value and should print the area of square in square centimeters. Use only integers.

Sol:#area of square

```
            def areaSquare(side): #(4)
              area = side * side #4*4 = 16
              return area

            side = int(input("user enter a value in sq.mts :"))#4
            print(“area of square in centimeters:”,area)
            print(areaSquare(side))
```

Output:

```
            user enter a value in sq.mts:16
           The area of square in centimeters: <function areaSquare at 0x03F4B468>
          256
```

10. Rewrite the above program by using a floating variables. But this program should read the size in centimeters and should print the area in square meters.

Sol: #area of square

```
          def areaSquare(side): #(4)
          area = side * side #4*4 = 16
         return area

         side = float(input("user enter a value in sq.mts:"))#4
         print("The area of square in centimeters:",areaSquare)
        print(areaSquare(side))
```

Output:

```
== RESTART: C:/Users/Sandya-Samara/Desktop/sandhya programs/areasqufloat.py ==
user enter a value in sq.mts:14
The area of square in centimeters: <function areaSquare at 0x0347E468>
196.0
```

11. Write a program having single main function. This function asks user to enter a length and breadth of a rectangle in centimeters. Use integer variable to read these length and breadth. Next the function should

print area and perimeter of the rectangle.

```
Sol:#area and perimeter of a rectangle
       def rectangle(l,b):
     area = l*b
     perimeter = 2*(l+b)
     Total = (area,perimeter)
     return Total

       x = rectangle(3,4)
        print ("Area and Perimeter = ",x)
```

Output:

```
                Area and Perimeter =  (12, 14)
```

12. Write a program having single main function(use any name of a function). This function asks user to enter length, breadth and height of a block in centimeters.
 Use integer variable to read these length, breadth and height.
 Next the function should print the surface area and volume of the
 Block.

```
Sol:#write a program to find volume and surface area of cuboid
l = int(input("Enter the length in mts:"))
b = int(input("Enter the breadth in mts:"))
h = int(input("Enter the height in mts:"))
volume = (l*b*h)
surfacearea = 2*((l*b)+(b*h)+(l*h))
print("The result  volume in cubic units:",volume)
print("The result of surface area in sq.units:",surfacearea)
```

Output:

```
Enter the length in mts:12
Enter the breadth in mts:24
Enter the height in mts:36
```

```
The result  volume in cubic units: 10368
The result of surface area in sq.units: 3168
```

13. Ask user to enter the radius of a circle in centimeters. Use floating point variable. Now compute the circumference in centimeters and and area in square meters. Also print the area of square whose perimeter is same as circumference of the above circle. This area also should be printed only in square meters. You can include math.h file and can use constant M_PI, which is set more precisely than 3.14.

Sol:

```
 radius = float(input("User enter the radius of circle in cms:"))
 PI = 3.14
circumference = 2*PI*radius
print("the cicumference in cms: %d cm.",circumference)
area = PI*(radius*radius)
print("the area in sqms: %d sqms.",area)
```

Output:

```
User enter the radius of circle in cms:12
the cicumference in cms: %d cm. 75.36
the area in sqms: %d sqms. 452.16
```

14. Ask user to enter principle, annual rate of interest and time in years. Use all float variables only. Print the interest and also total (i.e. interest + principle).

Sol: #program for interest

```
     P = float(input(" User enter the principle amount:"))
      T = float(input("User enter the time in years:"))
R = float(input("User enter the given rate of interest:"))
interest = (P*R*T)/100
```

```
print("Simple Interest : ", interest)
Total = (interest + P)
print("Total is :",Total)
```

Output:

```
 RESTART: C:/Users/Sandya-Samara/AppData/Local/Programs/Python/Python37-32/interest.py
User enter the principle amount:12000
User enter the time in years:2.5
User enter the given rate of interest:1.25
Simple Interest :  375.0
Total is : 12375.0
```

15. Read temperature in Fahrenheit degrees from the user and print it in Celsius degrees. Use floating point variables.

Formula: Celsius = (5.0/9.0) * (fahrenheit - 32)

Sol:#fahernheit to celcius

```
        fahrenheit = float(input("User enter a temparature in farenheit:"))
       Celsius = (5.0/9.0) * (fahrenheit - 32)
         print("The celcius degrees is:",Celsius)
```

Output:

```
=========== RESTART: C:/Users/Sandya-Samara/Desktop/fahrentocel.py ===========
User enter a temparature in farenheit:97.5
The celcius degrees is: 36.38888888888889
```

16. Read temperature in Celsius degrees from the user and print it in Fahrenheit degrees. Use floating point variables.

Sol: #convert celsius to fahrenheit

```
          celsius = float(input("user enter celsius degree:"))
           fahrenheit = (celsius * 1.8) + 32
           print("Fahrenheit degree is:", fahrenheit)
```

Output:

```
RESTART: C:/Users/Sandya-Samara/AppData/Local/Programs/Python/Python37-32/celciustfah.py
user enter celcius degree:45.05
Fahrenheit degree is: 113.09
```

17. This program gives you ideal weight based on your height.
    Ask user to enter his height in feet and inches. Convert height
    into centimeters. Subtract 100 from the centimeters. Finally print
    ideal weight i.e. the remaining centimeters as weight in kilograms.
    For example if someone's height is 168 centimeters, his ideal weight
    is 68 Kilograms. Assume one Inch is 2.5 centimeters. One foot is equal
    to 12 inches.

