**Python Class 1:**

These are use for mathacmatical operation

var3=var1 symbol var2

var1=7

var2=3

+  (10)

- 4

\* 21

% 1

/ 2.5

\*\*

//

7//2=3

7.0//2=3.0

var3=var1 symbol var2

Asiignment ope:

Optimized arithmeetic operator

1)Third variable is not requited

2)One of the variable is conveted into output

var3=var1 symbol var2

var1 symbol =var2

var1+=var2(var1=var1+var2)

1)a+b+c+d

2)a\*b\*c\*d

3)(a+b)\*(c+d)

4)(a\*b)+(c\*d)

1)Direct method

2)Intermetidate method

a+b+c+d

t1=a+b

t2=c+d

sum=t1+t2

2)Find out sqare andd quabe given numbers

1)Arithmetic

2)Assignment

Bitwise operators:

THese opeator working on binay data

It is a combination of one and 0

10=

?

1010

pow(2,3) pow(2,2) pow(2,1) pow(2,0)

8  4   2    1

0  0  0   0

(8X0+4X0+2X0+1X0)

0 0 0 1

0  0 1 1

1 1 1 1

15

1)And 2)or 3)not

AND:

if both inputs are 1 we get 1 otherwise 0

bit1 bit2     bit1&bit2

-----        ------      -----

0         0          0

0    1       0

1    0      0

1    1     1

Print “Enter var1”

var1=input()

Print “Enter var2”

var2=input()

var3=var1&var2

Print “Result:”,var3

Python Class 2:

INDUSTRY STANDARD KEYWORDS:

Some of keywords which are frequently seen the engineers:

Client,project,fail,pass==>Client side

Error,warning,notes,complied,sucessfully==>Tool side

March,low,medium,high ==>Apprisal time

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

This is world famuous company in automobile industry

I has head office at Mumbai.

Xxx is a co-founder of this company.

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

User :xxxx

Exp :8

Id   :199372

**AND operator:**

var1=3

var2=4

0 1 1

1 0 0

------

0 0 0

------

**Not operator:**

Output =-1 -input

                                 =-1-0  =-1

                                 =-1 -1=-2

                                 =-1-5=-6

**Shift operator:**

→ Cannot accept negative values

Semi bit wise operator:

**1)left shift**

 Var1 (in binary) var2(Number of zeros )

**<<**

var1=5

var2=8

1 0 1       0 0 0 0 0 0 0 0

 512

2048

  25  6 0

3

011 00 = 12

8 4 2 1

8    4    2   1

1   1    0    0=8+4=12

**RIght shift: >>**

Var2 (Number of zeros) var1 (Binary )

Notes: Output should have equal number bits in var1

          will start from left hand side

00000 011

000  ===>0

00 1000

0010==2

**Concatenation operator:**

Var1 is followeb by var2

+

Var1    var2      +

----     -----      -----

Int      int        Addintion

Str     str        Concatenation

Str    int        concat

**Repetitive Operator:**

Var1 is repetited n number of times

\*

Var1   var2    \*

-----    -----    ----

Int      int     multication

Str     int     Repetitive

**Conditional Dependent operators:**

1)Eqaul and non qual

2)Relational

3)Logical

4)Membership

All these operators performs its action depenindg on the conditions

Depending on number of alternatives the condtions are classified into 3 types

1)NO alternate ---->  (if)

2)One alternate ←→ (if-else)

3)More than one alternate (Branch operator ..if-elif-elif----else)

**If:**

If (condition):

     Print “Message”

1)Equal and non equal operator:

        ==  (both are equal or not)

  If ( var1==var2):

     Print “Condition true”

**If-else:**

 If ( condition ):

    Print “Condition True”

Else:

   Print “Condition fail”

**Branching**:If more than one change is available we can go for branching

   1)Week of the day

   2)What is your favoraite color

   3)Month of the year

If  (condiiton1):

   Print “Message1”

elif(conditon2):

  Print “Message2”

…..

…..

….

