

PYTHON INTERNSHIP TRAINING PROGRAM



The industrial internship Training Program was organized by the Department of Basic sciences and humanities, TPCT'S college of Engineering, Osmanabad. It was arranged for all branches of F.Y.B.tech students. It was conducted by Fantasy Technologies in the period of 7th June to 2021 to 21/7/21 (45 Days) 90 Hrs. Founder of Fantasy technology are Mrs. Jyoti Tarange and trainer of this course were Mrs.Prarthana Bakshi. This industrial internship Training Program was held under the guidance of Honorable principal Dr.Vikramsingh Mane and coordinator of this program were Prof. Usha Wadne Head of BSH Dept.

Introduction to Python Programming

Python is a widely used high-level, general-purpose, interpreted, dynamic programming language. Its design philosophy emphasizes code readability, and its syntax allows programmers to express concepts in fewer lines of code than would be possible in languages such as C++ or Java. The language provides constructs intended to enable clear programs on both a small and large scale. Python supports multiple programming paradigms, including object-oriented, imperative and functional programming or procedural styles. It features a dynamic type system and automatic memory management and has a large and comprehensive standard library. Python interpreters are available for installation on many operating systems, allowing Python code execution on a wide variety of systems.

Introduction to Python Programming

- Why do we need Python?
- Program structure in Python

Execution steps

- Interactive Shell
- Executable or script files.
- User Interface or IDE

Memory management and Garbage collections

- Object creation and deletion
- Object properties

PYTHON INTERNSHIP TRAINING PROGRAM

Data Types and Operations

- Numbers
- Strings
- List
- Tuple
- Dictionary
- Other Core Types

Statements and Syntax in Python

- Assignments, Expressions and prints
- If tests and Syntax Rules
- While and For Loops
- Iterations and Comprehensions

File Operations

- Opening a file
- Using Files
- Other File tools

Functions in Python

- Function definition and call
- Function Scope
- Arguments
- Function Objects
- Anonymous Functions

Modules and Packages

- Module Creations and Usage
- Module Search Path
- Module Vs. Script
- Package Creation and Importing

PYTHON INTERNSHIP TRAINING PROGRAM

Classes in Python

- Classes and instances
- Classes method calls
- Inheritance and Compositions
- Static and Class Methods
- Bound and Unbound Methods
- Operator Overloading
- Polymorphism

Exception Handling in Python Programming

- Default Exception Handler
- Catching Exceptions
- Raise an exception
- User defined exception

Advanced Python Concepts

- Decorators
- Generators
- Iterators
- Co-routines

Standard Library Modules

Exercises

Roadmap with Python

PYTHON INTERNSHIP TRAINING PROGRAM

-: ADVANCED PYTHON TRAINING DETAILS :-

#MODULE1

Command Line arguments, Display Hooks

Standard data streams and Redirections

Osmodule , Sub-process module

Forking processes

Exec functions

Working with comprehensions

Working with Descriptors, Iterators, Generators and Decorators

The yield statement

range and x-range

Working with Context Managers

Wrapping Objects

Callback functions

Duck Typing, Monkey Patching in Python

Encapsulating Object Creation: Factory

MODULE 2

Introduction to Threads in python

thread module

threading module

Introduction to Pipes in python

anonymous pipes

named pipes, fifos

Introduction to Recursion

PYTHON INTERNSHIP TRAINING PROGRAM

Recursive functions in Python

Depth of Recursion

MODULE 3

CGI Programming

Introduction to WSGI

Introduction to PEP3333

Bottle Framework , Flask Framework

WebTest Framework

Create a basic Web Service in python

Working with Databases

Connecting with Cassandra DB, SQLite3, MySQL

Database Operations

MODULE 4

Network Programming

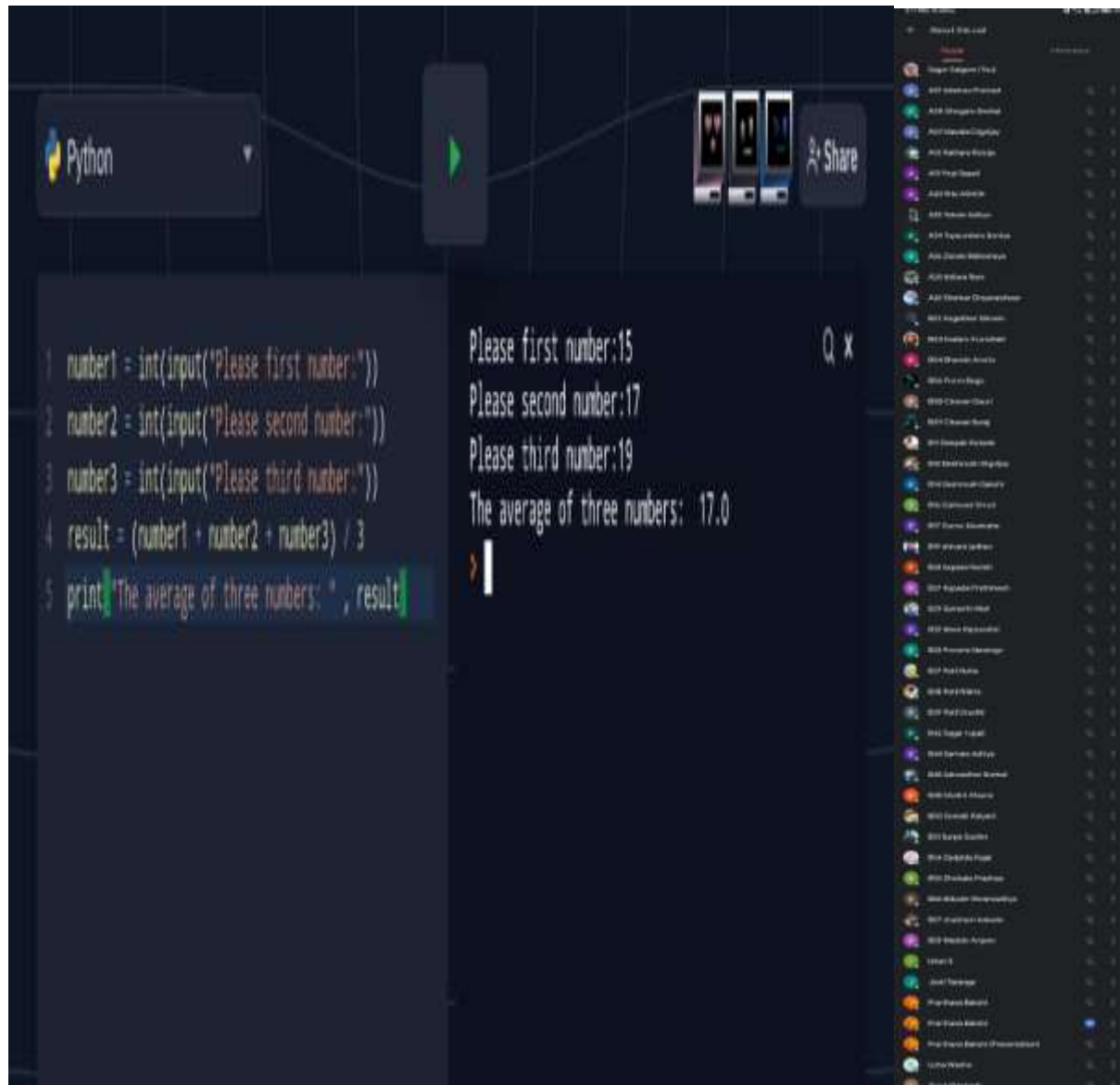
Working with XML Files

Developing GUIs

Working with SMTP

Integrating Python with other Languages

PYTHON INTERNSHIP TRAINING PROGRAM



The screenshot displays a Python IDE interface. On the left, a code editor contains the following Python code:

```
1 number1 = int(input("Please first number:"))
2 number2 = int(input("Please second number:"))
3 number3 = int(input("Please third number:"))
4 result = (number1 + number2 + number3) / 3
5 print("The average of three numbers: ", result)
```

On the right, a terminal window shows the execution output:

```
Please first number:15
Please second number:17
Please third number:19
The average of three numbers: 17.0
```

At the top of the IDE, there is a 'Python' dropdown menu, a green play button, and a 'Share' button. On the far right, a vertical list of participants is visible, each with a profile icon and a name.

