COURSE OUTCOME 1

Date: 18/09/2023

1. Familiarizing Text Editor, IDE, Code Analysis Tools etc // Use any IDE like PyCharm, PyDev….

An integrated development environment (IDE) refers to a software application that offers computer programmers with extensive software development abilities. IDEs most often consist of a source code editor, build automation tools, and a debugger. Most modern IDEs have intelligent code completion. An IDE enables programmers to combine the different aspects of writing a computer program and increase programmer productivity by introducing features like editing source code, building executable, and debugging. IDEs are usually more feature-rich and include tools for debugging, building and deploying code. An IDE typically includes:

- A source code editor
- A compiler or interpreter
- An integrated debugger
- A graphical user interface (GUI)

A code editor is a text editor program designed specifically for editing source code. It typically includes features that help in code development, such as syntax highlighting, code completion, and debugging. The main difference between an IDE and a code editor is that an IDE has a graphical user interface (GUI) while a code editor does not. An IDE also has features such as code completion, syntax highlighting, and debugging, which are not found in a code editor. Code editors are generally simpler than IDEs, as they do not include many other IDE components. As such, code editors are typically used by experienced developers who prefer to configure their development environment manually. Some IDEs are given below:

1. IDLE

IDLE (Integrated Development and Learning Environment) is a default editor that accompanies Python. This IDE is suitable for beginner-level developers. The IDLE tool can be used on Mac OS, Windows, and Linux. The most notable features of IDLE include:

 Ability to search for multiple files

 Interactive interpreter with syntax highlighting, and error and i/o messages

 Smart indenting, along with basic text editor features

 A very capable debugger

 A great Python IDE for Windows

2. PyCharm

PyCharm is a widely used Python IDE created by JetBrains This IDE is suitable for professional developers and facilitates the development of large Python projects.

The most notable features of PyCharm include:

 Support for JavaScript, CSS, and TypeScript

 Smart code navigation

 Quick and safe code refactoring

 Support features like accessing databases directly from the IDE

3. Visual Studio Code

Visual Studio Code (VS Code) is an open-source (and free) IDE created by Microsoft. It finds great use in Python development. VS Code is lightweight and comes with powerful features that only some of the paid IDEs offer. The most notable features of Visual Studio Code include Git integration and Code debugging within the editor.

4. Sublime Text 3

Sublime Text is a very popular code editor. It supports many languages, including Python. It is highly customizable and also offers fast development speeds and reliability. The most notable features of Sublime Text 3 include:

 Syntax highlighting

 Custom user commands for using the IDE

 Efficient project directory management

 It supports additional packages for the web and scientific Python development

5. Atom

Atom is an open-source code editor by GitHub and supports Python development. Atom is similar to Sublime Text and provides almost the same features emphasis on speed and usability. The most notable features of Atom include:

 Support for a large number of plugins

 Smart autocompletion

 Supports custom commands for the user to interact with the editor

 Support for cross-platform development

6. Jupyter

Jupyter is widely used in the field of data science. It is easy to use, interactive and allowslive code sharing and visualization. The most notable features of Jupyter include:

 Supports for the numerical calculations and machine learning workflow

 Combine code, text, and images for greater user experience

 Intergeneration of data science libraries like NumPy, Pandas, and Matplotlib

7. Spyder

Spyder is an open-source IDE most commonly used for scientific development. Spydercomes with Anaconda distribution, which is popular for data science and machine learning. Themost notable features of Spyder include:

 Support for automatic code completion and splitting

 Supports plotting different types of charts and data manipulation

 Integration of data science libraries like NumPy, Pandas, and Matplotlib

Code Analysis Tools

Source code analysis tools, also known as Static Application Security Testing (SAST)Tools, can help analyse source code or compiled versions of code to help find security flaws. SAST tools can be added into IDE. Such tools can help to detect issues during software development. Static code analysis techniques are used to identify potential problems in code before it is deployed, allowing developers to make changes and improve the quality of the software. Three techniques include syntax analysis, data and control flow analysis, and security analysis.