Else

    Print “Please provide app value”

**Relational operators:**

1   4

It explains the relation bwtween two number

>  ( 3>1) T

<  (1<3)  T

>= (3>=2)  T

<= (2<=2)  T

Nested condition:

One condition is calling with in another condition it is called nested conditon

10  7   5

1. 10   7
2. 10 5
3. Max=10

Print “Eter var1”

var1=input()

Print “Enter var2”

var2=input()

Print “Enter var3”

var3=input()

If ( var1>var2):

   if(var1>var3):

       Prinnt “Max:”,var1

   Else:

      Print “Max:”,var3

elIf ( var2>var1):

   if(var2>var3):

       Prinnt “Max:”,var2

   Else:

      Print “Max:”,vaF

Else:

     Print “Max:”,var3

**Python Class 3:**

Find out max of given 4 numbers

Find out max of given 5 numbers

Logical operators:

These are working on conditions

1)and

2)or

And: If both conditions true output is true otherwise fail

If ( Cond1 and Cond2 ):

    Print “Max:”,var1

OR: if any one of the conditon is true output is true otherwise false

ARRAY::

List of elemements:

3 types of arrays in python:

1.List     2.Tuple     3.Dictonary

[   ]           ()                 {}

Updated  permanent   Relation

LIST:

list=[ e1,e2,e3………….en]

E1 ,e2,e3 are any data types

names=[ “Modi” ,”Sha”,”Advani”]

chars=[ ‘a’,’b’,’c’]

id=[1,2,4,5]

sal=[12.332,5.5,6.6]

all=[“xyz”,’M’,10,39.43,3+4j]

Accessing methods:

1.All elements: list

2.Particular element:

 List[index]

  Index is started with 0 and incr by 1

        0    first element of list

        1   Second eleent of list

3.Particular start to end of list

      List[n:]

List[2:]  =>Third element to end of list

4.Range of elements:  
  List[n1:n2]

    It prints n1 to n2-1 index

  Display 2 element to 5 element

5.Reverse accessing:

  Access last element:list[-1]

 2.Access reverse range:

   List[-3:-1]

Operators:

1.max(list)

2.min(list)

3.len(list)

6.Concatenate operator:

list3=list1+list2

7.Repetitive :

List \* n

Where n is integer

8.Updation

 list[index]=new\_value

LIST METHODS:

It is process of add and remove elements from/to list

1.ADD

  1.list.append(var)=>Add the element to the end of list

  2.list.extend(list2)=>list2 elements are appended to the list

  3.list.insert(index,value)=>specific value is inserted at specific position

2.Remove:

list.pop()==>It Removes the last element

list.pop(0) ==>first element remove

3.list.count(var)==>It gives how many times the var is repeated in the list

4.list.index(var) ==>It gives the index of the given var

5.list.sort()

tuple:

list=(e1,e2,e3………….en)

E1 ,e2,e3 are any data types

names=( “Modi” ,”Sha”,”Advani”)

chars=( ‘a’,’b’,’c’)

id=(1,2,4,5)

sal=(12.332,5.5,6.6)

all=(“xyz”,’M’,10,39.43,3+4j)

Accessing methods:

1.All elements: tuple

2.Particular element:

 List[index]

  Index is started with 0 and incr by 1

        0    first element of list

        1   Second eleent of list

3.Particular start to end of list

      List[n:]

**Python Class 4:**

Limitation of tuple and list:

Elements are accessing with index .

Index should be interger otherwise it fails.

It doen’t explain relation.

TO over come this limitation we can go for dictionary.

DICTIONARY:

It explains the relation bw variables:

Name : “xyz”

Age   :  29

Edu   : M.Tech

Id      : 192

Sal    : 103.43

First variable is key and second variable as value.

dict={‘key’:’value’}

Key and values are any data types.

**Python Class 5:**

Loops:

Loops are used to generates the numbers.

Seq 1 2 3 4 5 6

First number :1

Last number:6

Relation : +1

Above three parameter are called loop parameters.

Depending on the loop parameters the loops are classified into 2 types.

