NAME:- CHAUHAN AJAY
EN.NO:-2202031000014
SUBJECT :-OJT PRACTICALS(INTERNSHIP-II)
COURSE:-B.TECH(IT)

1.

Aim:-Write a C program to print the address of a variable using a pointer.

```c
#include <stdio.h>

int main() {
    int num= 10;
    int *ptr = &num;

    printf("Address of num: %p\n",
&num);
    printf("Value of num: %d\n", num);

    printf("Address of ptr: %p\n",
&ptr);
    printf("Value of ptr: %p\n", ptr);
    printf("Value pointed to by ptr:
%d\n", *ptr);

    return 0;
}
```

Output:-

Address of x: 0061FF1C

Value of x: 10

Address of ptr: 0061FF18

Value of ptr: 0061FF1C

Value pointed to by ptr: 10

2.

Aim:-Write a C program to create a Calculator using a pointer.

```
#include<stdio.h>
{
    main()
    int x,y,choice;
    int *ptr1,*ptr2,result;
    ptr=&x;
    ptr=&y;
    printf("Enter the value 1:");
    scanf("%d",ptr1);
    printf("Enter the value 2:");
    scanf("%d"ptr2);
    printf("select all operation:");
    printf("1. sum\n");
    printf("2. sub\n");
```

```
    printf("3. multi\n");
    printf("4. div\n");
    scanf("%d",choice);
    switch (choice)
    {
        case 1:
        result=*ptr1+*ptr2;
        printf("result:%d\n",result);
        break;

         case 2:
        result=*ptr1 - *ptr2;
        printf("result:%d\n",result);
        break;

         case 3:
        result=*ptr1 * *ptr2;
        printf("result:%d\n",result);
        break;

         case 4:
        result=*ptr1 / *ptr2;
        printf("result:%d\n",result);
        break;
```

```
        default
        printf("invild choice\n:");
        break;
        return 0;
    }
}
```

Output:-

Enter first number: 1

Enter second number: 5

Select an operation:

1. Addition

2. Subtraction

3. Multiplication

4. Division

2

Result: -4

3.

Aim:-Write a C program to swap the two values using call by value and call by reference.

```
#include <stdio.h>
void swap(int *num1, int *num2);
int main()
{
```

```
  int num1, num2;
  printf("Enter two numbers: \n");
  scanf("%d %d", &num1, &num2);
  printf("Before swapping in main n:\n");
  printf("Value of num1 = %d \n", num1);
  printf("Value of num2 = %d \n\n", num2);
  swap(&num1, &num2);
  printf("After swapping in main n:\n");
  printf("Value of num1 = %d \n", num1);
  printf("Value of num2 = %d \n\n", num2);
  return 0;
}
void swap(int *num1, int *num2)
{
  int temp;
  temp = *num1;
  *num1 = *num2;
  *num2 = temp;
  printf("After swapping in swap function
n:\n");
  printf("Value of num1 = %d \n", *num1);
  printf("Value of num2 = %d \n\n", *num2);
}
```

Output:-

Enter two numbers: 10

20

Before swapping in main n:

Value of num1 = 10

Value of num2 = 20

After swapping in swap function n:

Value of num1 = 20

Value of num2 = 10

After swapping in main n:

Value of num1 = 20

Value of num2 = 10

4.

Aim:- . **Define a structure type struct personal that would contain person name, Date of birth and age using this**

Structure to read this information of 4 people and display the same.

```c
#include <stdio.h>
struct personal
{
  char name[50];
  int birth_day;
  int birth_month;
  int birth_year;
  int age;
};
int main()
{
  struct personal people[4];
  int i;
  for (i = 0; i < 4; i++)
```

```
  {
    printf("Enter person %d name: ", i + 1);
    scanf("%s", people[i].name);
    printf("Enter person %d date of birth (DD MM YYYY): ",
i + 1);
    scanf("%d %d %d", &people[i].birth_day,
&people[i].birth_month, &people[i].birth_year);
    printf("\n");
    people[i].age = 2023 - people[i].birth_year;
  }
  for (i = 0; i < 4; i++)
  {
    printf("Person %d:\n", i + 1);
    printf("Name: %s\n", people[i].name);
    printf("Date of Birth: %02d/%02d/%04d\n",
people[i].birth_day, people[i].birth_month,
people[i].birth_year);
    printf("Age: %d\n", people[i].age);
    printf("\n");
  }

  return 0;
}
```

Output:-

- **Enter person 1 name: Ram**

Enter person 1 date of birth (DD MM YYYY): 08 8 2005

- **Enter person 2 name: ramu**

Enter person 2 date of birth (DD MM YYYY): 17 09 2006

- **Enter person 3 name: ajay**

Enter person 3 date of birth (DD MM YYYY): 29 10 2003

- **Enter person 4 name: jatin**

Enter person 4 date of birth (DD MM YYYY): 03 09 2004

- **Person 1:**

Name: Ram

Date of Birth: 08/08/2005

Age: 18

- **Person 2:**

Name: ramu

Date of Birth: 17/09/2006

Age: 17

- **Person 3:**

Name: ajay

Date of Birth: 29/10/2003

Age: 20

- **Person 4:**

Name: jatin

Date of Birth: 03/09/2004

Age: 19

5.

Aim:- Write a C program to calculate the sum of n numbers entered by the user using dynamic memory allocation.

```
#include<stdio.h>
#include<stdlib.h>
```

```c
int main()
{
  int i,a,*ptr,sum=0;
  printf("Enter The Number Of Element:");
  scanf("%d",&a);
  ptr=(int*)malloc(a* sizeof(int));
  if(ptr==NULL)
  {
    printf("Memory allocation faild.");
    return 1;
  }
  printf("Enter %d Elements:",a);
  for(i=0;i< a;i++)
  {
    scanf("%d",ptr+i);
    sum+=*(ptr+i);
  }
  printf("sum = %d ",sum);
  free(ptr);
  return 0;
}
```

Output:-

Enter The Number Of Element:5

Enter 5 Elements:4

5

6

7

8

sum = 30

6.