Class conduction and Attendance

PYTHON INTERNSHIP TRAINING PROGRAM

1:47 PM | 9.0KB/s

replit.com/languages/python3

Python and more online from your browser

Sign up for the full experience

Python

```

1 while True:
2     print("Menu Drive Program")
3     print("1.Area Of Circle")
4     print("2.Area Of Rectangle")
5     print("3.Area Of Square")
6     print("4.Exit")
7     choice = int(input("Enter radius of circle: "))
8     if choice == 1:
9         radius=int(input("Enter radius of circle: "))
10        print("Area of circle: 3.14*radius*radius")
11    elif choice == 2:
12        length=int(input("Enter length of rectangle: "))
13        breadth=int(input("Enter breadth of rectangle: "))
14        print("Area of rectangle: ", length*breadth)
15    elif choice == 3:
16        side=int(input("Enter side of square: "))
17        print("Area: ", side*side)
18    elif choice == 4:
19        break
20    else:
21        print("Wrong Choice")
22

```

Menu Drive Program

- 1.Area Of Circle
- 2.Area Of Rectangle
- 3.Area Of Square
- 4.Exit

Enter radius of circle:3

Enter side of Square: 14

Area: 196

Menu Drive Program

- 1.Area Of Circle
- 2.Area Of Rectangle
- 3.Area Of Square
- 4.Exit

Enter radius of circle:4

About this file

Repl.it

Python

about this file

latest git commit information

1	Regular Expression Cheat	100%	1
2	SQL Database Tutorial	100%	1
3	Java Database Programming	100%	1
4	Java Database Tutorial	100%	1
5	Java Database Tutorial	100%	1
6	Java Database Tutorial	100%	1
7	Java Database Tutorial	100%	1
8	Java Database Tutorial	100%	1
9	Java Database Tutorial	100%	1
10	Java Database Tutorial	100%	1
11	Java Database Tutorial	100%	1
12	Java Database Tutorial	100%	1
13	Java Database Tutorial	100%	1
14	Java Database Tutorial	100%	1
15	Java Database Tutorial	100%	1
16	Java Database Tutorial	100%	1
17	Java Database Tutorial	100%	1
18	Java Database Tutorial	100%	1
19	Java Database Tutorial	100%	1
20	Java Database Tutorial	100%	1
21	Java Database Tutorial	100%	1
22	Java Database Tutorial	100%	1
23	Java Database Tutorial	100%	1
24	Java Database Tutorial	100%	1
25	Java Database Tutorial	100%	1
26	Java Database Tutorial	100%	1
27	Java Database Tutorial	100%	1
28	Java Database Tutorial	100%	1
29	Java Database Tutorial	100%	1
30	Java Database Tutorial	100%	1
31	Java Database Tutorial	100%	1
32	Java Database Tutorial	100%	1
33	Java Database Tutorial	100%	1
34	Java Database Tutorial	100%	1
35	Java Database Tutorial	100%	1
36	Java Database Tutorial	100%	1
37	Java Database Tutorial	100%	1
38	Java Database Tutorial	100%	1
39	Java Database Tutorial	100%	1
40	Java Database Tutorial	100%	1
41	Java Database Tutorial	100%	1
42	Java Database Tutorial	100%	1
43	Java Database Tutorial	100%	1
44	Java Database Tutorial	100%	1
45	Java Database Tutorial	100%	1
46	Java Database Tutorial	100%	1
47	Java Database Tutorial	100%	1
48	Java Database Tutorial	100%	1
49	Java Database Tutorial	100%	1
50	Java Database Tutorial	100%	1
51	Java Database Tutorial	100%	1
52	Java Database Tutorial	100%	1
53	Java Database Tutorial	100%	1
54	Java Database Tutorial	100%	1
55	Java Database Tutorial	100%	1
56	Java Database Tutorial	100%	1
57	Java Database Tutorial	100%	1
58	Java Database Tutorial	100%	1
59	Java Database Tutorial	100%	1
60	Java Database Tutorial	100%	1
61	Java Database Tutorial	100%	1
62	Java Database Tutorial	100%	1
63	Java Database Tutorial	100%	1
64	Java Database Tutorial	100%	1
65	Java Database Tutorial	100%	1
66	Java Database Tutorial	100%	1
67	Java Database Tutorial	100%	1
68	Java Database Tutorial	100%	1
69	Java Database Tutorial	100%	1
70	Java Database Tutorial	100%	1
71	Java Database Tutorial	100%	1
72	Java Database Tutorial	100%	1
73	Java Database Tutorial	100%	1
74	Java Database Tutorial	100%	1
75	Java Database Tutorial	100%	1
76	Java Database Tutorial	100%	1
77	Java Database Tutorial	100%	1
78	Java Database Tutorial	100%	1
79	Java Database Tutorial	100%	1
80	Java Database Tutorial	100%	1
81	Java Database Tutorial	100%	1
82	Java Database Tutorial	100%	1
83	Java Database Tutorial	100%	1
84	Java Database Tutorial	100%	1
85	Java Database Tutorial	100%	1
86	Java Database Tutorial	100%	1
87	Java Database Tutorial	100%	1
88	Java Database Tutorial	100%	1
89	Java Database Tutorial	100%	1
90	Java Database Tutorial	100%	1
91	Java Database Tutorial	100%	1
92	Java Database Tutorial	100%	1
93	Java Database Tutorial	100%	1
94	Java Database Tutorial	100%	1
95	Java Database Tutorial	100%	1
96	Java Database Tutorial	100%	1
97	Java Database Tutorial	100%	1
98	Java Database Tutorial	100%	1
99	Java Database Tutorial	100%	1
100	Java Database Tutorial	100%	1

Explore Multiplayer

The image displays two separate Python code snippets and their corresponding outputs in a dark-themed IDE.

Top Screenshot:

```

1 number1 = int(input("Please first number:"))
2 number2 = int(input("Please second number:"))
3 number3 = int(input("Please third number:"))
4 result = (number1 + number2 + number3) / 3
5 print("The average of three numbers is: ", result)

```

Output in the console:

```

Please first number:15
Please second number:17
Please third number:18
The average of three numbers: 17.0

```

Bottom Screenshot:

```

1 r = 10
2 area = 3.14 * r * r
3
4 print("Area of circle is: ", area)

```

Output in the console:

```

Area of circle is: 314.0

```

```
10,30,50,20,40]

10,50,30,40,20)

t (len (L1))
t (len(T1))

t ('max of L1', max (L1))
t ('max of T1', max (T1))

t ('min of L1', min(L1))
t ('min of T1', min(T1))

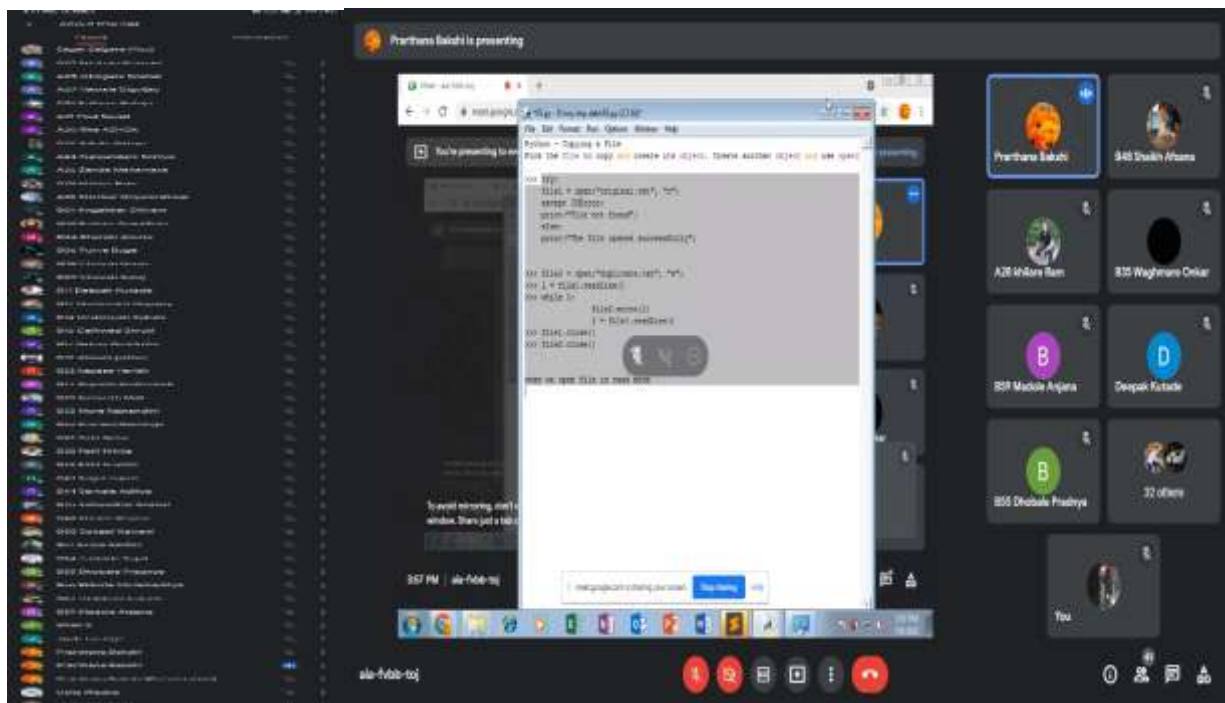
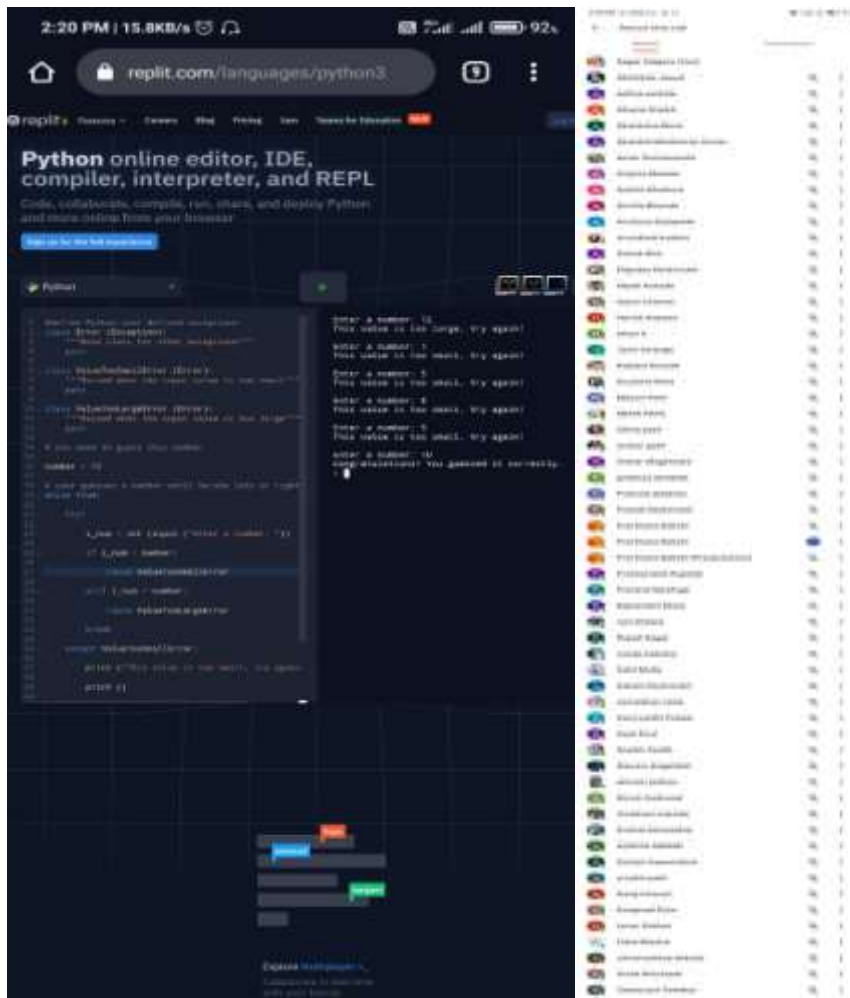
t ('sum of L1', sum(L1))
t ('sum of T1', sum(T1))
t ('L1 in sorted order',sorted (L1))
t ('T1 in sorted order',sorted (T1))
```