Sol:

```
hift = float(input(" User enter his Feet: "))
hinch = float(input(" User enter his  Inches: "))
hinch += hift * 12
hcm = round(hinch * 2.5, 1)
print("Height is : %d cm." % hcm)
c = hcm-100
print("Ideal weight is: %d kgs.",c)
```

```
Output:===== RESTART: C:/Users/Sandya-Samara/Desktop/sandhya programs/height.py =====
 User enter his Feet: 12
 User enter his inches Inches: 21
Height is : 419 cm.
```

```
Ideal weight is: %d kgs. 319.1
>>>
===== RESTART: C:/Users/Sandya-Samara/Desktop/sandhya programs/height.py =====
 User enter his Feet: 5.4
 User enter his  Inches: 12
Height is : 192 cm.
Ideal weight is: %d kgs. 92.0
```

Date: 31--08---2019

1.We add a Leap Day on February 29, almost every four years. The leap day is an extra, or intercalary day and we add it to the shortest month of the year, February. In the Gregorian calendar three criteria must be taken into account to identify leap years:-------1.The year can be evenly divided by 4, is a leap year, unless:-------------------------2.The year can be evenly divided by 100, it is NOT a leap year, unless:---------------3.The year is also evenly divisible by 400. Then it is a leap year.This means that in the Gregorian calendar, the years 2000 and 2400 are leap years, while 1800, 1900, 2100, 2200, 2300 and 2500 are NOT leap years.--------------Task You are given the year, and you have to write a function to check if the year is leap or not.--Note that you have to complete the function and remaining code is given as template.----Input

Format----Read y, the year that needs to be checked.Constraints-----Output FormatOutput is taken care of by the template. Your function must return a boolean value (True/False)---Sample Input 0------1990--

Sample Output

0------------False--

Answer: def is_leap(year):

Explanation 0

1990 is not a multiple of 4 hence it's not a leap year

REGULAR EXPRESSIONS:

To specify regular expressions, metacharacters are used. In the above example, `^` and `$` are metacharacters.

Regular expressions:

Sequence of characters that defines a search pattern.Here meta characters are used.

MetaCharacters

Metacharacters are characters that are interpreted in a special way by a RegEx engine. Here's a list of metacharacters:

[] . ^ $ * + ? {} () \ |

[] - Square brackets

Square brackets specifies a set of characters you wish to match.

Expression	String	Matched?
[abc]	**a**	**1 match**
	ac	**2 matches**
	Hey Jude	**No match**

	abc de ca	5 matches

Here, [abc] will match if the string you are trying to match contains any of the a, b or c.

You can also specify a range of characters using - inside square brackets.

- [a-e] is the same as [abcde].
- [1-4] is the same as [1234].
- [0-39] is the same as [01239].

You can complement (invert) the character set by using caret `^` symbol at the start of a square-bracket.

- **`[^abc]` means any character except `a` or `b` or `c`.**
- **`[^0-9]` means any non-digit character.**

. - Period

A period matches any single character (except newline '\n').

Expression	String	Matched?
..	a	**No match**
	ac	**1 match**
	acd	**1 match**

	acde	2 matches (contains 4 characters)

---

^ - Caret

The caret symbol ^ is used to check if a string starts with a certain character.

Expression	String	Matched?
^a	**a**	**1 match**
	abc	**1 match**
	bac	**No match**
^ab	**abc**	**1 match**

	acb	No match (starts with a but not followed by b)

$ - Dollar

The dollar symbol $ is used to check if a string ends with a certain character.

Expression	String	Matched?
a$	**a**	**1 match**
	formula	**1 match**
	cab	**No match**

*** - Star**

The star symbol * matches zero or more occurrences of the pattern left to it.

Expressio n	String	Matched?
ma*n	**mn**	**1 match**

	man	1 match
	maaan	1 match
	main	No match (a is not followed by n)
	woman	1 match

+ - Plus

The plus symbol + matches one or more occurrences of the pattern left to it.

Expression	String	Matched?

`ma+n`	`mn`	No match (no `a` character)
	`man`	1 match
	`maaan`	1 match
	`main`	No match (a is not followed by n)
	`woman`	1 match

`?` - Question Mark

The question mark symbol `?` matches zero or one occurrence of the pattern left to it.

Expression	String	Matched?

`ma?n`	`mn`	1 match
	`man`	1 match
	`maaan`	No match (more than one `a` character)
	`main`	No match (a is not followed by n)
	`woman`	1 match

`{}` - Braces

Consider this code: `{n,m}`. This means at least `n`, and at most `m` repetitions of the pattern left to it.

Expression	String	Matched?

a{2,3}	abc dat	No match
	abc daat	1 match (at daat)
	aabc daaat	2 matches (at aabc and daaat)
	aabc daaaat	2 matches (at aabc and daaaat)

Let's try one more example. This RegEx [0-9]{2, 4} matches at least 2 digits but not more than 4 digits

Expression	String	Matched?

[0-9]{2,4}	ab123csde	1 match (match at ab123csde)
	12 and 345673	2 matches (at 12 and 345673)
	1 and 2	No match

| - Alternation

Vertical bar `|` is used for alternation (`or` operator).

Expression	String	Matched?
`a|b`	**`cde`**	**No match**
	`ade`	**1 match (match at `ade`)**
	`acdbea`	**3 matches (at `acdbea`)**

Here, `a|b` match any string that contains either `a` or `b`