Aim:- A file named “New” contains a series of integer numbers. Write a c program to read all numbers from a file and then copy all odd numbers into a file named “odd” and write all even numbers into a file named “even”.

```c
#include <stdio.h>

int main()
{
  FILE *fp, *fpo, *fpe;
  int num;
  fp = fopen("New.Text", "r");
  if (fp == NULL)
  {
    printf("Error openig file!\n");
    return 1;
  }
  fpo =fopen("odd.text", "w");
  if (fopen == NULL)
  {
    printf("Error Creating odd File!\n");
    return 1;
  }
  fpe =fopen("even.text", "w");
  if (fpe == NULL)
  {
    printf("Error Creating even file!\n");
    return 1;
```

```
}
while (fscanf(fp, "%d", &num) != EOF)
{
  if (num % 2 == 0)
  {
    fprintf(fpe, "%d", num);
  }
  else
  {
    fprintf(fpo, "%d", num);
  }
}
fclose(fp);
fclose(fpo);
fclose(fpe);
// DISPLAY CONTENTS OF ODD NUMBER
printf("Odd Number:\n");
fpo = fopen("odd.text", "r");
if (fpo == NULL)
{
  printf("Error Opning Odd File!\n");
  return 1;
}
while (fscanf(fpo, "%d\n", num) != EOF)
{
  printf("%d\n", num);
}
fclose(fpo);
// DISPLAY CONTENT OF EVEN FILE
```

```
  printf("Even Number:\n");
  fpe = fopen("even.text", "r");
  if (fpe = NULL)
  {
    printf("Error Opining even file !\n");
    return 1;
  }
  while (fscanf(fpe, "%d\n", &num) != EOF)
  {
    printf("%d\n", num);
  }
  fclose(fpe);
  return 0;
}
```

Output:-

Odd Numbers:

1

3

5

7

9

Even Numbers:

2

4

6

8

10

7.

Aim:- Write a C++ program to Check if the number is prime or not using a function

```
#include <iostream>
using namespace std;
void Prime(int num) {
    if (num <= 1) {
        cout << num << " is not a prime number" << endl;
        return;
    }

    for (int i = 2; i <= num/2; i++) {
        if (num % i == 0) {
            cout << num << " is not a prime number" <<
endl;
            return;
        }
    }
    cout << num << " is a prime number" << endl;
}
    int main() {
    int num;
    cout << "Enter a number: ";
    cin >> num;
    Prime(num);
    return 0;
}
```

Output:-

Enter a number: 0

0 is not a prime number

8.

Aim:- Write a C++ program that prompts the user to enter a letter and check whether a letter is a vowel or constant.

```
#include<iostream>
using namespace std;
void vovel()
{
  char c;
  cout<<"Enter The Character:";
  cin>>c;

if(c=='a'||c=='e'||c=='i'||c=='o'||c=='u'||c=='A'||c=='E'|
|c=='I'||c=='O'||c=='U')
  {
    cout<<" The Character Is Vovel";
  }
  else
  {
    cout<<"The Character Is Cosonant";
  }
}
  int main()
  {
    vovel();
    return 0;
  }
```

Output:-

Enter The Character:b

The Character Is Cosonant

Enter The Character:a

The Character Is Vovel

9.

Aim:- Write a C++ program to demonstrate the concept of constructor and destructor.

```
#include<iostream>
using namespace std;
class example{
  public:
  example(){

  cout<<"Constructor called"<<endl;
  }
  ~example()
  {
    cout<<"Destructor called"<<endl;
  }
};
int main()
{
  example obj;
  return 0;
}
```

Output:-

Constructor called

Destructor called

10.

Aim:- Create a class student that stores roll_nos, name. Create a class test that stores marks obtained in five subjects. Class result derived from student and test contains the total marks and percentage obtained in test. Input and display information of a student.

```
#include <iostream>
#include <string>
using namespace std;
class Student
{
public:
    int roll_no;
    string name;
};
class Test
{
public:
    int subject_marks[5];
    int total_marks()
    {
        int total = 0;
        for (int i = 0; i < 5; i++)
        {
            total += subject_marks[i];
        }
        return total;
    }
    double percentage()
    {
        return (total_marks() / 5.0);
    }
};
class Result : public Student, public Test
{
public:
    Result(int roll_no, string name, int marks[])
    {
        this->roll_no = roll_no;
        this->name = name;
        for (int i = 0; i < 5; i++)
        {
```

```
            subject_marks[i] = marks[i];
        }
    }
};
int main()
{
    int marks[] = {80, 90, 85, 75, 95};
    Result r(1, "John Doe", marks);
    cout << "Roll No: " << r.roll_no << endl;
    cout << "Name: " << r.name << endl;
    cout << "Marks: ";
    for (int i = 0; i < 5; i++)
    {
        cout << r.subject_marks[i] << " ";
    }
    cout << endl;
    cout << "Total Marks: " << r.total_marks() << endl;
    cout << "Percentage: " << r.percentage() << endl;
    return 0;
}
```

Output:-

Roll No: 5

Name: John Doe

Marks: 80 90 85 75 95

Total Marks: 425

Percentage: 85

11.

Aim:- :- Write a C++ program to overload binary + operator.

```
#include <iostream>
using namespace std;
class Number
{
private:
    int value;

public:
    Number(int v = 0) : value(v) {}
    Number operator+(const Number &n)
    {
```

```
        return Number(value + n.value);
    }
    void display()
    {
        cout << "Value: " << value << endl;
    }
};
int main()
{
    Number n1(5), n2(10), n3;
    n3 = n1 + n2;
    n3.display();
    return 0;
}
```

Output:-

Value:15

12.

