# CLASS XII
# INFORMATICS PRACTICES
# PRACTICAL LIST

**1**       **Write a NumPy program to create a 3x3 matrix with values ranging from 2 to 10**

```python
import numpy as np
x =  np.arange(2, 11).reshape(3,3)
print(x)
```

**2**       **Write a NumPy program to generate six random integers between 25 and 55.**

```python
import numpy as np
x = np.random.randint(low=25, high=55, size=6)
print(x)
```

**3**       **Write a Pandas program to convert a Panda module Series to Python list and it's type**

```python
import pandas as pd
ds = pd.Series([2, 4, 6, 8, 10])
print("Pandas Series and type")
print(ds)
print(type(ds))
print("Convert Pandas Series to Python list")
print(ds.tolist())
print(type(ds.tolist()))
```

**4**       **Write a Pandas program to compare the elements of the two Pandas Series??**

```python
import pandas as pd
ds1 = pd.Series([2, 4, 6, 8, 10])
ds2 = pd.Series([1, 3, 5, 7, 10])
print("Series1:")
print(ds1)
print("Series2:")
print(ds2)
print("Compare the elements of the said Series:")
print("Equals:")
print(ds1 == ds2)
print("Greater than:")
print(ds1 > ds2)
print("Less than:")
print(ds1 < ds2)
```

**5**       **Write a Python program to convert a dictionary to a Pandas series. Sample Series:**
              **Dictionary:**
              **{'a': 100, 'b': 200, 'c': 300, 'd': 400, 'e': 800}**
              **Converted series:**
              **a 100**
              **b 200**

**c 300**
**d 400**
**e 800**
**dtype: int64**

```python
import pandas as pd
d1 = {'a': 100, 'b': 200, 'c':300, 'd':400, 'e':800}
print(,Dictionary:")
print(d1)
s1 = pd.Series(d1)
print("Converted series:")
print(s1)
```

**6        Write a Pandas program to add, subtract, multiple and divide two Pandas Series**
```python
import pandas as pd
ds1 = pd.Series([2, 4, 6, 8, 10])
ds2 = pd.Series([1, 3, 5, 7, 9])
ds = ds1 + ds2
print("Add two Series:")
print(ds)
print("Subtract two Series:")
ds = ds1 - ds2
print(ds)
print("Multiply two Series:")
ds = ds1 * ds2
print(ds)
print("Divide Series1 by Series2:")
ds = ds1 / ds2
print(ds)
```

**7        Write a program to sort the element of Series S1 into S2**
```python
import pandas as pd
s1 = pd.Series(['100', '200', 'python', '300.12', '400'])
print("Series before sorting:")
print(s)
s2 = pd.Series(s1).sort_values()
print("Series After sorting:")
print(s2)
```

**8        Write a NumPy program to reverse an array Ar**

```python
import numpy as np
L=[12,45,13,45,56,38]
Ar= np.array ( L)
print ( "Original Array:")
print ( Ar)
print ( "Reverse array:")
Ar = Ar[::-1]
print ( Ar)
```

9      **Write a NumPy program to create a 8x8 matrix and fill it with a checkerboard pattern.**

         **Checkerboard pattern:**
```
[[0 1 0 1 0 1 0 1]
 [1 0 1 0 1 0 1 0]
 [0 1 0 1 0 1 0 1]
 [1 0 1 0 1 0 1 0]
 [0 1 0 1 0 1 0 1]
 [1 0 1 0 1 0 1 0]
 [0 1 0 1 0 1 0 1]
 [1 0 1 0 1 0 1 0]]
```

```python
import numpy as np
x = np.ones((3,3))
print("Checkerboard pattern:")
x = np.zeros((8,8),dtype=int)
x[1::2,::2] = 1
x[::2,1::2] = 1
print(x)
```

10      **Write a NumPy program to append values to the end of an array. Expected Output:**
**Original array:**
**[10, 20, 30]**
**After append values to the end of the array:**
**[10 20 30 40 50 60 70 80 90]**

```python
import numpy as np
x = [10, 20, 30]
print("Original array:")
print(x)
x = np.append(x, [40, 50, 60,70, 80, 90])
print("After append values to the end of the array:")
print(x)
```