[illegible]

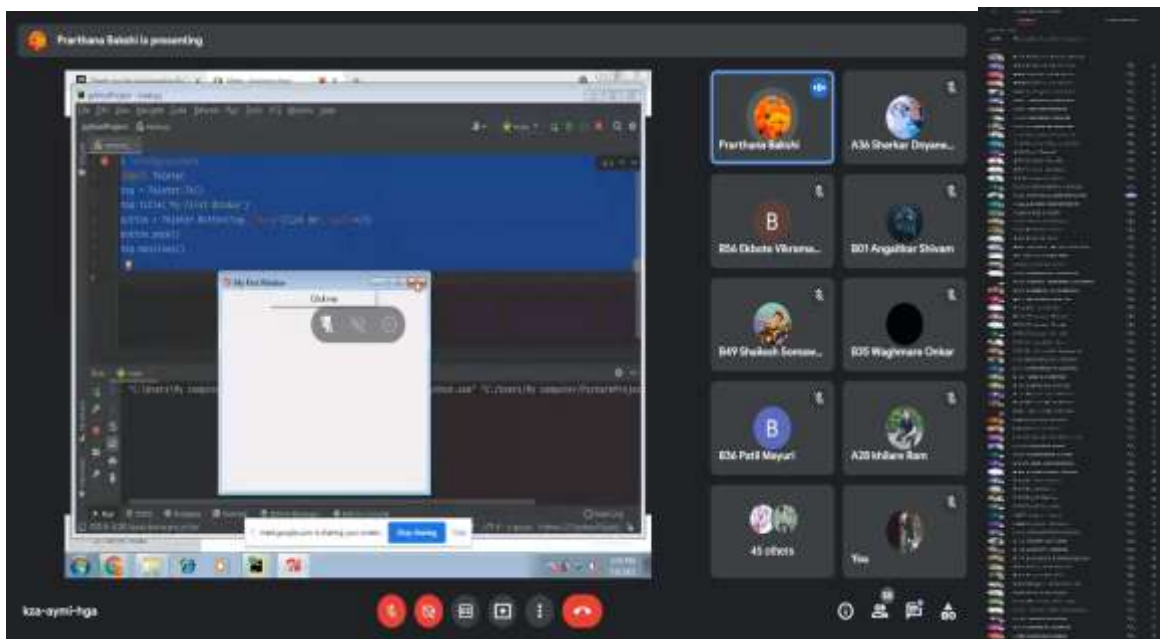
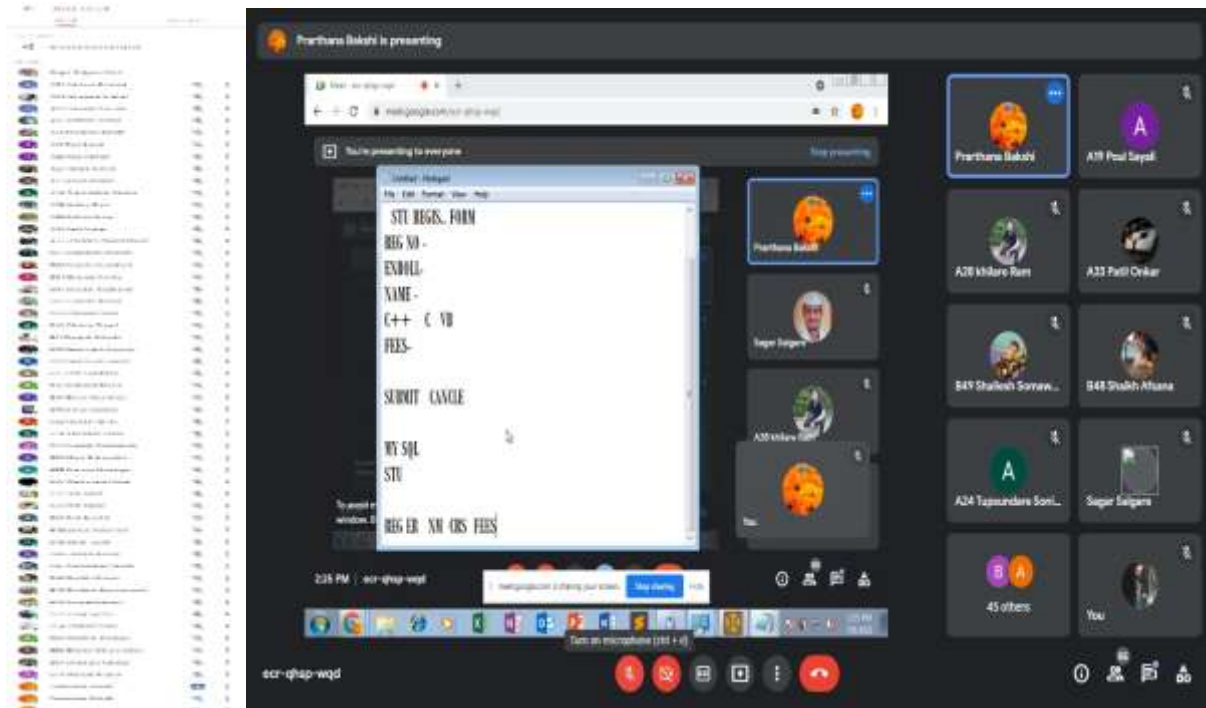
*