Aim:- Create a base class called 'SHAPE' having two data members of type double, member function get_data( ) to initialize base class data members, pure virtual member function display_area( ) to compute and display the area of the geometrical object. Derive two specific classes 'TRIANGLE' and 'RECTANGLE' from the base class. Using these three classes design a program that will accept dimension of a triangle / rectangle interactively and display the area.

Code:-

```
#include <iostream>
using namespace std;
class Shape
{
public:
    virtual void draw()
    {
        cout << "Drawing a shape" << endl;
    }
};
class Circle : public Shape
{
public:
    void draw()
    {
        cout << "Drawing a circle" << endl;
    }
};
class Rectangle : public Shape
{
```

```
public:
    void draw()
    {
        cout << "Drawing a rectangle" << endl;
    }
};
int main()
{
    Shape *s = new Shape();
    Shape *c = new Circle();
    Shape *r = new Rectangle();

    s->draw();
    c->draw();
    r->draw();

    return 0;
}
```

Output:-

Drawing a shape

Drawing a circle

Drawing a rectangle

DBMS

13.

Aim:- - To study DDL-create and DML-insert commands.

DDL (Data Definition Language) and DML (Data Manipulation Language) are two different types of SQL (Structured Query Language) commands used to manage and manipulate databases.

DDL commands are used to define and manipulate the structure of the database, including creating tables, altering tables, adding columns, and deleting tables

DML commands, on the other hand, are used to manipulate the data within the database, including

inserting, updating, and deleting data in tables

Let's start by looking at the syntax and usage of DDL-create and DMLinsert commands:

DDL-Create Command:

The create command is used to create a new table in the database. The basic syntax of the create command is as follows:

CODE :-

```
CREATE TABLE table_name
( column1 datatype,
column2 datatype,
column3 datatype,
.... );
```

For example, to create a table named "customers" with columns for "id", "name", "email", and "phone", the following command would be used:

CODE :-

```
CREATE TABLE customers (
id INT PRIMARY KEY,
name VARCHAR(50),
email VARCHAR(50),
phone VARCHAR(20)
);
```

DML-Insert Command:

The insert command is used to insert data into a table in the database. The basic syntax of the insert command is as follows:

CODE :-

```
NSERT INTO table_name (column1, column2, column3, ...)
VALUES (value1, value2, value3, ...);
```

For example, to insert a new row of data into the "customers" table created in the previous example, the following command would be used:

Code:-

INSERT INTO customers (id, name, email, phone)

VALUES (1, 'John Doe', 'john@example.com', '555-1234');

This command would insert a new row into the "customers" table with the values specified for the "id", "name", "email", and "phone" columns.

I hope this helps you understand the basics of DDL-create and DMLinsert commands in SQL. If you have any further questions, feel free to ask!

14.

Aim:-

(A). Job (job_id, job_title, min_sal, max_sal)

COLUMN NAME	**DATA TYPE**
job_id	Varchar(15)
job_title	Varchar(30)
min_sal	int
max_sal	int

Code:-

```
CREATE TABLE Job (
 job_id VARCHAR(15) PRIMARY KEY,
job_title VARCHAR(30),
min_sal INT,
 max_sal INT
);
```

(B). Employee (emp_no, emp_name, emp_sal, emp_comm, dept_no)

COLUMN NAME	DATA TYPE
emp_no	Int
emp_name	Varchar(30)
emp_sal	Decimal(8,1)
emp_comm	Decimal(6,1)
dept_no	Int

Code:-

```
CREATE TABLE Employee (
 emp_no INT PRIMARY KEY,
 emp_name VARCHAR(30),
 emp_sal DECIMAL(8,2),
emp_comm DECIMAL(6,1),
dept_no INT );
```

(c).deposit(a_no,cname,bname,amount,a_date)

COLUMN NAME	DATA TYPE
A_no	Int,identity

C_name	Varchar(50)
B_name	Varchar(30)
Amount	Decimal(4,2)
a_date	date

Code:-

```
CREATE TABLE deposit (
ano INT IDENTITY PRIMARY KEY,
cname VARCHAR(50),
 bname VARCHAR(30),
 amount DECIMAL(4,2),
 a_date DATE
```

(D). borrow(loanno,cname,bname,amount)

COLUMN NAME	DATA TYPE
loanno	Int
cname	Varchar(25)
bname	Varchar(20)
amount	Decimal(6,2)

Code:-

```
CREATE TABLE borrow (
 loanno INT PRIMARY KEY,
 cname VARCHAR(25),
```

```
 bname VARCHAR(20),
amount DECIMAL(6,2)
 );
```

16.

Aim:-Create tables and insert sample data in tables. Write SQL queries to insert following data into tables

Code:-

```
CREATE TABLE Employee (
emp_no INT PRIMARY KEY,
emp_name VARCHAR(30) NOT NULL,
emp_sal DECIMAL(8,2) NOT NULL,
emp_comm DECIMAL(6,1),
dept_no INT NOT NULL
);
INSERT INTO Employee (emp_no, emp_name, emp_sal, emp_comm, dept_no)
VALUES
(101, 'Smith', 800.00, 20, 0),
(102, 'Snehal', 1600.00, 300, 25),
(103, 'Adama', 1100.00, 0, 20),
(104, 'Aman', 3000.00, 15, 0),
(105, 'Anita', 5000.00, 50000, 10),
(106, 'Sneha', 2450.00, 24500, 10),
(107, 'Anamika', 2975.00, 30, 0)
```

(b).