1)while loop

2)for loop

1. While loop

             3 Parameters are mentationed in three different lines.

    Inilization (1)

    While (condition)  ---(2)

        Print “Element:”,var

        Direction -------(3)

Loop controls:

1. Break :Break the loop operation

for\_loop:All parameter are mentioned in a single line

For var in range (1,11):

     Print “Element:”,var

Files

1.It stores the information

2.It used data trasmission

3.FIle handle:It is a surname of the file

4.f=open(“file\_name”,”mode”)

                                   wb=>write mode in binary format

                                   rb=>read mode in binary format

                                   ab=>append mode in binary format

5.File properties :

    f.name=>It givess file name

    f.mode=>It gives the file mode (wb,rb,ab)

    f.closed=>It checks files is closed or not

    f.softspace=>File is attached to printer or not

6.File write:

  f=open(“file\_name”,”wb”)

 f.write(“Welcome to Python files”)

f.close()

Actions:

1. Create a new file :file\_name
2. Message is stored into file

**Python Class 6:**

FUNCTIONS**:**

**Limitations big task:**

1.Development and debug time high

2.Time consuming process

3.It is not reusable concept

FUnction :

1. Function has small amount of code
2. Design and debug time less
3. It is time sharable process
4. adv:Reusability

Def my\_fun(argi,argu2,......):

   Print “Message”

   Operation

   Return

my\_fun(argu1,argu2…..)

2.Different types of functions

  1.Function without argument

  2.Function with argument

3. Function without argument:

    The function declaration don’t any argument

    Def my\_fun():

4. Function with argument:

   The function atleat one argument

   Def my\_fun(var):

Need for return:

f(a,b,c,d):a+b+c+d find out sum of four variable by using with two arguments.

f(1,2,3,4)=1+2+3+4

                T1    t2

                 3     7

                    10

Nested function:

One function is calling with in another function.

1. Two level nested functions

             1. f3(f2,f1)

    2. Multi level nested function

             1.f3(f2(f1))

Lambda FUnctions:

1.It optimized and alternate to the function

2. my\_fun=lambda var1,var2:operation

3. Print “Result:”,my\_fun(var1,var2)

Module:

1.Function declaration done in one file

2.import program name

3.Function is calling within another program

COMMANDLINE ARGUMENT:

1. Let us conside sum.py
2. TO run the program these following steps are required

        1.python sum.py

         2.Enter var1

        3.10

        4.Enter var2

        5.20

        6.Result :30

3.In above execution 2-5 (4 steps are required for 2 varaible)

4.If number of variables are increased it is a time consuming process

5.To overcome this we can go for command line argument:

6.COmmand line argument is a process of assign values to variables while executiong the program in a single line

7.python sum.py 10 20

8.Optimized code

9.import sys

**Python Class 7: Advanced**

1.Python with linux:

 1. LINUX:Commandbase based os

 2.Every task is done commands only

 3.Command substitution:

   It is a process of commands are called by python

4.Command substitution two things

  Import os - operating system

  Import sys - our system/laptop

  os.system(“linux\_command”)

5.The out of the program is same as the linux environment

2.Randomization:

 1.By using this concept we generates with following things

    1.Float = 0.728859315732

    2.Integer = 7

    3.int\_float = 7.43806629331

    4.Character

    5.Names

2.import random

**Python Class 8:**

Schedule: It is process of time management

1.sleep is a command it is used for time scaling

2.sleep n ;;n is integer see

   Sleep 1 ,1s

  Sleep   5,5s

  Sleep   60  ,1m

  Sleep 1m,

Sleep   5m,   5 mins

Sleep  30m ,Half an hour

Sleep  60m ,1 Hour

3. Display the name after after 1s ,2s,5s

4.Day format

1. 12 hour format
2. 0  1 2 3 4 5 6 7 8 9 10 11 0   1 2 3 4 5 6 7 8 9 10 11 0
3. AM                                    AM
4. If the application runs two’s a day we can go for 12 hour format

9. Auto\_Shell program:

   1.Automatically generates the shell script

   2.Sh is default scripting langue in linux environment

   3.It is called mother of scripting language

   4.Every shell has the extension .sh

   5.sh simple.sh running command for shell script

   6.echo : Is a keyword to display the message

   7.read : Is a keyword take the data from keyboard

   8.((var1 symbol var2)) for operation

   9.Command line argument

      $0:Program name

**Python Class 9:**

Regular expressions:

1.It is a heart of the linux environmet

2.This concept contains the different symbols

3.Each symbol has specific actions

4. Symbol         Actions

   ----------         ------------

      \*                       All

     ?                       Character

     [a]                      Starting character

     [ad]                    Starting character either a or d

     [^ad]                  Except starting character either a or d

    [a-d]                    Starting character a to d(a,b,c,d)