11      **Write a NumPy program to test whether each element of a 1-D array is also present in a second array**
```python
import numpy as np
ar1 = np.array([0, 12, 22, 40, 67])
print("Array1: ",ar1)
ar2 = [0, 22]
print("Array2: ",ar2)
print("Compare each element of array1 and array2")
print(np.in1d(array1, array2))
```

12      **Write a NumPy program to find the number of elements of an array, length of one array element in bytes and total bytes consumed by the elements**
```python
import numpy as np
x = np.array([1,2,3], dtype=np.float64)
print("Size of the array: ", x.size)
print("Length of one array element in bytes: ", x.itemsize)
print("Total bytes consumed by the elements of the array: ",
x.nbytes)
```

**13**      **Write a Pandas program to select the rows where the height is not known, i.e. is NaN.**
         **'name': ['Asha', 'Radha', 'Kamal', 'Divy', 'Anjali'],**
         **'height': [ 5.5, 5, np.nan, 5.9, np.nan],**
         **'age': [11, 23, 22, 33, 22]**

```python
import pandas as pd
import numpy as np
pers_data  = {'name': ['Asha', 'Radha', 'Kamal', 'Divy',
'Anjali'],
          'height': [ 5.5, 5, np.nan, 5.9, np.nan],
          'age': [11, 23, 22, 33, 22]}

labels = ['a', 'b', 'c', 'd', 'e']

df = pd.DataFrame(pers_data , index=labels)
print("Persons whose height not known:")
print(df[df['height'].isnull()])
```

**14**      **Write a Pandas program to select the name of persons whose height is between 5 to 5.5 (both values inclusive)**
         **'name': ['Asha', 'Radha', 'Kamal', 'Divy', 'Anjali'],**
         **'height': [ 5.5, 5, np.nan, 5.9, np.nan],**
         **'age': [11, 23, 22, 33, 22]**

```python
import pandas as pd
import numpy as np
pers_data  = {'name': ['Asha', 'Radha', 'Kamal', 'Divy',
'Anjali'],
          'height': [ 5.5, 5, np.nan, 5.9, np.nan],
          'age': [11, 23, 22, 33, 22]}
labels = ['a', 'b', 'c', 'd', 'e']
df = pd.DataFrame(pers_data , index=labels)
print("Persons whose height is between 5 and 5.5")
print(df[(df['height']>= 5 )& (df['height']<= 5.5)])
```

**15**      **Write a panda program to read marks detail of Manasvi and Calculate sum of all marks**

```python
import pandas as pd
import numpy as np
data  = {'Manasvi': ['Physics', 'Chemistry', 'English',
'Maths', 'Computer Sc'],
          'marks': [ 89,99,97,99,98],}

df = pd.DataFrame(data )
print("Sum of Marks:")
print(df['marks'].sum())
```

**16**      **Write a Pandas program to sort the data frame first by 'Designation' in Ascending order, then by 'Name' in Descending order.**

```python
import pandas as pd
```

```
data1 = {'Name':['Akshat', 'Sameer', 'Pihu', 'Neha'] ,
'Age':[28,34,29,42], 'Designation':['Accountant' , 'Clerk',
'Clerk', 'Manager']}
df1 = pd.DataFrame(data1)
print (df1.sort_values(by=['Designation','Name'],
ascending=[True,False]))
```

**17**    **Draw the histogram based on the Production of Wheatin different Years**
**Year:2000,2002,2004,2006,2008,2010,2012,2014,2016,2018**
**Production':4,6,7,15,24,2,19,5,16,4**

```
import pandas as pd
import matplotlib.pyplot as plt
data={'Year':[2000,2002,2004,2006,2008,2010,2012,2014,2016,2018],\
      'Production':[4,6,7,15,24,2,19,5,16,4]}
d=pd.DataFrame(data)
print(d)
x=d.hist(column='Production',bins=5,grid=True)
plt.show(x)
```