AIM :- Create tables and insert sample data in tables. Write SQL queries to insert following data into tables Insert following values in the table Job

job_id	**job_name**	**min_sal**	**max_sal**
IT_PROG	Programmer	4000	10000
MK_MGR	Marketing manager	9000	15000
FI_MGR	Finance manager	8200	12000
FI_ACC	Account	4200	9000
LEC	Lecturer	6000	17000
COMP_OP	Computer Operator	1500	3000

Code:-

```
CREATE TABLE Job (
job_id VARCHAR(15) PRIMARY KEY,
job_name VARCHAR(30) NOT NULL,
min_sal DECIMAL(10,2) NOT NULL,
max_sal DECIMAL(10,2) NOT NULL
);
INSERT INTO Job (job_id, job_name, min_sal, max_sal) VALUES
('IT_PROG', 'Programmer', 4000.00, 10000.00),
('MK_MGR', 'Marketing manager', 9000.00, 15000.00),
('FI_MGR', 'Finance manager', 8200.00, 12000.00),
('FI_ACC', 'Account', 4200.00, 9000.00),
('LEC', 'Lecturer', 6000.00, 17000.00),
('COMP_OP', 'Computer Operator', 1500.00, 3000.00);
```

(c).

Aim:-

A_no	cname	bname	Amount	date
101	anil	Andheri	7000	01-jun-06
102	sunil	Virar	5000	15-jul-06
103	jay	Villeparle	6500	12-mar-06
104	vijay	Andheri	8000	17-sep-06
105	Keyur	Dadar	7500	19-nov-06
106	mayur	borivali	5500	21-dec-06

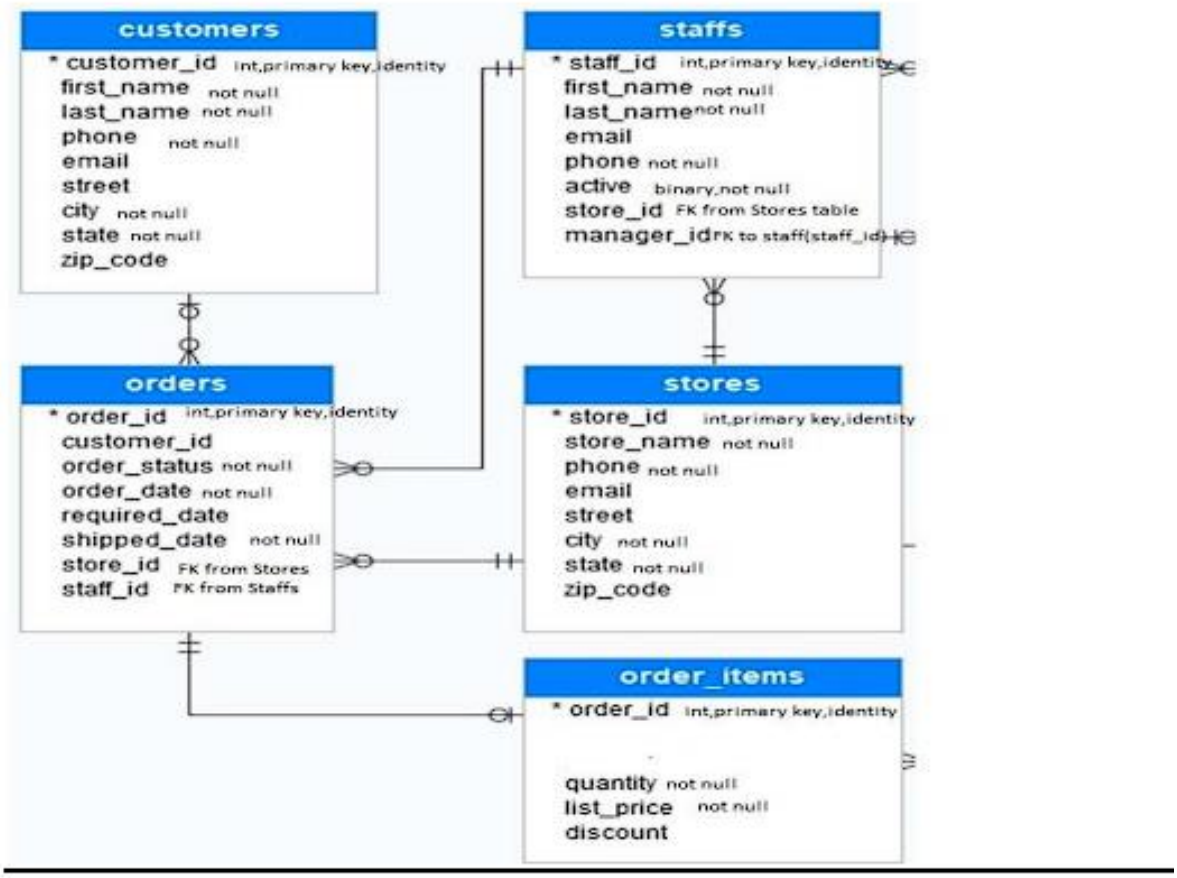
Code:-

```
CREATE TABLE deposit (
a_no INT IDENTITY PRIMARY KEY,
cname VARCHAR(50) NOT NULL,
bname VARCHAR(30) NOT NULL,
amount DECIMAL(8,2) NOT NULL,
a_date DATE NOT NULL
);
INSERT INTO deposit (cname, bname, amount, a_date) VALUES
('Anil', 'andheri', 7000.00, '2006-01-01'),
('sunil', 'virar', 5000.00, '2006-07-15'),
('jay', 'villeparle', 6500.00, '2006-03-12'),
('vijay', 'andheri', 8000.00, '2006-09-17'),
('keyur', 'dadar', 7500.00, '2006-11-19'),
('mayur', 'borivali', 5500.00, '2006-12-21');
```

16.