*

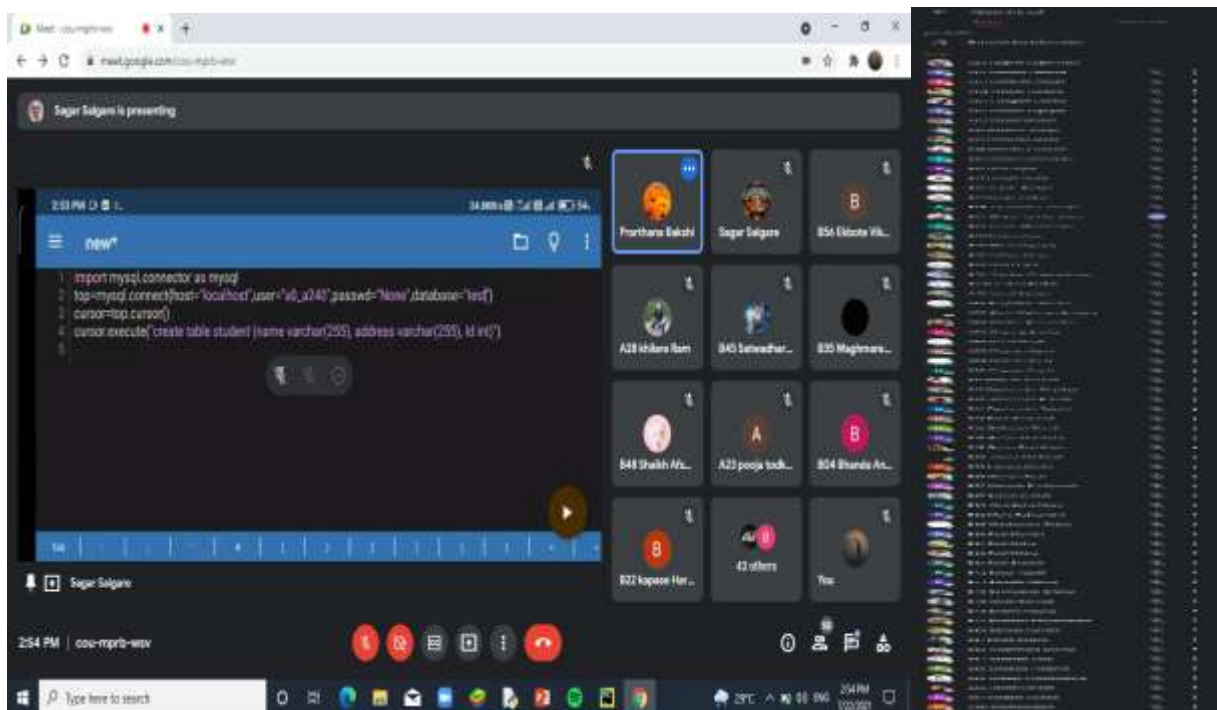
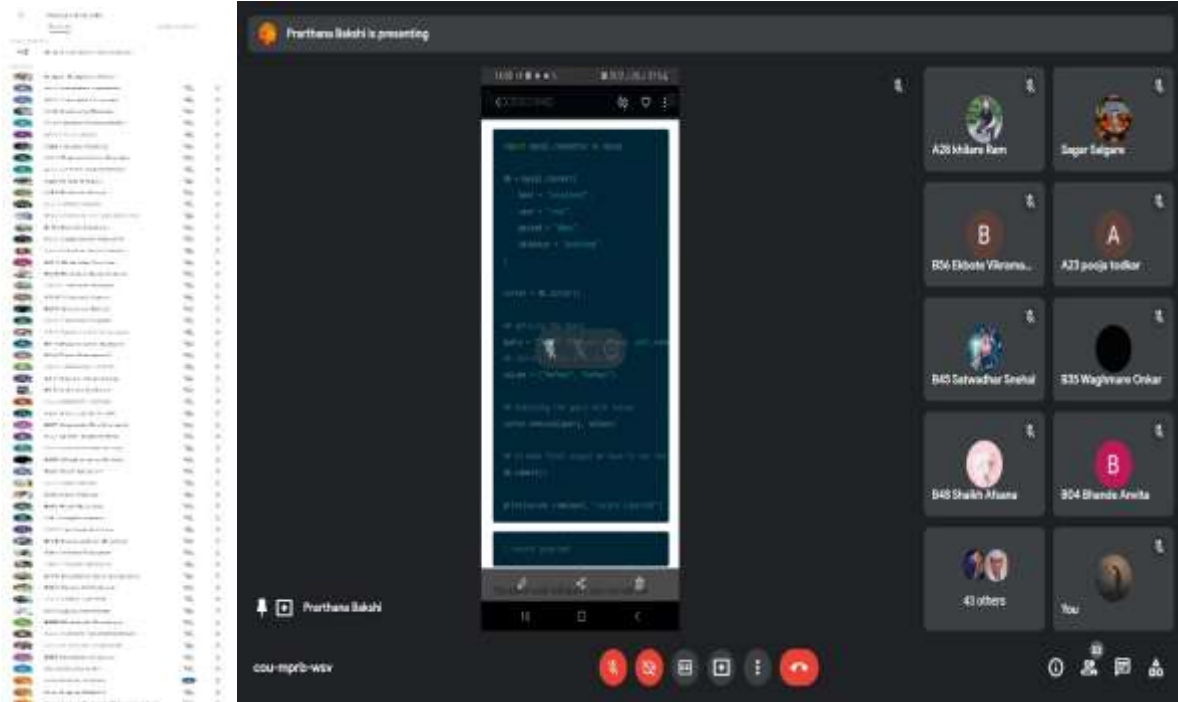
PYTHON INTERNSHIP TRAINING PROGRAM



PYTHON INTERNSHIP TRAINING PROGRAM



PYTHON INTERNSHIP TRAINING PROGRAM



PYTHON INTERNSHIP TRAINING PROGRAM

The top screenshot displays the Replit Python online editor interface. The main area shows a Python program that calculates the area of a circle. The code is as follows:

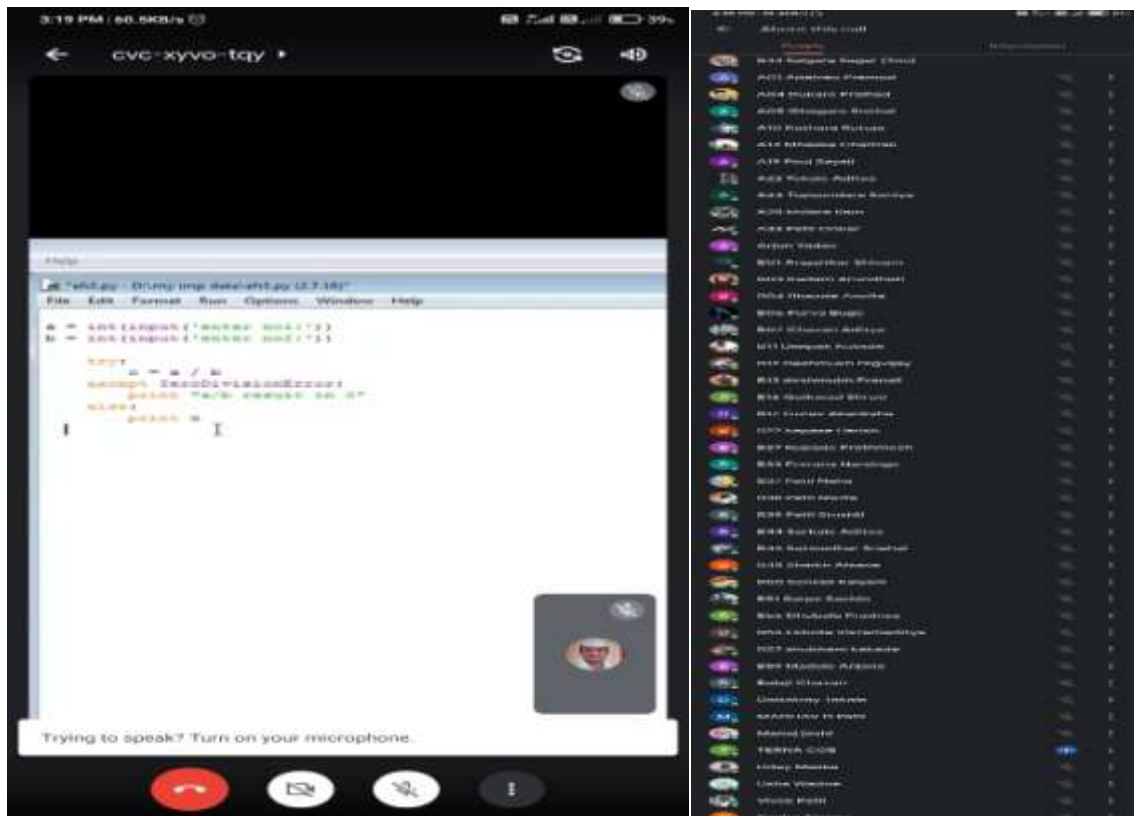
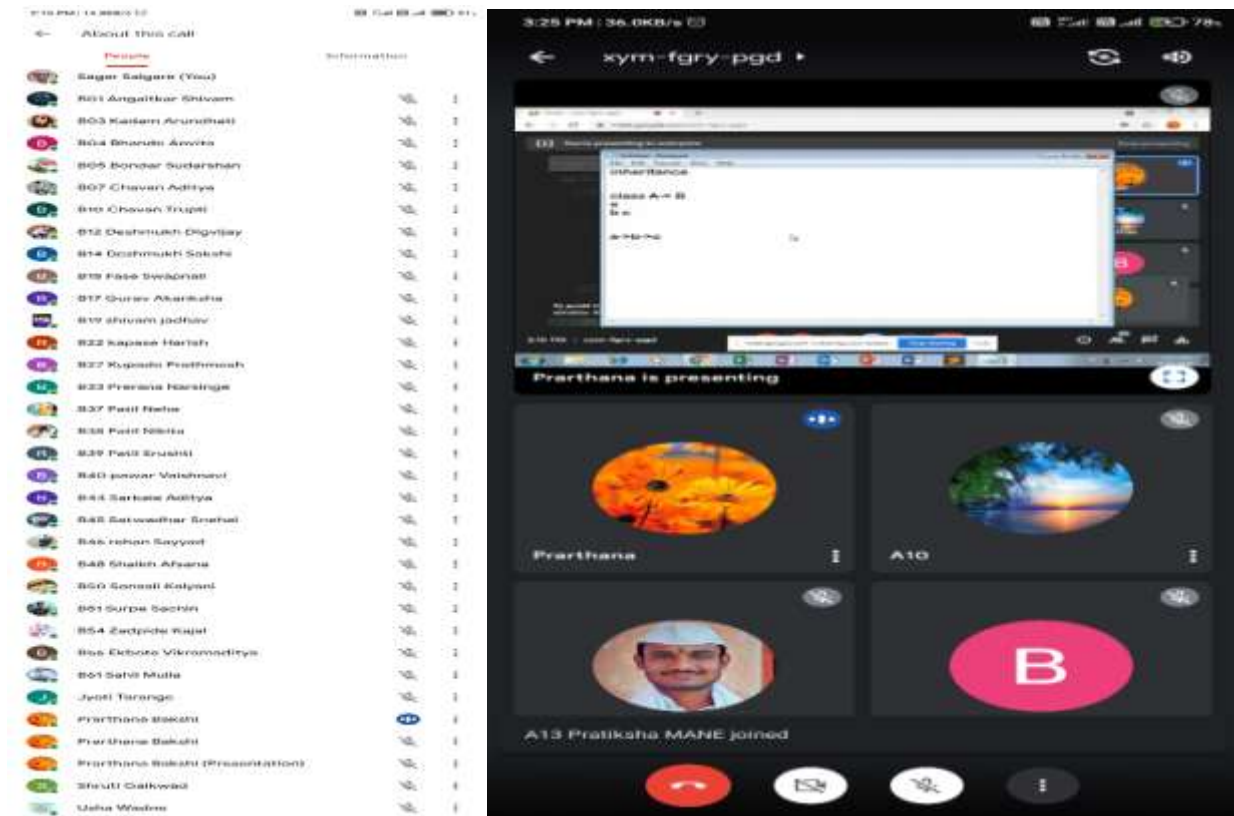
```

1 # Circle Area
2 # Circle Area Program
3 # Circle Area of Circle
4 # Circle Area of Circle
5 # Circle Area of Circle
6 # Circle Area of Circle
7 # Circle Area of Circle
8 # Circle Area of Circle
9 # Circle Area of Circle
10 # Circle Area of Circle
11 # Circle Area of Circle
12 # Circle Area of Circle
13 # Circle Area of Circle
14 # Circle Area of Circle
15 # Circle Area of Circle
16 # Circle Area of Circle
17 # Circle Area of Circle
18 # Circle Area of Circle
19 # Circle Area of Circle
20 # Circle Area of Circle
21 # Circle Area of Circle
22 # Circle Area of Circle
23 # Circle Area of Circle
24 # Circle Area of Circle
25 # Circle Area of Circle
26 # Circle Area of Circle
27 # Circle Area of Circle
28 # Circle Area of Circle
29 # Circle Area of Circle
30 # Circle Area of Circle
31 # Circle Area of Circle
32 # Circle Area of Circle
33 # Circle Area of Circle
34 # Circle Area of Circle
35 # Circle Area of Circle
36 # Circle Area of Circle
37 # Circle Area of Circle
38 # Circle Area of Circle
39 # Circle Area of Circle
40 # Circle Area of Circle
41 # Circle Area of Circle
42 # Circle Area of Circle
43 # Circle Area of Circle
44 # Circle Area of Circle
45 # Circle Area of Circle
46 # Circle Area of Circle
47 # Circle Area of Circle
48 # Circle Area of Circle
49 # Circle Area of Circle
50 # Circle Area of Circle
51 # Circle Area of Circle
52 # Circle Area of Circle
53 # Circle Area of Circle
54 # Circle Area of Circle
55 # Circle Area of Circle
56 # Circle Area of Circle
57 # Circle Area of Circle
58 # Circle Area of Circle
59 # Circle Area of Circle
60 # Circle Area of Circle
61 # Circle Area of Circle
62 # Circle Area of Circle
63 # Circle Area of Circle
64 # Circle Area of Circle
65 # Circle Area of Circle
66 # Circle Area of Circle
67 # Circle Area of Circle
68 # Circle Area of Circle
69 # Circle Area of Circle
70 # Circle Area of Circle
71 # Circle Area of Circle
72 # Circle Area of Circle
73 # Circle Area of Circle
74 # Circle Area of Circle
75 # Circle Area of Circle
76 # Circle Area of Circle
77 # Circle Area of Circle
78 # Circle Area of Circle
79 # Circle Area of Circle
80 # Circle Area of Circle
81 # Circle Area of Circle
82 # Circle Area of Circle
83 # Circle Area of Circle
84 # Circle Area of Circle
85 # Circle Area of Circle
86 # Circle Area of Circle
87 # Circle Area of Circle
88 # Circle Area of Circle
89 # Circle Area of Circle
90 # Circle Area of Circle
91 # Circle Area of Circle
92 # Circle Area of Circle
93 # Circle Area of Circle
94 # Circle Area of Circle
95 # Circle Area of Circle
96 # Circle Area of Circle
97 # Circle Area of Circle
98 # Circle Area of Circle
99 # Circle Area of Circle
100 # Circle Area of Circle

```

The bottom screenshot shows the same Replit Python online editor interface, but with a list of participants on the right side. The list includes names like Sagar Salgaonkar, Aniket Salgaonkar, and others. The code in the main area is the same as in the top screenshot.

PYTHON INTERNSHIP TRAINING PROGRAM



PYTHON INTERNSHIP TRAINING PROGRAM

3:12 PM | 2.6KB/s

replit.com/languages/python3

Python online editor, IDE, compiler, interpreter, and REPL

Code, collaborate, compile, run, share, and deploy Python and more online from your browser

Python

```

1 # Python code for Fibonacci series using a for loop
2 def fib(n):
3     a, b = 0, 1
4     for i in range(n):
5         print(a, end=" ")
6         a, b = b, a+b
7     print()
8
9 fib(10)
  
```

To return to your call, open Meet

3:23 PM | 1.7KB/s

replit.com/languages/python3

Python online editor, IDE, compiler, interpreter, and REPL

Code, collaborate, compile, run, share, and deploy Python and more online from your browser

Python

```

1 # Fibonacci series using a for loop
2 def fib(n):
3     a, b = 0, 1
4     for i in range(n):
5         print(a, end=" ")
6         a, b = b, a+b
7     print()
8
9 fib(10)
  
```

Python

3:28 PM | 1.4KB/s

About this call

People

Sagar Salgare (You)