SonarQube (Community Edition) is an open source static + dynamic code analysis platform developed by SonarSource for continuous inspection of code quality to perform fully automated code reviews / analysis to detect code smells, bugs, performance enhancements andsecurity vulnerabilities.

Date: 18/09/2023

2. Display future leap years from current year to final year entered by user.

Program

```
start=int(input("Enter the current year:"))
end=int(input("Enter the last year:"))
print("Upcoming leap years are:")
for year in range(start,end):
  if (year%4==0 and year%100!=0) or (year%400==0):
    print(year)
```

Output

```
Enter the current year:2020
Enter the last year:2050
Upcoming leap years are:
2024
2028
2032
2036
2040
2044
2048
```

Date: 18/09/2023

3. List comprehensions:

a. Generate positive list of numbers from a given list of integers

Program

```
listnum=[2,-4,6,-10,8,12]

positivenum = [i for i in listnum if i>= 0]

print(positivenum)
```

Output

```
[2, 6, 8, 12]
```

b. Square of N number

Program

```
n=int(input("Enter the limit:"))

enlist=[i**2 for i in range(n+1)]

enlist
```

Output

```
Enter the limit:5

[0, 1, 4, 9, 16, 25]
```

c. Form a list of vowels selected from a given word

Program

```
Word=(input("Enter a word:"))

v=[i for i in word if i in 'aeiouAEIOU']

v
```

Output

```
Enter a word:abhimanue

['i', 'a', 'a', 'e', 'u']
```

d. List ordinal value of each element of a word (Hint: use ord() to get ordinal values)

Program

```
print('hai')

c=(input("Enter a character:"))

o=[ord(i) for i in c]

o
```

Output

```
hai

Enter a character:hai

[104, 97, 105]
```

Date: 18/09/2023

4. Count the occurrences of each word in a line of text.

Program

```
str=input("Enter a sentence : ")

a=[]

a=str.split();

words=[a.count(i) for i in a]

print(dict(zip(a,words)))
```

Output

Enter a sentence: Python is a popular programming language. Python can be used on a server to create web applications.

{'Python': 2, 'is': 1, 'a': 2, 'popular': 1, 'programming': 1, 'language.': 1, 'can': 1, 'be': 1, 'used': 1, 'on': 1, 'server': 1, 'to': 1, 'create': 1, 'web': 1, 'applications.': 1}

Date:20/09/2023

5. Prompt the user for a list of integers. For all values greater than 100, store ‘over’ instead.

Program

```
n=int(input("Enter the number of elements:"))
list=[]
print("Enter the numbers")
for i in range (n):
  a=int(input())
  if(a<100):
    list.append(a)
  else:
    list.append('over')
    print(list)
```

Output

Enter the number of elements:4

Enter the numbers

10

30

80

120

[10, 30, 80, 'over']

Date: 20/09/2023

6. Store a list of first names. Count the occurrences of ‘a’ within the list.

Program

```
name=["abhimanue"]

for i in name:

  print("a occurs in", i, i.count('a'),"times")
```

Output

a occurs in abhimanue 2 times

Date: 20/09/2023

7. Enter 2 lists of integers. Check (a) Whether list are of same length (b) whether list sums to same value (c) whether any value occur in both

Program

```
n1=int(input("Enter the limit of first list:"))
list1=[]
print("Enter the elements:")
for i in range(n1):
  a1=int(input())
  list1.append(a1)
n2=int(input("Enter the limit of second list:"))
list2=[]
print("Enter the elements:")
for i in range(n2):
  a2=int(input())
  list2.append(a2)
print("The first list is",list1)
print("The second list is",list2)
if len(list1) == len(list2):
  print("The number of elements in list 1 and List 2 are EQUAL=",len(list1),"," ,len(list2))
```

```
else:
  print("The number of elements in list 1 and List 2 are NOT EQUAL=",len(list1),"," ,len(list2))
if sum(list1)==sum(list2):
  print("Sum of List 1 and List 2 are EQUAL",sum(list1), "and",sum(list2))
else:
  print("Sum of List 1 and List 2 are NOT EQUAL",sum(list1), "and",sum(list2))2
check = any(item in list1 for item in list2)
if check is True:
  print("Yes, Same value occur in both")
```