Aim:-

Write the SQL queries to provide constraints on given tables.Create A Database Sales and Write SQL Queries to create following tables with all constrains mentioned in image.



Code:-

```
CREATE TABLE customers (
customer_id INT PRIMARY KEY,
first_name VARCHAR(50) NOT NULL,
last_name VARCHAR(50) NOT NULL,
phone VARCHAR(20),
```

```
email VARCHAR(100),
street VARCHAR(100),
city VARCHAR(50) NOT NULL,
state VARCHAR(50) NOT NULL,
zip_code VARCHAR(20)
);
CREATE TABLE staff (
staff_id INT PRIMARY KEY,
first_name VARCHAR(50) NOT NULL,
last_name VARCHAR(50) NOT NULL,
email VARCHAR(100),
phone VARCHAR(20) NOT NULL,
active BOOLEAN NOT NULL,
store_id INT,
FOREIGN KEY (store_id) REFERENCES stores(store_id)
);
CREATE TABLE stores (
store_id INT PRIMARY KEY,
store_name VARCHAR(50) NOT NULL,
phone VARCHAR(20) NOT NULL,
email VARCHAR(100),
street VARCHAR(100),
city VARCHAR(50) NOT NULL,
state VARCHAR(50) NOT NULL,
zip_code VARCHAR(20),
manager_id INT,
```

```
FOREIGN KEY (manager_id) REFERENCES staff(staff_id)
);
CREATE TABLE orders (
order_id INT PRIMARY KEY,
order_date DATE NOT NULL,
required_date DATE,
shipped_date DATE NOT NULL,
order_status VARCHAR(20) NOT NULL,
customer_id INT,
staff_id INT,
store_id INT,
FOREIGN KEY (customer_id) REFERENCES customers(customer_id),
FOREIGN KEY (staff_id) REFERENCES staff(staff_id),
FOREIGN KEY (store_id) REFERENCES stores(store_id)
);
CREATE TABLE order_items (
order_id INT,
item_id INT PRIMARY KEY,
quantity INT NOT NULL,
list_price DECIMAL(10, 2) NOT NULL,
discount DECIMAL(5, 2),
FOREIGN KEY (order_id) REFERENCES orders(order_id)
);
```

17.

Aim:- Write the SQL queries to perform various aggregate functions on table data

1. List total deposit from deposit.

CODE :- SELECT SUM(amount) AS total_deposit FROM deposit;

2. List total amount from andheri branch

CODE :- SELECT SUM(amount) AS total_amount FROM deposit WHERE bname = 'andheri';

3. Count total number of customers

CODE :- SELECT COUNT(DISTINCT cname) AS total_customers FROM deposit;

4. Count total number of customer's cities

CODE :- SELECT COUNT(DISTINCT bname) AS total_cities FROM deposit;

5. Update the value dept_no to 10 where second character of emp. name is 'm'.

CODE :- UPDATE Employee SET dept_no = 10 WHERE emp_name LIKE '_m%';

6. Update the value of employee name whose employee number is 103.

CODE :- UPDATE Employee SET emp_name = 'Adam' WHERE emp_no = 103;

7. Write a query to display the current date. Label the column Date

CODE :- SELECT GETDATE() AS Date;

8. For each employee, display the employee number, salary, and salary in

creased by 15% and expressed as a whole number. Label the column New Salary

CODE :- SELECT emp_no, emp_sal, ROUND(emp_sal*1.15,0) AS "New Salary" FROM Employee;

9. Modify your previous query to add a column that subtracts the old salary from the new salary. Label the column Increment.

CODE :- SELECT emp_no, emp_sal, ROUND(emp_sal*1.15,0) AS "New Salary",

ROUND(emp_sal*0.15,0) AS Increment FROM Employee;

18.

1. Retrieve all data from employee, jobs and deposit.

CODE :- SELECT * FROM employee;

SELECT * FROM jobs;

SELECT * FROM deposit;

2. Give details of account no. and deposited rupees of customers having account

opened between dates 01-01-06 and 25-07-06.

CODE :- SELECT a_no, amount FROM deposit

WHERE a_date BETWEEN '2006-01-01' AND '2006-07-25';

3. Display all jobs with minimum salary is greater than 4000.

CODE:- SELECT * FROM jobs

WHERE min_sal > 4000;

4. Display name and salary of employee whose department no is 20. Give alias

name to name of employee.

CODE :- SELECT emp_no, emp_name AS employee_name, emp_sal, dept_no

FROM

employee

WHERE dept_no = 20;

5. Display employee no,name and department details of those employee whose

department lies in(10,20)

CODE :- SELECT emp_no, emp_name, dept_no FROM employee

WHERE dept_no IN (10, 20);

6. Display all employee whose name start with 'A' and third character is ' 'a'.

CODE :- SELECT * FROM employee

WHERE emp_name LIKE 'A _a%';

7. Display name, number and salary of those employees whose name is 5 characters long and first three characters are 'Ani'.

CODE :-

SELECT emp_name, emp_no, emp_sal FROM employee

WHERE emp_name LIKE 'Ani___';

8. Display the non-null values of employees and also employee name second charactershould be 'n' and string should be 5 character long.

CODE :-

SELECT * FROM employee

WHERE emp_name LIKE '_n%' AND LENGTH(emp_name) = 5 AND emp_name IS

NOT NULL;

9. Display the null values of employee and also employee name's third character

should be 'a'.

CODE :-

SELECT * FROM employee

WHERE emp_name LIKE '__a%' AND emp_name IS NULL;

HTML :: CSS ::JS

18.

Aim:- Make a Resume using the HTML tags without CSS.