801 Angartkar Shyam

803 Kadam Arundhat

804 Shinde Arvika

805 Mondar Sudarshan

807 Chavan Aditya

810 Chavan Trupti

812 Dandekar Digvijay

814 Deshmukh Sakshi

816 Fale Swapnali

817 Gurus Akanksha

819 shirani jathav

822 Kapase Harish

827 Rupase Prathmesh

833 Prasanna Harsinge

827 Paili Neha

838 Patil Nikita

839 Patil Shreshth

840 Pawar Vaishnavi

844 Sarkale Aditya

845 Sarwadhkar Snehal

846 rehan Sayyed

848 Shelke Afsana

850 Semaali Kalyani

851 Surpe Sachin

854 Zaidpate Rajat

856 Eklavik Vikramaditya

861 Sarvi Mula

Jyoti Tarange

Prarthana Bakshi

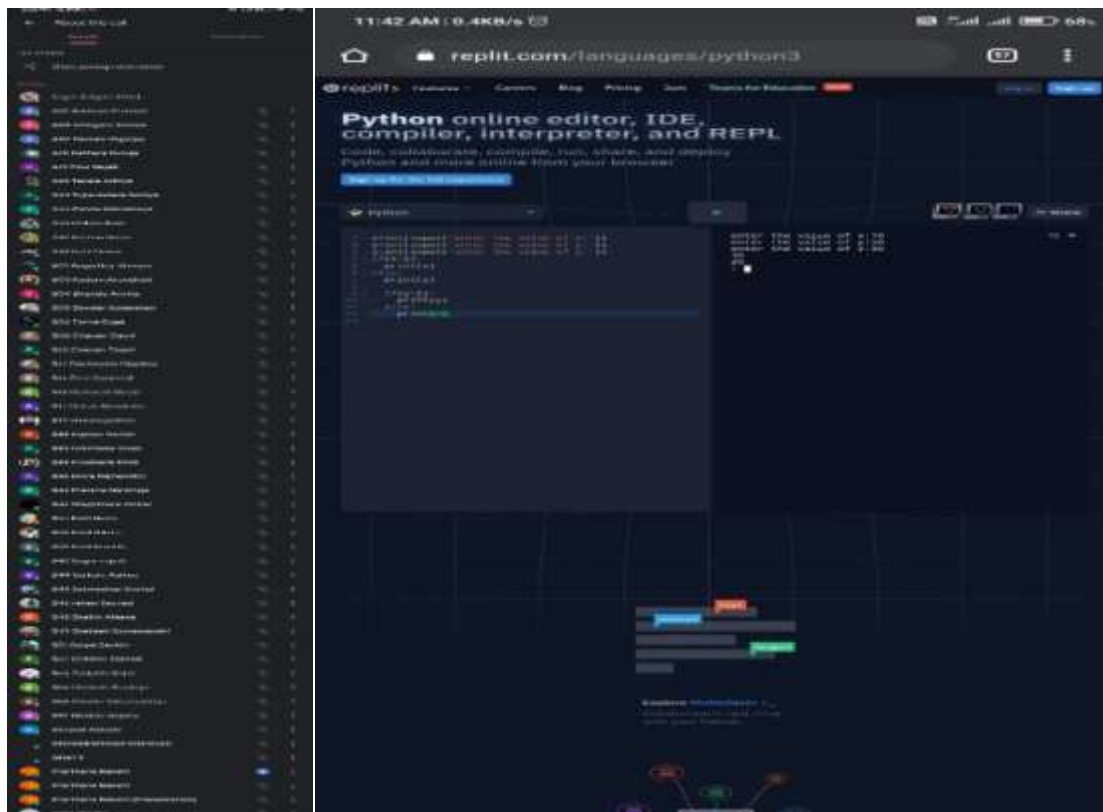
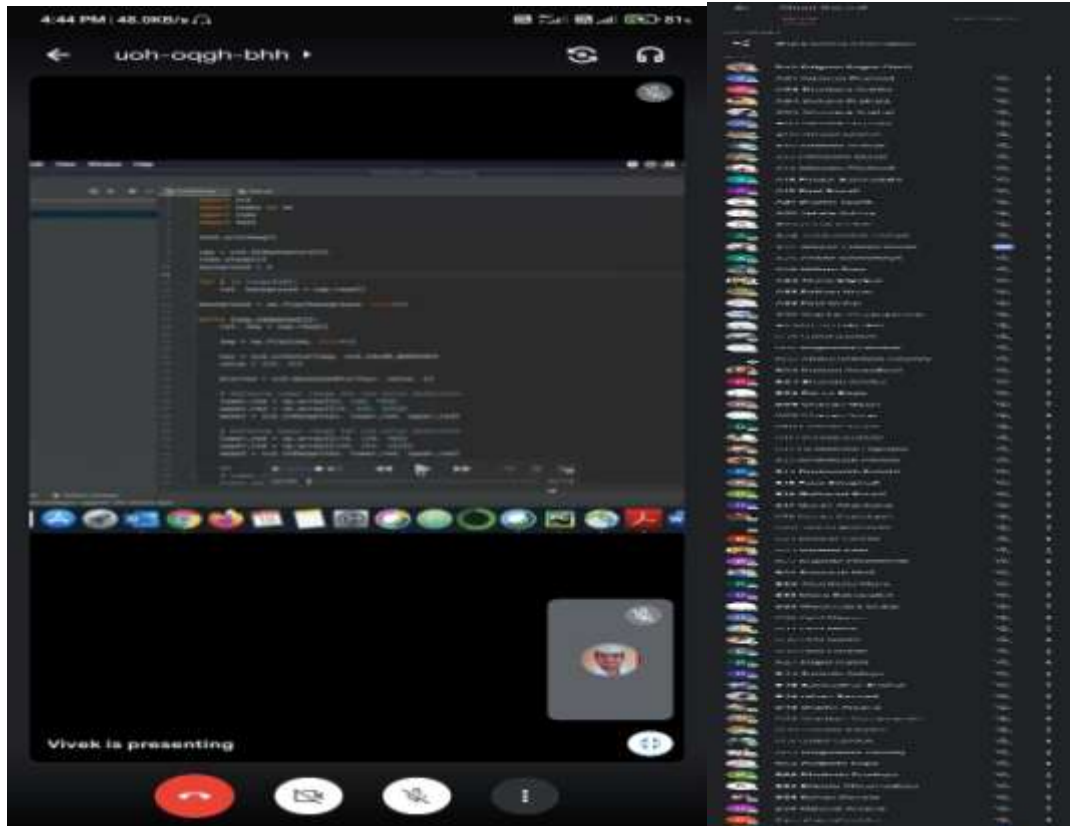
Prarthana Bakshi

Prarthana Bakshi (Presentation)

Shruti Gaihwad

Usha Wadne

PYTHON INTERNSHIP TRAINING PROGRAM



PYTHON INTERNSHIP TRAINING PROGRAM

Some Front End Windows

Slambook

☆ NAME:- Shradha Patil

☆ MOBILE NO:- 775953661

☆ DATE OF BIRTH:- 8 May 2002

☆ DREAM:- To Meet Virat Kohli

☆ HOBBIES:- Reading Books

☆ AIM:- To Become a Engineer

☆ NICK NAME:- Sweety

☆ FAVOURITE COLOUR:- Purple

☆ LIVES BIGGEST WISH:- I want land in London

☆ FAVOURITE SPOT:- Kadamath

☆ ENTER YOUR BIOGRAPHY:-

☆ MY SWEET SELFIE:-

UPLOAD IMAGE

☆ GENDER:- ☐ Male ☒ Female

RESET UPDATE DELETE DISPLAY

MY Life, MY Rules

Subject Marking System

—(Subject Details)—

☆ STUDENT NAME:- Umar Pathan

☆ ROLL NO:- 32

☆ CLASS:- B

☆ COLLEGE NAME:- TPCT's College Of Engi

☆ GENDER:- ☒ Male ☐ Female

☆ SELECT SUBJECT:- Engineering Mechanics

RESET UPDATE DELETE

STUDENT SIGN:- @umar

SUBMIT

PARENTS SIGN:- @ayyubkhan

Subject ID	Subject Name	Sessional Marks	End Exam Marks	Total Marks
C-101	English	30	70	100
C-102	Engineering Mathematics	20	80	100
C-103	Engineering Chemistry	20	80	100
C-104	Engineering Physics	20	80	100
C-105	Engineering Drawings	20	80	100
C-106	Engineering Mechanics	20	80	100