**18**    **Write a program to create dataframe for 3 student including name and roll numbers.**
**and add new columns for 5 subjects and 1 column to calculate percentage. It should**
**include random numbers in marks of all subjects**

```
import pandas as pd, numpy as np, random
D={'Roll':[1,2,3],'Name':['Sangeeta','Shanti','Swati']}
P=[]
C=[]
M=[]
E=[]
H=[]
SD=pd.DataFrame(D)
for i in range(3):
P.append(random.randint(1,101))
C.append(random.randint(1,101))
M.append(random.randint(1,101))
E.append(random.randint(1,101))
H.append(random.randint(1,101))
SD['Phy']=P
SD['Chem']=C
SD['Maths']=M
SD['Eng']=E
SD['Hin']=H
SD['Total']=SD.Phy+SD.Chem+SD.Maths+SD.Eng+SD.Hin
SD['Per']=SD.Total/5
print(SD)
```

**19**    **The table shows passenger car fuel rates in miles per gallon for several years. Make a**
**LINE GRAPH of the data. During which 2-year period did the fuel rate decrease?**

**YEAR: 2000   2002  2004   2006**
**RATE: 21.0 20.7 21.2 21.6**

```
import matplotlib.pyplot as p
Yr=[2000,2002,2004,2006]
rate=[21.0,20.7,21.2,21.6]
p.plot(Yr,rate)
p.show()
```

**20**     **The number of bed-sheets manufactured by a factory during five consecutive weeks is given below.**

| Week | First | Second | Third | Fourth | Fifth |
|------|-------|--------|-------|--------|-------|
| Number of Bed-sheets | 600 | 850 | 700 | 300 | 900 |

**Draw the bar graph representing the above data**

```
import matplotlib.pyplot as p

x=['First','Second','Third','Fourth','Fifth']  # Assigning Data for X axis
y=[600,850,700,300,900]                         # Assingning Data for Y axis

p.title('Production By Factory')
p.xlabel('Week')
p.ylabel('NO. of Bed Sheets')
p.bar(x,y,color='Blue',width=.50) # To draw bar graph, with each bar of .50 width
p.show()
```

**21**     **The number of students in 7 different classes is given below. Represent this data on the bar graph.**

| Class | 6th | 7th | 8th | 9th | 10th | 11th | 12th |
|-------|-----|-----|-----|-----|------|------|------|
| Number of Students | 130 | 120 | 135 | 130 | 150 | 80 | 75 |

```
import matplotlib.pyplot as p

x=[6,7,8,9,10,11,12]           # Assigning Data for X axis
y=[130,120,135,130,150,80,75] # Assingning Data for Y axis

p.title('Class Strength')
p.xlabel('CLASS')
p.ylabel('NO. of Students')
p.bar(x,y,color='green',width=.40) # To draw bar graph, with each bar of .40 width
p.show()
```

**22**     **The number of students in 7 different classes is given below. Represent this data on the bar graph.**

| Class | 6th | 7th | 8th | 9th | 10th | 11th | 12th |
|-------|-----|-----|-----|-----|------|------|------|
| Number of Students | 130 | 120 | 135 | 130 | 150 | 80 | 75 |

```
import matplotlib.pyplot as p

x=[6,7,8,9,10,11,12]          # Assigning Data for X axis
y=[130,120,135,130,150,80,75] # Assingning Data for Y axis

p.title('Class Strength')
p.xlabel('CLASS')
p.ylabel('NO. of Students')
p.bar(x,y,color='green',width=.40) # To draw bar graph, with each bar of .40 width
p.show()
```

23     **An analysis has been done in the school to identify hobby of Students as given below.**

| Hobby | Music | Dance | Games | Reading | Drawing |
|---|---|---|---|---|---|
| Number of Students | 130 | 150 | 180 | 75 | 160 |

**Represent this data on the Pie Chart . Slice colour must be pink,green,blue,gold and light sky blue**

```
import matplotlib.pyplot as p
L=["Drawing","Music","Dance","Games","Reading"]  # Defining labels
nos=[130,150,180,75,160] # Assingning Data for pie
c=["pink","green","blue","gold","lightskyblue"]
expld= ( 0.1,0,0,0,0)
p.figure ( figsize=[8,5])
p.title ( 'Hobbies Analysis')
p.pie ( nos,explode=expld,labels=L,colors=c,autopct='%.2f%%' ,shadow=True,startangle=170)
p.show ()
```