Output (case 1)

```
Enter the limit of first list:2
Enter the elements:
1
2
Enter the limit of second list:2
Enter the elements:
2
3
The first list is [1, 2]
The second list is [2, 3]
```

The number of elements in list 1 and List 2 are EQUAL= 2 , 2

Sum of List 1 and List 2 are NOT EQUAL 3 and 5

Yes, Same value occur in both

Output (case 2)

Enter the limit of first list:4

Enter the elements:

1

2

6

5

Enter the limit of second list:4

Enter the elements:

0

3

7

4

The first list is [1, 2, 6, 5]

The second list is [0, 3, 7, 4]

The number of elements in list 1 and List 2 are EQUAL= 4 , 4

Sum of List 1 and List 2 are EQUAL 14 and 14

Date: 20/09/2023

8. Get a string from an input string where all occurrences of first character replaced with ‘$’, except first character.

Program

```
string = input("Enter any string:")

char=string[0]

string=string.replace(char, '$')

strmod=char + string[1:]

print(strmod)
```

Output

```
Enter any string: abhimanue

abhimanu$
```

Date: 27/09/2023

9. Create a string from given string where first and last characters exchanged.

Program

```
name=" abhimanue "

print(name[-1] + name[1:-1] + name[0])
```

Output

ebhimanua

Date: 27/09/2023

10. Accept the radius from user and find area of circle.

Program

```
r=float(input("Enter the radius of the circle:"))

print("Area =",3.14*r*r)
```

Output

```
Enter the radius of the circle:10

Area = 314
```

Date: 27/09/2023

11. Find biggest of 3 numbers entered.

Program

```
a=int(input("Enter num 1="))
b=int(input("Enter num 2="))
c=int(input("Enter num 3="))
if(a>b and a>c):
  print(a,"is greater")
elif(b>c and a>a):
  print(b,"is greater")
elif(c>a and c>b):
  print(c,"is greater")
```

Output (case 1)

```
Enter num 1=55
Enter num 2=77
Enter num 3=234
234 is greater
```

Output (case 2)

```
Enter num 1=33
Enter num 2=56
Enter num 3=23
56 is greater
```

Output (case 3)

Enter num 1=78

Enter num 2=56

Enter num 3=67

78 is greater

Date: 27/09/2023

12. Accept a file name from user and print extension of that.

Program

```
fname=input("Enter a file name:")

extension=fname.split('.')

print("Extension=",extension[-1])
```

Output

```
Enter a file name:image. png

Extension= png
```

Date: 04/10/2023

13. Create a list of colors from comma-separated color names entered by user. Display first and last colors.

Program

```
n=int(input("Enter the no of colors ypou want to insert:"))
clist=[]
print("Enter the color:")
for i in range(n):
  color=input()
  clist.append(color)
print(clist)
print("The first and last colors are:")
print(clist[0],clist[-1])
```

Output

```
Enter the no of colors you want to insert:3
Enter the color:
 red
blue
black
['red', 'blue', 'black']
red black
```

Date: 04/10/2023

14. Accept an integer n and compute n+nn+nnn.

Program

```
n=int(input("Enter a value to n:"))

sum=n+(n*11)+(n*111)

print(n,'+',(n*11),'+',(n*111),'=',sum)
```

Output

```
Enter a value to n:3

3 + 33 + 333 = 369
```

Date: 04/10/2023

15. Print out all colors from color-list1 not contained in color-list2.

Program

```
n=int(input("Enter the no of colors you want to insert:"))
clist=[]
print("Enter the color:")
for i in range(n):
  color=input()
  clist.append(color)
print(clist)
print("The first and last colors are:")
print(clist[0],clist[-1])
```