Code:-

```
<!doctype html>
<html>

    <head>
        <title>resume</title>
    </head>

    <body>
        <fieldset>
            <center>
                <h1><u><b>RESUME</b></u></h1>
            </center>
            <span>Photo:</span>
            <img src="" />
            <br>
            <span>First Name:</span>
            <input type="text area" />
            <br>
            <span>Second Name:</span>
            <input type="text area" />
            <br>
            <span>DOB:</span>
            <input type="date" />
            <br>
            <span>E-Mail</span>
            <input type="text area" />
            <br>
            <span>Gender:</span>
            <span>Male</span>
            <input type="radio" id="g1" name="gender" />
            <span>Female</span>
            <input type="radio" id="g2" name="gender" />
```

```
        <br>
        <span>Address:</span>
        <textarea plceholder="Enter The Address"></textarea>
        <br>
        <span>Hobbies:</span>
        <input type="checkbox" />Reading
        <input type="checkbox" />Gaming
        <input type="checkbox" />travelling
        <input type="checkbox" />Cricket
        <hr>
        <center>
            <h3><u><b>SKILLS</b></u></h3>
        </center>
        <ol type="square">
            <li>Technical Skills</li>
            <li>Soft Skills</li>
            <li>Aptitude Skills</li>
            <li>Comunication Skills</li>
        </ol>
        <hr>
        <center>
            <h3><u><b>QUAILIFICATIONS</b></u></h3>
        </center>
        <table border=1 align="left">
            <thead>
                <tr>
                    <th>s.no</th>
                    <th>Class</th>
                    <th>Grade</th>
                    <th>Board</th>
                </tr>
            </thead>
            <tbody>
                <tr>
                    <td>1</td>
                    <td>B.Tech</td>
                    <td>Running</td>
                    <td>ASOIT</td>
                </tr>
                <tr>
                    <td>2</td>
                    <td>12<sup>th</sup> </td>
                    <td>B</td>
                    <td>GSHEB</td>
                </tr>
                <tr>
                    <td>3</td>
                    <td>10<sup>th</sup> </td>
```

```
                        <td>A</td>
                        <td>GSHEB</td>
                    </tr>
                </tbody>
            </table>
        </fieldset>
    </body>

</html>
```

Output:-

RESUME

Photo:
First Name:
Second Name:
DOB: dd-mm-yyyy
E-Mail
Gender: Male Female
Address:
Hobbies: Reading Gaming travelling Cricket

SKILLS

1. Technical Skills
2. Soft Skills
3. Aptitude Skills
4. Comunication Skills

QUAILIFICATIONS

s.no	Class	Grade	Board
1	B.Tech	Running	ASOIT
2	12th	B	GSHEB
3	10th	A	GSHEB

19.

Aim:- Create an HTML webpage that shows Poster Presentation using all Table Properties.

Code:-

```
<!doctype html>
<html>

    <head>
        <title> Table </title>
    </head>

    <body>
        <table border=1 align="center">
            <thead>
                <tr>
                    <th>s.no</th>
                    <th>first name</th>
                    <th>last name</th>
```

```
                <th>email</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td>1</td>
                <td>jatin </td>
                <td rowspan="2">chauhan</td>
                <td>Ajay69</td>
            </tr>
            <tr>
                <td>2</td>
                <td>Aditya </td>
                <td>Aditya19</td>
            </tr>
            <tr>
                <td>3</td>
                <td>jatin</td>
                <td>sharma</td>
                <td>jatin6969</td>
            </tr>
            <td>4</td>
            <td colspan="3">divya chauhan divya 123
            </td>
            </tr>
    </table>
</body>

</html>
```

Output:-

s.no	first name	last name	email
1	jatin	chauhan	Ajay69
2	Aditya		Aditya19
3	jatin	sharma	jatin6969
4	divya chauhan divya 123		

20.

Aim :- Create an HTML page table and form.

Code:-

```
<!doctype html>
<html>

    <head>
        <title>FORM TAGS</title>
    </head>

    <body>
        <form>
            <span>First Name:</span>
            <input type="text" autofocus placeholder="Enter The First Name" />
            <br><span>Last Name:</span>
            <input type="text" autofocus placeholder="Enter The Last Name" />
            <br>
            <span>Gender:</span>
            <span>Male</span>
            <input type="radio" id="g1" name="gender">
            <span>Female</span>
            <input type="radio" id="g1" name="gender">
            <br> <span>Hobby:</span>
            <input type="checkbox" />Cricket
            <input type="checkbox" />Badminton
            <input type="checkbox" />Football
            <input type="checkbox" />Kabadi
            <br><span>Address:</span>
            <textarea plceholder="Enter The Address"></textarea>
            <br><span>City:</span>
            <select>
                <option> ahemdabad</option>
                <option>surat</option>
                <option>rajkot</option>
            </select>
            <br>
            <span>File:</span>
            <input type="file" multi accept=".jpg,.png,.pdf,.gif ">
            <br>
            <span>Password:</span>
            <input type="password" />
            <br>
            <span>Date:</span>
            <input type="date">
            <br>
```

```
            <span>Time:</span>
            <input type="time">
            <br>
            <span>Color:</span>
            <input type="color">
            <br>
            <span>Range:</span>
            <input type="range" />
            <fieldset>
                <legend>sign in</legend>
                <span>username</span><br>
                <input type="text" /><br>
                <span>Password</span><br>
                <input type="password" /><br>
                <input type="button" value="sign in" />
            </fieldset>
        </form>
    </body>

</html
```

Output:-

21.

Aim:- Create Registration form and do proper validation with HTML 5 inbuilt functionality. (Don’t use JavaScript).