24     **The Production(in Tone) by Factory in Years is shown below Represent this data on the scatter graph.**

| Year | 2000 | 2005 | 2010 | 2015 |
|---|---|---|---|---|
| Production in Tons | 50 | 40 | 30 | 60 |

```
import matplotlib.pyplot as p
y=[50,40,30,60]    # Assingning Data for Y axis
x=[2000,2005,2010,2015]    # Assingning Data for X axis
p.title('Production By Factory')
p.grid(True)
p.ylabel('Production of Iron in Tons')
p.xlabel('Year')
p.scatter(x,y,color='b') # To draw SCATTER graph WITH BLUE DOTS
p.show()
```

25     **Consider the Following set of Data**
**34,18,100,27,54,52,93,59,61,87,68,85,78,82,9 .Create a box plot and add title as**
**Horizontal Boxplot and y-axis title as "Value Range"**

```
import matplotlib.pyplot as plt
D=[34,18,100,27,54,52,93,59,61,87,68,85,78,82,9]
plt.boxplot(D,vert=False,showmeans=True)
plt.title("Horizontal Boxplot")
plt.xlabel("value Range")
plt.show()
```

**26**        **Consider the table given below and write the query for the following**
                                **Table:  CLUB**

| GCode | GameName | Number | Fees | StartingDate |
|-------|----------|--------|------|--------------|
| 101 | Carom | 2 | 5000 | 2004-01-23 |
| 102 | Badminton | 2 | 12000 | 2003-12-12 |
| 103 | Table | 4 | 8000 | 2004-02-14 |
| 104 | Chess | 2 | 9000 | 2004-01-01 |
| 105 | Lawn Tennis | 4 | 25000 | 2004-03-19 |

1. **To display the name of all games with their Gcodes.**
2. **To display details of those games which are having Fees more than 7000.**
3. **To display the content of the CLUB table in descending order of startDate.**
4. **To delete the record of all  GameNames.**
5. **List the minimum and maximum fees from CLUB.**

**27**        Consider the tables *FLIGHTS* & *FARES*. Write SQL commands for the statements

### Table : FLIGHTS

| FNO | SOURCE | DEST | NO_OF_FL | NO_OF_STOP |
|-----|--------|------|----------|------------|
| IC301 | MUMBAI | BANGALORE | 3 | 2 |
| IC799 | BANGALORE | KOLKATA | 8 | 3 |
| MC101 | DELHI | VARANASI | 6 | 0 |
| IC302 | MUMBAI | KOCHI | 1 | 4 |
| AM812 | LUCKNOW | DELHI | 4 | 0 |
| MU499 | DELHI | CHENNAI | 3 | 3 |

### Table : FARES

| FNO | AIRLINES | FARE | TAX_percentage |
|-----|----------|------|----------------|
| IC301 | Indian Airlines | 9425 | 5 |
| IC799 | Spice Jet | 8846 | 10 |
| MC101 | Deccan Airlines | 4210 | 7 |
| IC302 | Jet Airways | 13894 | 5 |
| AM812 | Indian Airlines | 4500 | 6 |
| MU499 | Sahara | 12000 | 4 |
|  |  |  |  |

i)  Display flight number & number of flights from Mumbai from the table flights.
ii) Arrange the contents of the table flights in the descending order of destination.        iii) Increase the tax by 2% for the flights starting from Delhi.
iv) Display the flight number and fare to be paid for the flights from Mumbai to Kochi using the tables, Flights & Fares, where the fare to be paid =fare+fare*tax/100.

v) Display total no of source stations(eliminate duplicate) present in the table.
vi) Display the fare for the flight for MUMBAI to BANGLORE
vii)Display the records of source stations started with letter 'B'.
 viii) Display the flight no. for which fare to be paid is less than 3000.
ix) Display total no. of flights available for each Airlines
x) Add a new column Dep_Time in the table Flight.
xi) Delete the record of flight no. IC301 from the table FARE.xii) increase the size of the column 'source' to 30 in the Table FLIGHT