Output

```
Enter the no of colors ypou want to insert:3
Enter the color:
red
blue
black
['red', 'blue', 'black']
red black
```

Date: 04/10/2023

16. Create a single string separated with space from two strings by swapping the character at position 1.

Program

```
s1=input("Enter the first string : ")

s2=input("Enter the second string : ")

s3=s2[0]+s1[1:]+" "+s1[0]+s2[1:]

print("Swapped string is :",s3)
```

Output

```
Enter the first string : Adarsh

Enter the second string : Reji

Swapped string is : Rdarsh Aeji
```

Date: 09/10/2023

17. Sort dictionary in ascending and descending order.

Program

```
d = {'Vinayak': 10, 'Hari': 20,'Basil': 30,'Amal': 40,'Subin': 50}


print("Ascending order : ",dict(sorted(d.items())))


print("Descending order : ",dict(sorted(d.items(), reverse=True)))
```

Output

Ascending order : {' abhimanue ': 40, 'Basil': 30, 'Hari': 20, 'Subin': 50, 'Vinayak': 10}

Descending order : {'Vinayak': 10, 'Subin': 50, 'Hari': 20, 'Basil': 30, ' abhimanue ': 40}

Date: 09/10/2023

18. Merge two dictionaries.

Program

```
d1 = {' abhimanue ': 10,'Hari': 20, 'Subin': 30}

d2 = {'Amal': 20,'Anjaly': 40, 'Meenakshy': 50}

print("Merged dictionaries :",d1|d2)
```

Output

Merged dictionaries : {' abhimanue ': 10, 'Hari': 20, 'Subin': 30, 'Amal': 20, 'Anjaly': 40, 'Meenakshy': 50}

Date: 09/10/2023

19. Find gcd of 2 numbers.

Program

```
x=int(input("Enter the first number : "))

y=int(input("Enter the second number : "))

print("GCD : ",math.gcd(x,y))
```

Output

```
Enter the first number : 18

Enter the second number : 81

GCD :  9
```

Date: 09/10/2023

20. From a list of integers, create a list removing even numbers.

Program

```
n=int(input("Enter the limit:"))
list=[]
print("Enter the numbers:")
for i in range(n):
  a=int(input())
  list.append(a)
print("The list with even numbers is:",list)
for i in list:
  if(i%2==0):
    list.remove(i)
print("The new list is:",list)
```

Output

Enter the limit:5

Enter the numbers:

2

3

4

5

6

The list with even numbers is: [2, 3, 4, 5, 6]

The new list is: [3, 5]

COURSE OUTCOME 2

Date: 11/10/2023

1. Program to find the factorial of a number

Program

```
num=int(input("Enter the number:"))
fact=1
if num<0:
  print("You entered a negative number")
else:
  for i in range(1,num+1):
    fact=fact*i;
  print("Factorial of",num, "is:",fact)
```

Output

```
Enter the number:5
Factorial of 5 is: 120
```

Date: 11/10/2023

2. Generate Fibonacci series of N terms

Program

```
t1=0
t2=1
next=t1+t2
n=int(input("Enter the limit:"))
print("The fibonacci series is:")
print(t1,"\n",t2)
for i in range(3,n+1):
  print(next)
  t1=t2;
  t2=next;
  next=t1+t2;
```

Output

```
Enter the limit:10
The fibonacci series is:
0
1
1
2
3
5
8
13
21
34
```

Date: 11/10/2023

3. Find the sum of all items in a list

Program

```
l=[] n=int(input("Enter the size of the list : "))

print("Enter elements : ")

for i in range(n):

  i=int(input())

  l.append(i)

print("Sum : ",sum(l))
```

Output

```
Enter the size of the list : 5

Enter elements :

2

3

5

1

0

Sum : 11
```