Thank You

CODING FOR THIS FRONT END WINDOW :-

PYTHON INTERNSHIP TRAINING PROGRAM

```
Python 3.8.5 Shell
File Edit Format Run Options Window Help

from tkinter import *
from tkinter.ttk import Combobox

root=Tk()
root.config(bg="pink")
root.title('Subject Marking System')

w1=Label(root, text="---Subject Details---",width=300,font= ("times 20 bold "),bg="silver",fg="black",bd=5,relief="raised",cursor="cross")
w1.pack(fill=X)

f2=Frame(root,height="700",highlightbackground="white",bg="pink",highlightthickness=15).place(x=80,y=60)

s1=Label(root, text="Subject ID",width="100",bg="pink",font="Algerian",).place(x=80,y=140)
s2=Label(root, text="Subject Name",width="150",bg="pink",font="Algerian",).place(x=250,y=140)
s3=Label(root, text="Sessional Marks",width="100",bg="pink",font="Algerian",).place(x=420,y=140)
s4=Label(root, text="End Exam Marks",width="100",bg="pink",font="Algerian",).place(x=590,y=140)
s5=Label(root, text="Total Marks",width="100",bg="pink",font="Algerian",).place(x=760,y=140)

t1=Table(root,width="700",height="150",bg="pink",font="Algerian",).place(x=80,y=180)

t1.insert(0,["C-101","English","20","30","50"])
t1.insert(1,["C-102","Engineering Mathematics","20","30","50"])
t1.insert(2,["C-103","Engineering Chemistry","20","30","50"])
t1.insert(3,["C-104","Engineering Physics","20","30","50"])
t1.insert(4,["C-105","Engineering Drawing","20","30","50"])
t1.insert(5,["C-106","Engineering Mechanics","20","30","50"])

s6=Label(root, text="Student Name:-",width="100",bg="pink",font="Algerian",).place(x=80,y=240)
s7=Label(root, text="Roll No:-",width="100",bg="pink",font="Algerian",).place(x=250,y=240)
s8=Label(root, text="Class:-",width="100",bg="pink",font="Algerian",).place(x=420,y=240)
s9=Label(root, text="College Name:-",width="150",bg="pink",font="Algerian",).place(x=590,y=240)
s10=Label(root, text="Student Sign:-",width="100",bg="pink",font="Algerian",).place(x=760,y=240)
s11=Label(root, text="Parents Sign:-",width="100",bg="pink",font="Algerian",).place(x=80,y=300)

e1=Entry(root).place(x=180,y=240)
e2=Entry(root).place(x=350,y=240)
e3=Entry(root).place(x=520,y=240)
e4=Entry(root).place(x=690,y=240)
e5=Entry(root).place(x=860,y=240)
e6=Entry(root).place(x=80,y=360)

c1=Combobox(root,values=["English","Engineering Mathematics","Engineering Physics","Engineering Chemistry","Engineering Mechanics","Engineering Drawing"])
c1.place(x=180,y=360)

b1=Button(root, text="SUBMIT",width="100",bg="pink",font="Algerian",).place(x=80,y=420)
b2=Button(root, text="INSERT",width="100",bg="pink",font="Algerian",).place(x=250,y=420)
b3=Button(root, text="UPDATE",width="100",bg="pink",font="Algerian",).place(x=420,y=420)
b4=Button(root, text="DELETE",width="100",bg="pink",font="Algerian",).place(x=590,y=420)
b5=Button(root, text="DISPLAY",width="100",bg="pink",font="Algerian",).place(x=760,y=420)
b6=Button(root, text="Thank You",width="100",bg="pink",font="Algerian",).place(x=80,y=480)
```

```
Python 3.8.5 Shell
File Edit Format Run Options Window Help

Label(root, text="Student ID:-",width="100",bg="pink",font="Algerian",).place(x=80,y=360)
var1 = IntVar()
Checkbutton(root, text="Male", bg="pink",variable =var1,pady=10).place(x=180,y=360)
var2 = IntVar()
Checkbutton(root, text="Female", bg="pink",variable =var2,pady=10).place(x=350,y=360)
s1=Label(root, text="Select Subject:-",width="150",bg="pink",font="Algerian",).place(x=520,y=360)

c1=Combobox(root,values=["English","Engineering Mathematics","Engineering Physics","Engineering Chemistry","Engineering Mechanics","Engineering Drawing"])
c1.place(x=620,y=360)

b1=Button(text="SUBMIT",width="100",activebackground="pink",bg="orange",font="Algerian",).place(x=760,y=420)
b2=Button(text="INSERT",width="100",activebackground="blue",bg="red",font="Algerian",).place(x=930,y=420)
b3=Button(text="UPDATE",width="100",activebackground="violet",bg="skyblue",font="Algerian",).place(x=1100,y=420)
b4=Button(text="DELETE",width="100",activebackground="red",bg="green",font="Algerian",).place(x=1270,y=420)
b5=Button(text="DISPLAY",width="100",activebackground="yellow",bg="purple",font="Algerian",).place(x=1440,y=420)
Label(root, text="Thank You",width="100",bg="silver",font="times 12 bold",bd=5,relief="sunken").pack(side= BOTTOM, fill = X)

root.mainloop()

import mysql.connector
def add_data():
    conn = mysql.connector.connect(
        host="localhost",
        user="root",
        password="mahadevi2008",
        database="wsl10")
    cursor = conn.cursor()
    query = "INSERT INTO users(name,username) VALUES (%s,%s)"
    values = ("shroo","sjs")
    cursor.execute(query, values)
    conn.commit()
    print("success, record inserted")

def update_data():
    conn = mysql.connector.connect(
        host="localhost",
        user="root",
        password="mahadevi2008",
        database="wsl10")
    cursor = conn.cursor()
```

PYTHON INTERNSHIP TRAINING PROGRAM

```
project insert.py - C:\Users\Def\AppData\Local\Programs\Python\Python38\project insert.py (3.8.5)
File Edit Format Run Options Window Help

cursor.execute(query, values)
conn.commit()
print(cursor, rowcount, "records inserted")

def update_data():
    conn = mysql.connector.connect(
        host="localhost",
        user="root",
        passwd="mahadev1234",
        database="world")
    cursor = conn.cursor()
    query = "UPDATE users SET name = 'aditya' WHERE id= 1"
    cursor.execute(query)
    conn.commit()

def remove_data():
    conn = mysql.connector.connect(
        host="localhost",
        user="root",
        passwd="mahadev1234",
        database="world")
    cursor = conn.cursor()
    query = "DELETE FROM users WHERE id= 1"

    cursor.execute(query)
    conn.commit()

def display_data():
    n = mysql.connector.connect(
        host="localhost",
        user="root",
        passwd="mahadev1234",
        database="world")
    cursor = conn.cursor()
    query = "SELECT * FROM users"

    cursor.execute(query)
    conn.commit()
```

```
project insert.py - C:\Users\Def\AppData\Local\Programs\Python\Python38\project insert.py (3.8.5)
File Edit Format Run Options Window Help

def remove_data():
    conn = mysql.connector.connect(
        host="localhost",
        user="root",
        passwd="mahadev1234",
        database="world")
    cursor = conn.cursor()
    query = "DELETE FROM users WHERE id= 1"

    cursor.execute(query)
    conn.commit()

def display_data():
    n = mysql.connector.connect(
        host="localhost",
        user="root",
        passwd="mahadev1234",
        database="world")
    cursor = conn.cursor()
    query = "SELECT * FROM users"

    cursor.execute(query)
    conn.commit()

Button(text = "INSERT", command = add_data)
Button(text = "UPDATE", command = update_data)
Button(text = "DELETE", command = remove_data)
Button(text = "DISPLAY", command = select_data)
```

PYTHON INTERNSHIP TRAINING PROGRAM

Employee Record System

Employee Details

Employee Name :

Employee ID :

Date of Joining :

Gender : ☒ Male ☐ Female

Address :


Salary :

Designation :

Department Name :

Registration Form

Home New Search <<--Course Information System-->> Help



Register ---:

Student Name

Class

School or College

Age

Contact

Gender ☐ Male ☐ Female

select course

☐ Agree all terms and conditions.

Course Information ---:

Course Id	Course Name	Duration	Fees
100	Python	45 Days	500rs
101	C++	50 Days	1000rs
102	Java	60 Days	1500rs
103	HTML	70 Days	2000rs
104	C	50 Days	2000rs

<https://www.python.org/doc/faq/faq1.html> <https://www.python.org/doc/faq/faq2.html> <https://www.python.org/doc/faq/faq3.html> <https://www.python.org/doc/faq/faq4.html> <https://www.python.org/doc/faq/faq5.html>

PYTHON INTERNSHIP TRAINING PROGRAM

OUTCOMES

MODULE 1

1. Exec functions :-

`exec()` function is used for the dynamic execution of Python program which can either be a string or object code. If it is a string, the string is parsed as a suite of Python statements which is then executed unless a syntax error occurs and if it is an object code, it is simply executed. We must be careful that the return statements may not be used outside of function definitions not even within the context of code passed to the `exec()` function. It doesn't return any value, hence returns `None`.

Syntax:

```
exec(object[, globals[, locals]])
```

It can take three parameters:

object: As already said this can be a string or object code

globals: This can be a dictionary and the parameter is optional

locals: This can be a mapping object and is also optional

Now let's see how this function works. In the following code, we have used an object code and executed it using `exec()` function. We have just taken the object parameter and omitted the other two fields.

Example:

```
prog = 'print("The sum of 5 and 10 is", (5+10))'  
exec(prog)
```

Output:

The sum of 5 and 10 is 15

MODULE 2

1. Thread module :-

PYTHON INTERNSHIP TRAINING PROGRAM

There are two modules which support the usage of threads in Python:

thread
and
threading

Please note: The thread module has been considered as "deprecated" for quite a long time. Users have been encouraged to use the threading module instead. So, in Python 3 the module "thread" is not available anymore. But that's not really true: It has been renamed to "_thread" for backwards incompatibilities in Python3.

The module "thread" treats a thread as a function, while the module "threading" is implemented in an object oriented way, i.e. every thread corresponds to an object.

Example for a Thread in Python:

```
from thread import start_new_thread
```

```
def heron(a):  
    """Calculates the square root of a"""  
    eps = 0.0000001  
    old = 1  
    new = 1  
    while True:  
        old,new = new, (new + a/new) / 2.0  
        print old, new  
        if abs(new - old) < eps:  
            break  
    return new
```

```
start_new_thread(heron,(99,))  
start_new_thread(heron,(999,))  
start_new_thread(heron,(1733,))
```

```
c = raw_input("Type something to quit.")
```

The raw_input() in the previous example is necessary, because otherwise all the threads would be exited, if the main program finishes. raw_input() waits until something has been typed in.

We expand the previous example with counters for the threads.

PYTHON INTERNSHIP TRAINING PROGRAM

```
from thread import start_new_thread
```

```
num_threads = 0
```

```
def heron(a):
```

```
    global num_threads
```

```
    num_threads += 1
```

```
    # code has been left out, see above
```

```
    num_threads -= 1
```

```
    return new
```

```
start_new_thread(heron,(99,))
```

```
start_new_thread(heron,(999,))
```

```
start_new_thread(heron,(1733,))
```

```
start_new_thread(heron,(17334,))
```

```
while num_threads > 0:
```

```
    pass
```

MODULE 3

1 Database Operations :-

Questions

How can I access databases from programs written in Python?