Code:-

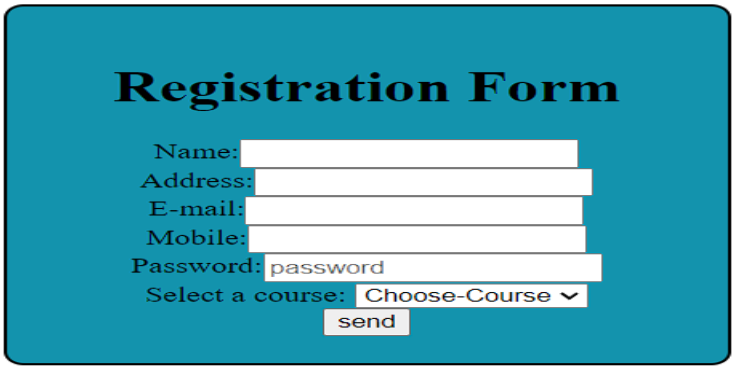
```
<!DOCTYPE html>
```

```
<head>
    <title>Registration Form</title>
    <style>
        #form {
            width: 320px;
            background-color: rgb(19, 147, 173);
            color: #000000;
            margin: 50px auto;
            padding: 20px;
            text-align: center;
            border-radius: 10px;
            border: 2px solid #000000;
        }

        h1 {
            text-align: center;
            margin-bottom: 20px;
        }
    </style>
    <script>
        function validation() {
            var name =
document.forms.regform.Name.value;
            var address =
document.forms.regform.address.value;
            var email =
document.forms.regform.email.value;
            var mobile =
document.forms.regform.mobile.value;
            var password =
document.forms.regform.password.value;
            var subject =
document.forms.regform.subject.value;
```

```
            alert(name + " " + address + " " + email + "
" + mobile + " " + password + " "
                + subject);
            if (name == "" || name == "null") {
                alert("Must Enter The Name");
            }
            if (address == "" || address == "null") {
                alert("Must Enter The Address");
            }
            if (email == "" || email == "null") {
                alert("Must Enter The E-mail");
            }
            if (mobile == "" || mobile == "null") {
                alert("Must Enter The Mobile");
            }
            if (password == "" || password == "null") {
                alert("Must Enter The Password");
            }
            if (subject == "" || subject == "null") {
                alert("Must Enter The Subject");
            }
        }
    </script>
</head>

<body>
    <div id="form">
        <h1>Registration Form</h1>
        <form name="regform" onsubmit="return
validation()" method="post">
            Name:<input type="text" name="Name" /><br>
            Address:<input type="text" name="address"
/><br>
            E-mail:<input type="text" name="email"
/><br>
```

```
            Mobile:<input type="text" name="mobile"
/><br>
            Password:<input type="text" name="password"
placeholder="password" /><br>
            <label for="subject">Select a
course:</label>
            <select id="subject" name="subject">
                <option value="">Choose-Course</option>
                <option value="math">Math</option>
                <option value="english">English</option>
                <option value="history">History</option>
                <option value="science">Science</option>
            </select>
            <br>
            <input type="submit" value="send"
name="submit" /><br>
        </form>
    </div>
</body>

</html>
```

Output:-



22.

Aim:-make a resume using html tag with css.

```
<!DOCTYPE html>
<html lang="en">

    <head>
        <meta charset="UTF-8">
        <title>Resume</title>
        <style>
            body {
                font-family: Arial, sans-serif;
                margin: 0;
                padding: 0;
                font-size: 16px;
                line-height: 1.5;
                color: #333;
            }

            .container {
                max-width: 800px;
                margin: 0 auto;
                padding: 50px;
                box-sizing: border-box;
                background-color: #f2f2f2;
            }

            h1,
            h2 {
                margin-top: 0;
                margin-bottom: 10px;
            }

            h1 {
                font-size: 36px;
                font-weight: bold;
                color: #333;
```

```
        }

        h2 {
            font-size: 24px;
            font-weight: bold;
            color: #333;
        }

        p {
            margin: 0;
            margin-bottom: 10px;
        }

        ul {
            margin: 0;
            padding: 0;
            list-style: none;
        }

        ul li {
            margin-bottom: 5px;
        }

        section {
            margin-bottom: 50px;
        }

        section:last-child {
            margin-bottom: 0;
        }
    </style>
</head>

<body>
    <div class="container">
```

```
            <header>
              <center><h2>resume</h2></center>
                <h1>Ajay chauhan</h1>
                <p>c-72, Ahmedabad india | 1234567890 |
Ajaychauhan@email.com</p>
            </header>
            <section>
                <h2>Summary</h2>
                <p>Creative and people-oriented computer
scientist with up to 2 years of working with tech
startups.
                    Advanced knowledge of software
design principles and agile development principles. Led
and managed a
                    team of five in developing new
software from concept to delivery</p>
            </section>
            <section>
                <h2>Skills</h2>
                <ul>
                    <li>JavaScript</li>
                    <li>HTML/CSS</li>
                    <li>c++</li>
                    <li>C language</li>
                    <li>bootstrap</li>
                </ul>
            </section>
            <section>


            </section>
            <section>
                <h2>Education</h2>
                <h3>Bachelor of information
technology</h3>
```

```
                <p>SOU| India | pursuing
                </p>
                <h2>GSHB</h2>
                <h3>HSC</h3>
                <p>GVV| India | 2021-2022
                </p>
                <h2>GSHB</h2>
                <h3>SSC</h3>
                <p>GVV| India | 2019-2020
                </p>
            </section>
        </div>
    </body>

</html>
```

Output:-

resume

Ajay chauhan

c-72, Ahmedabad india | 1234567890 | Ajaychauhan@email.com

Summary

Creative and people-oriented computer scientist with up to 2 years of working with tech startups. Advanced knowledge of software design principles and agile development principles. Led and managed a team of five in developing new software from concept to delivery

Skills

JavaScript
HTML/CSS
c++
C language
bootstrap

Education

Bachelor of information technology

SOU| India | pursuing

GSHB

HSC

GVV| India | 2021-2022

GSHB

SSC

GVV| India | 2019-2020

23.