[^a-d]                      Except starting character a to d

[a-z]                        Starting character a to z

[0-9]                        Starting character letter 0 to 9

[A-Z]                       Starting character A to Z

[A-Za-z0-9]            Starting character A to Z ,a-z,0-9

^                            Starting  letter of line

$                             Ending letter of line

.                             one or more

\d                           Digit

\D                          Non Digit

\s                           Space

\S                         Non space

10. Find identification :

1. It is process of files identification
2. Let us consider the project contains 10000 number of file,among 5000 python files ,3000 perl ,2000 shell

          How much time is required to identification 5000 python files

            30 Mins

    3.<1minute

    4. To solve this types of problems easyly using find

    5. It is linux utility to used for identification of files

    6. Find Regular\_expression symbols

    7.All files : find \*

    8.Starting letter of file

         1.Find out all files starting with a :find [a]\*

         2.Find out all files starting with a or d :find [ad]\*

         3.Find out all files except starting with a or d :find [^ad]\*

         4.Find out all files starting with a to d :find [a-d]\*

         5.Find out all files except starting with a to d :find [^a-d]\*

         6.Find out all files starting with number :find [0-9]\*

   9.Endind letter

1. Find out all text files: find \*.txt

              1.Find out all python files: find \*.py

  10.Starting and ending letter:

   1. FInd out all python files starting b:find [b]\*.py

   2.Repeat step 8 with python files -(replace with .py)

  11.Number of characters:

  1. FInd out all files which has single character :find ?

12. Relation:

1. List out all files ,first letter are characters: find [a-z][a-z]\*.py

13.Position:

1. Lsit out all files ,first letter is a and third letter is b: find a?b\*.py

**11.File Content search :**

1.Let us consider the project contains 5000 python files ,Every file should have at least one “var” word

          How much time is required to count number of “var”  words in every python file

            2 hours

2. <1minute

3.Grep : Is a linux utility used for file content search

4.grep ‘letter,word’ file

5.Starting letter:

  List out all lines starting with s : grep ‘^s’ file

6.Ending letter:

  List out all lines ending  with t : grep ‘t$’ file

7.Starting and end letter:

  List out all lines starting letter a and ending letter t : grep ‘^a.\*t$’ file

8.More than one search : grep -e ‘letter’ -e ‘letter’ file

  List out all files start with a and c : grep -e ‘^a’ -e ‘^c’ file

9. Words:

    Grep ‘word’ file

10 . Combination of steps 5-8 along with step 9

11.Options: grep -letter ‘letter,character’

   1.-i :Not matching

   2.-v : Not matching

   3. -o : Only matched

   4.-c:count matched words

   5.-n :lines

   6.Before matching : -Bn ;where n is integer - displays n no. of lines present before matched line along with matched line

   7.After Matching    :-An ;where n is integer

   8.Before and after  :-Cn ; Before and after

12. Files:

1.if you want to search word in two files: grep ‘word’ file1 file2

2.Directory : grep ‘letter,word’ \*

FIle\_ALTER:

1. Sed : It is for file alter process
2. Search and replace:

           1.sed ‘s/old/new/n’ file ,where n is integer ; n represents 1st word in all sentences.

              n=1  first occurrence old  in every line is replaced with new

              n=g   all occurrences of old in all lines

            2.Particular line : sed ‘L1s/old/new/g’ file ,L1 integer

            3.Range of line : sed ‘L1,L2s/old/new/g’ file ,L1 ,L2 are line numbers

3. Change:

   1.It is used for toal line change

   2.sed ‘cx’ whre x ,int,float,char,string,tentese,sypcial symbol

       It is applicable for all lines

4.Delete

    Sed ‘d’ file ;delete all lines

    Sed ‘L1d’ file ;L1 is line number

    Sed ‘L1,l2d’ file ;L1,L2 are range of lines

5.Before line

    Sed  ‘ax’ file

6.Before and after

  Sed -e ‘ix’ -e ‘ay’ file

7.Enter text before every line

  Sed ‘s/^/Hyd  /g’ file

8.Add the text end of line

  Sed ‘s/$/ ;/g’ file

**Python Class 10:**

Oops:

1. Python is combined language
2. c,c++,shell,awk,smalltalk,modula 3
3. Class and objects are main things in oops
4. Class :

       1. It is a group of functions

  5.Object : It is instantiation of class(alias,calling as another name)

  6. Class my\_class ():

            ‘Descripting of class’

            Def  \_\_init\_\_(self,var1,var2….varn):

                        self.var1=var1

                        self.var2=var2

                         ……………...