Date: 11/10/2023

4. Generate a list of four digit numbers in a given range with all their digits even and the number is a perfect square.

Program

```
res=[]

for i in range(1000,9999):

 if all(int(x)%2==0 for x in str(i)):

  if int(i**0.5)**2==i:

    res.append(i)

print("List of numbers : ",res)
```

Output

List of numbers : [4624, 6084, 6400, 8464]

Date: 16/10/2023

5. Display the given pyramid with step number accepted from user. Eg: N=4

1

2 4

3 6 9

4 8 12 16

Program

```
N = int(input("Enter the number of steps for the pyramid: "))
for i in range(1, N + 1):
  for j in range(1, i + 1):
    value = i * j
    print(value, end=" ")
  print()
```

Output

```
Enter the number of steps for the pyramid: 6
1
2 4
3 6 9
4 8 12 16
5 10 15 20 25
6 12 18 24 30 36
```

Date: 16/10/2023

6. Count the number of characters (character frequency) in a string.

Program

```
string = input("Enter a string: ")

char_count = {}

for char in string:

 char_count[char] = char_count.get(char, 0) + 1

for char, count in char_count.items():

 print(f"'{char}': {count}")
```

Output

```
Enter a string: aryan

'a': 2

'r': 1

'y': 1

'n': 1
```

Date: 16/10/2023

7. Add ‘ing’ at the end of a given string. If it already ends with ‘ing’, then add ‘ly’

Program

```
s=input("Enter any string:")
if s.endswith("ing"):
  print(s+"ly")
else:
  print(s+"ing")
```

Output (case 1)

```
Enter any string:play
Playing
```

Output (case 2)

```
Enter any string:playing
playingly
```

Date: 16/10/2023

8. Accept a list of words and return length of longest word.

Program

```
n = int(input("Enter the size of the list: "))

a = [input("Enter word: ") for _ in range(n)]

temp = max(a, key=len)

print("Word with max length is", temp, "Its length is", len(temp))
```

Output

```
Enter the size of the list: 3

Enter word: abhimanue

Enter word: basil

Enter word: Amal

Word with max length is abhimanue Its length is 9
```

Date: 18/10/2023

9. Construct following pattern using nested loop

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

Program

```
n = int(input("Enter number of rows: "))

for i in range(n):

 for j in range(i):

  print ('* ', end="")

 print('')

for i in range(n,0,-1):

 for j in range(i):

  print('* ', end="")

 print('')
```

Output

Enter number of rows: 5

*****

*** ***

*** * ***

*** * * ***

*** * * * ***

*** * * ***

*** * ***

*** ***

*****

Date: 18/10/2023

10. Generate all factors of a number.

Program

```
def facts(x):
  return [i for i in range(1, x + 1) if x % i == 0]
n = int(input("Enter a number: "))
factors = facts(n)
print(f"Factors of {n} are: {factors}")
```

Output

```
Enter a number: 564
Factors of 564 are: [1, 2, 3, 4, 6, 12, 47, 94, 141, 188, 282, 564]
```

Date: 18/10/2023

11. Write lambda functions to find area of square, rectangle and triangle.

Program

```
area1=lambda a: a*a

area2=lambda l,b: l*b

area3=lambda b,h: 0.5*(b*h)

s=int(input("Enter the side of square : "))

print("Area of square : ",area1(s))

l=int(input("Enter the length of rectangle : "))

b=int(input("Enter the breadth of rectangle : "))

print("Area of rectangle : ",area2(l,b))

b=int(input("Enter the base of triangle : "))

h=int(input("Enter the height of triangle : "))

print("Area of triangle : ",area3(b,h))
```

Output

```
Enter the side of square : 5

Area of square :  25

Enter the length of rectangle : 4

Enter the breadth of rectangle : 6

Area of rectangle :  24

Enter the base of triangle : 10

Enter the height of triangle : 4

Area of triangle :  20.0
```