Objectives

Write short programs that execute SQL queries.

Trace the execution of a program that contains an SQL query.

Explain why most database applications are written in a general-purpose language rather than in SQL.

Example

```
import sqlite3
```

```
connection = sqlite3.connect("survey.db")
```

```
cursor = connection.cursor()
```


PYTHON INTERNSHIP TRAINING PROGRAM

```
cursor.execute("SELECT Site.lat, Site.long FROM Site;")
results = cursor.fetchall()
for r in results:
    print(r)
cursor.close()
connection.close()
```

Output

```
(-49.85, -128.57)
(-47.15, -126.72)
(-48.87, -123.4)
```

MODULE 4

1Developing GUIs :-

Python offers multiple options for developing GUI (Graphical User Interface). Out of all the GUI methods, tkinter is the most commonly used method. It is a standard Python interface to the Tk GUI toolkit shipped with Python. Python with tkinter is the fastest and easiest way to create the GUI applications. Creating a GUI using tkinter is an easy task.

To create a tkinter app:

Importing the module – tkinter

Create the main window (container)

Add any number of widgets to the main window

Apply the event Trigger on the widgets.

Importing tkinter is same as importing any other module in the Python code. Note that the name of the module in Python 2.x is 'Tkinter' and in Python 3.x it is 'tkinter'.

```
importtkinter
```

There are two main methods used which the user needs to remember while creating the Python application with GUI.

Tk(screenName=None, baseName=None, className='Tk', useTk=1): To create a main window, tkinter offers a method 'Tk(screenName=None, baseName=None, className='Tk', useTk=1)'. To change the name of the window, you can change the className to the desired one. The basic code used to create the main window of the application is:

PYTHON INTERNSHIP TRAINING PROGRAM

`m=tkinter.Tk()` where `m` is the name of the main window object

`mainloop()`: There is a method known by the name `mainloop()` is used when your application is ready to run. `mainloop()` is an infinite loop used to run the application, wait for an event to occur and process the event as long as the window is not closed.

```
m.mainloop()
```

```
importtkinter
```

```
m = tkinter.Tk()
```

```
'''
```

```
widgets are added here
```

```
'''
```

```
m.mainloop()
```

tkinter also offers access to the geometric configuration of the widgets which can organize the widgets in the parent windows. There are mainly three geometry manager classes class.

`pack()` method:It organizes the widgets in blocks before placing in the parent widget.

`grid()` method:It organizes the widgets in grid (table-like structure) before placing in the parent widget.

`place()` method:It organizes the widgets by placing them on specific positions directed by the programmer.

There are a number of widgets which you can put in your tkinter application. Some of the major widgets are explained below:

Button:To add a button in your application, this widget is used.

The general syntax is:

```
w=Button(master, option=value)
```

`master` is the parameter used to represent the parent window.

There are number of options which are used to change the format of the Buttons.

Number of options can be passed as parameters separated by commas. Some of them are listed below.

activebackground: to set the background color when button is under the cursor.

activeforeground: to set the foreground color when button is under the cursor.

bg: to set he normal background color.

command: to call a function.

font: to set the font on the button label.

image: to set the image on the button.

width: to set the width of the button.

height: to set the height of the button.

PYTHON INTERNSHIP TRAINING PROGRAM

```
importtkinter as tk  
r = tk.Tk()  
r.title('Counting Seconds')  
button = tk.Button(r, text='Stop', width=25, command=r.destroy)  
button.
```

PYTHON INTERNSHIP TRAINING PROGRAM



CERTIFICATE



PYTHON INTERNSHIP TRAINING PROGRAM

TPCT's
COLLEGE OF ENGINEERING, OSMANABAD
Dept of BSH
Class: F.Y.B.Tech Group: A&B Academic Year: 2020-21

Given list is of the students who have Successfully completed Python internship training program.

Sr.No.	Number	Name of Student	Branch
1	EN20221592	GHOGARE SNEHAL ASHOK	CIVIL
2	EN20171538	KATHARE RUTUJA JAYSHING	CIVIL
3	EN20178771	MANE PRATIKSHA MOHAN	CIVIL
4	EN20150874	POTDAR SAMRUDDHI SHISHIR	CIVIL
5	EN20155237	POUL SAYLI VIJAYKUMAR	CIVIL
6	EN20243218	RITE ASHOK RAMRAJE	CIVIL
7	EN20247388	SHAIKH TAUFIK TAJODDIN	CIVIL
8	EN20203372	TEKALE ADITYA DATTATRAY	CIVIL
9	EN20220548	TODKAR POOJA PANDURANG	CIVIL
10	EN20202056	TUPSUNDARE SONIYA BALAJI	CIVIL
11	EN20221185	KHILARE RAM SHAHU	MECH
12	EN20219309	PATHAN UMAR AYYUBKHAN	MECH
13	EN20178497	PATIL ONKAR NARSING	MECH
14	EN20203286	RASAL SAMADHAN BALASAHEB	MECH
15	EN20213067	SHERKAR DNYANESHWAR RAJENDRA	MECH
16	EN20202257	CHAURE OMKAR SAMBHAJI	CIVIL
17	EN20111256	ANGAITKAR SHIVAM AMAN	CSE
18	EN20227615	ARUNDHATI DINESH KADAM	CSE
19	EN20179406	BHANDE ANVITA BHARAT	CSE
20	EN20214840	BUGE PURVA SANTOSH	CSE
21	EN20201279	CHAVAN ADITYA DNYANDEO	CSE
22	EN20157405	CHVAN GAURI SHRIKANT	CSE
23	EN20198955	CHAVAN SURAJ BALASAHEB	CSE
24	EN20224334	CHAVAN TRUPTI LAXMAN	CSE
25	EN20113247	DEEPAK ARJUN KUPADE	CSE
26	EN20230377	DESHMUKH DIGVIJAY BHASKAR	CSE
27	EN20214248	DESHMUKH PRANALI PRASHANT	CSE
28	EN20142465	DESHMUKH SAKSHI MILIND	CSE
29	EN20217766	GAIKWAD SHRUTI CHANDRAKANT	CSE
30	EN20203400	GURAV AKANKSHA DATTATRAY	CSE
31	EN20198918	GURAV RUSHIKESH BHASKAR	CSE
32	EN20200755	JADHAV SHIVAM RAM	CSE
33	EN20197798	KAPSE HARISH VIVEKANAND	CSE

PYTHON INTERNSHIP TRAINING PROGRAM

34	EN20160860	KHICHADE VIVEK VIKRAM	CSE
35	EN20128280	KHOT KRUSHNA CHANDRAKANT	CSE
36	EN20167808	KUPADE PRATHMESH SUDHIR	CSE
37	EN20151936	MAGAR ROHAN ASHOK	CSE
38	EN20174720	MALI SAMARTH MADHUKAR	CSE
39	EN20215251	MORE AKANKSHA SURESH	CSE
40	EN20209760	MORE RAJNANDINI BALASAHEB	CSE
41	EN20222001	NARSINGHE PRERANA GOVIND	CSE
42	EN20203189	ONKAR VITTHAL WAGHMARE	CSE
43	EN20200121	PATIL MAYURI PRAVIN	CSE
44	EN20153508	PATIL NEHA PRAKASH	CSE
45	EN20151657	PATIL NIKITA PRAKASH	CSE
46	EN20222295	PAWAR VAISHNAVI UMAKANT	CSE
47	EN20224554	RANKHAMB DHANESH DHANAJI	CSE
48	EN20243015	SAGAR RUPALI ANIL	CSE
49	EN20171522	SALGARE SAGAR CHANDRAHAS	CSE
50	EN20247741	SARKALE ADITYA MAKRAND	CSE
51	EN20129015	SATWADHAR SNEHAL UDHAV	CSE
52	EN20148394	SAYYAD REHAN SHAMSHODDIN	CSE
53	EN20121151	SHAIKH AFSANA AKHALAQUE	CSE
54	EN20151152	SOMWANSI SHAILESH SHAHURAJ	CSE
55	EN20219003	SONSALI KALYANI AMOL	CSE
56	EN20150663	SURPE SACHIN CHANDRAKANT	CSE
57	EN20223557	SWAMI SONIYA SANJAY	CSE
58	EN20172980	WAGHMARE SHIVRAJ TUKARAM	CSE
59	EN20221889	ZADPIDE KAJAL DURGADAS	CSE
60	EN20225922	DHOBAL PRADNYA BIBHISHAN	ETC
61	EN20166738	EKBOTE VIKRAMADITYA NARHAR	ETC
62	EN20174524	KAKADE SHUBHAM AMBRESHWAR	ETC
63	EN20142396	MADOLE ANJANA MANIK	ETC
64	EN20119849	MULLA SAHIL SATTAR	CSE

Prof.U.K.Wadne

Dr.V.V.Mane

HOD BSH

Principal