**28**    Consider the following tables Employee and salary. Write SQL commands for the statements (i) to (iv)

### Table : Employee

| Eid | Name | Deptid | Qualification | Sex |
|-----|------|--------|---------------|-----|
| 1 | Deepali Gupta | 101 | MCA | F |
| 2 | Rajat Tyagi | 101 | BCA | M |
| 3 | Hari Mohan | 102 | B.A | M |
| 4 | Harry | 102 | M.A | M |
| 5 | Sumit Mittal | 103 | B.Tech | M |
| 6 | Jyoti | 101 | M.Tech | F |

### Table : Salary

| Eid | Basic | DA | HRA | Bonus |
|-----|-------|-----|-----|-------|
| 1 | 6000 | 2000 | 2300 | 200 |
| 2 | 2000 | 300 | 300 | 30 |
| 3 | 1000 | 300 | 300 | 40 |
| 4 | 1500 | 390 | 490 | 30 |
| 5 | 8000 | 900 | 900 | 80 |
| 6 | 10000 | 300 | 490 | 89 |

(i)    To display the frequency of employees department wise.
(ii)   To list the names of those employees only whose name starts with 'H'
(iii)  To add a new column in salary table . the column name is total_sal.
(iv)   To store the corresponding values in the total_sal column.

**29. Observe the following tables and write the queries on the basis the given tables :-**
**Table: hospital**

| | Name | Age | Department | Charges | Gender |
|----|------|-----|------------|---------|--------|
| 1 | Arprit | 62 | Surgery | 300 | M |
| 2 | Zarina | 22 | ENT | 250 | F |
| 3 | Kareem | 32 | Orthopaedic | 200 | M |
| 4 | Arun | 12 | Surgery | 300 | M |
| 5 | Zubin | 30 | ENT | 250 | M |
| 6 | Kettaki | 16 | ENT | 250 | F |
| 7 | Ankita | 29 | Cardiology | 800 | F |
| 8 | Zareen | 45 | Gynecology | 300 | F |
| 9 | Kush | 19 | Cardiology | 800 | M |
| 10 | Shilpa | 23 | Nuclear Medicine | 400 | F |

**Write MySql command for the following:**
   **a) To displaye Total no. of employees present in the Hospital**
   **b) To display all information about the patients of cardiology department.**
   **c)  To list the name of female patients who are in ENT department.**
   **To display name and gender of all the patients whose age is in the range of 40 to 50 in ascending order of their name.**

Consider the following tables SCHOOL and ADMIN. Write SQL commands for the statements (i) to (iv)).

**SCHOOL**

| CODE | TEACHERNAME | SUBJECT | DOJ | PERIODS | EXPERIENCE |
|------|-------------|---------|-----|---------|------------|
| 1001 | RAVI SHANKAR | ENGLISH | 12/03/2000 | 24 | 10 |
| 1009 | PRIYA RAI | PHYSICS | 03/09/1998 | 26 | 12 |
| 1203 | LISA ANAND | ENGLISH | 09/04/2000 | 27 | 5 |
| 1045 | YASHRAJ | MATHS | 24/08/2000 | 24 | 15 |
| 1123 | GANAN | PHYSICS | 16/07/1999 | 28 | 3 |
| 1167 | HARISH B | CHEMISTRY | 19/10/1999 | 27 | 5 |
| 1215 | UMESH | PHYSICS | 11/05/1998 | 22 | 16 |

**ADMIN**

| CODE | GENDER | DESIGNATION |
|------|--------|-------------|
| 1001 | MALE | VICE PRINCIPAL |
| 1009 | FEMALE | COORDINATOR |
| 1203 | FEMALE | COORDINATOR |
| 1045 | MALE | HOD |
| 1123 | MALE | SENIOR TEACHER |
| 1167 | MALE | SENIOR TEACHER |
| 1215 | MALE | HOD |

i) To display TEACHERNAME, PERIODS of all teachers whose periods less than 25.
ii) To display TEACHERNAME, CODE and DESIGNATION from tables SCHOOL and ADMIN whose gender is male.
iii) To display the number of teachers in each subject.
iv) To display CODE, TEACHERNAME and SUBJECT of all teachers who have joined the school after 01/01/1999.

**30, Design a Django Based Application to obtain a search criteria and fetches record based on that from Books Table.**

**31. Design a Django based application that fetches all records from student table of School database.**

**32. Design a Django based application that fetches all records of those employees who are 'Salesman'**