                                     self.varn=var2

                   Def ope (self):

                              sum=self.var1+self.var2

                              Print “sum:”,sum

 7.class properties:

     \_\_name\_\_: Name of the class

    \_\_doc\_\_    :Description of class

    \_\_main\_\_  :main class

   \_\_bases\_\_:basics of class properties

Cygwn

python org

  cygwincy

**Python Class 11:**

1.\_\_init\_\_(self,var1,var2..)

2.self.var1=var1

3.By using this pass the arguments to the class function

4.Class varaible:

    1.It is mentationed after the class declaration

    2.Variable is accessing with by class name

5.Inheritance : It is process of child class is derived from parent class

Class child(parent):

AWK:

1. It is a used for file in a table format

2. By using we can access any columns,rows

3.awk ‘{print $n}’ file

   n=0  :It display the total file

   n=1  : It give the first column

4. Access more than column

   Awk ‘{print $n1,$n2}’ file

   N1,n2 are the coumn numbers

5.Relation :

    Awk ‘{print $n1”=>”$n2}’ file

6.Searching :

        Awk’/name/{print $n}’ file

         N:specifi column

7.NR:Number of Records (Horizontal lines)

   Awk NR%2 file (Even lines)

0%3(0)

1%3(1)

2%3(2)

**Python Class 12:**

File merging:  
It is a process of more than two files are arranged in different formats.

1)file by file vertically

   Cat file file1

2)Specific part

  Head -2 file1; head -3 file2

3) file by file horizontal

    Paste file1 file2

4) Conversion of total file into single line

  Paste -s file

5)Each line separated by

  Paste -s -d’symbol’ file

6) Repeat 4 ,5 steps for multiple files

7) Repetation:

    Paste file file file

8)Report generation

    Echo “Message”;paste command >Report

14) CUT:

    1. Cut the particular portion of line

     2.Parameter evaluation is done by using this method

    3. If the value is attached with syecial symbol we can go for cut

    4. Cut -c1 file ;first charcter of all line

    5. Particular start to end : cut -cn- file,where n is integer

    6. Range : cut -cn1-n2 : It cuts the range of characters

PROJECT:

1. Any project contains 3 parts
2. 1.Input part 2.Manipulation part 3.Output part
3. Input part:

           1. Input taken from keyboard

           2. File

           3.Static

            Output in may be in any format:

            1. Display

            2.Another file

1. Manipulation part:

           1. Input taken from keyboard

           2. File

           3. Static

           4. Operators,loops,conditions

           5. Report analasys

            Output in may be in any format:

            1. Display

            2.Another file

1. Output part:

           1. Input taken from keyboard

           2. File

           3. Static

           4. File merging

           5. Linux commands

            Output in may be in any format:

            1. Display

            2. Another file

            3. Report generation

1. Develop the file in the following format

          SNO NAME ID SAL COM LOC

           1.  Message concept

           2.  Variable concept

           3.  List,Tuple,Dictionary

           4.   Loops

           5.   Files

           6.   Functions

           7.   Command line argument

           8.   Linux command

           9.   File merging process

2. Manipulation part

   Let us consider  the file is in this format

   SNO NAME ID SAL COM LOC

   Calculate for the following things

1. FInd out total salary of the  file
2. FInd out total salary with respect to location
3. FInd out total salary with respect to company
4. Hike percentage

            Hyd 10 %

            BAN 15%

            CHENNAI   20%

             DELHI   25%

             NOIDA  30%

             MUMBAI 35

      5. Repeat step 2 for 4

3.Output :

       Let us consider  the file is in this format

   SNO NAME ID SAL COM LOC

1. Report generation

          1. Emps file name

         2.Sal

         3.Company

         4.Location file

         5.Emp\_ID,EMP\_SAL,EMP\_LOC,EMP\_COM

         6. EMP names with location

         7. EMP names with company

         8.LInux command

         9.REPORT FORMAT

           1. HYD   CHENNAI   BAN  DELHI NOIDA MUMBAI

                   1.Emp

                   2.Sal

                   3.IDs

       10. Repeat step 9 for vertical

   2.Report analasys

       1.ALL employees into list

       2.All salaries of the list

       3.Total salary