Aim:- Create an HTML Page containing the following Gray Layout using CSS.

Code:-

(i)

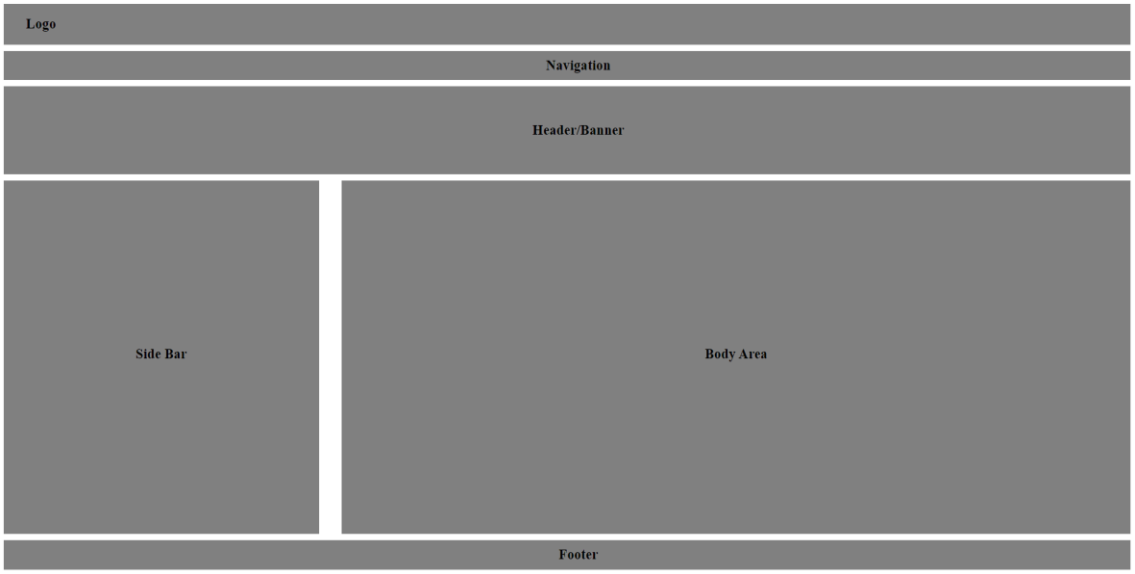
```
<!DOCTYPE html>
```

```
<head>
    <title>Gray Layout</title>
    <style>
        div {
            font-size: 25px;
            font-weight: bold;
        }

        #r1 {
            background-color: gray;
            height:
                50px;
            width: 98%;
            padding-top: 20px;
            padding-left: 2%;
            margin-bottom:
                10px;
        }

        #r2,
        #r5 {
            background-color: gray;
            height:
                40px;
            width: 98%;
            padding-top: 10px;
            padding-left: 2%;
            margin-bottom:
                10px;
            text-align: center;
        }

        #r3 {
            background-color: gray;
            height:
```

```
            90px;
        width: 98%;
        padding-top: 60px;
        padding-left: 2%;
        margin-bottom:
            10px;
        text-align: center;
    }

    #r4 {
        height: 600px;
        width: 100%;
        margin-bottom: 10px;
    }

    #r4 div {
        float: left;
        background-color:
            gray;
        padding-top: 280px;
        height: 320px;
        text-align: center;
    }

    #r4-c1 {
        width: 28%;
        margin-right: 2%;
    }

    #r4-c2 {
        width: 70%;
    }
</style>
</head>
```

```
<body>
    <div>
        <div id="r1">Logo</div>
        <div id="r2">Navigation</div>
        <div id="r3">Header/Banner </div>
        <div id="r4">
            <div id="r4-c1">Side Bar</div>
            <div id="r4-c2">Body Area</div>
        </div>
        <div id="r5">Footer</div>
    </div>
</body>

</html>
```

Output:-



(ii)

Code:-

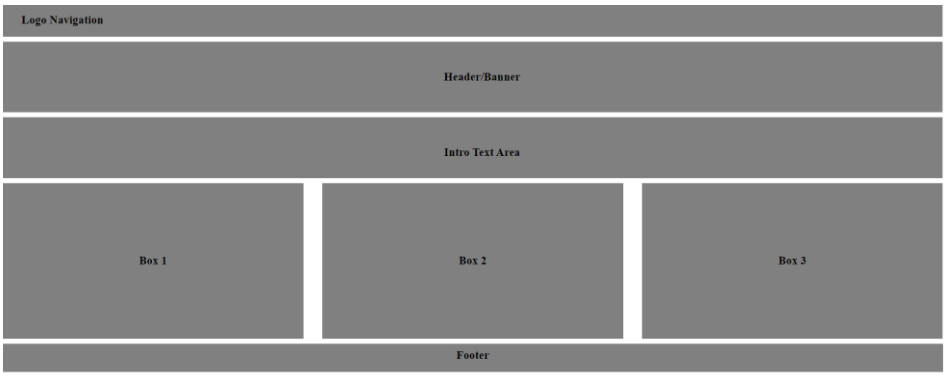
```
<!DOCTYPE html>

<head>
    <title>
```

```
        Gray Layout
    </title>
    <style>
        div {
            font-size: 25px;
            font-weight: bold;
        }

        #logo {
            background-color: gray;
            height: 50px;
            width: 98%;
            padding-top: 20px;
            padding-left: 2%;
            margin-bottom: 10px;
        }

        #Header {
            background-color: gray;
            height: 90px;
            width: 98%;
            padding-top: 60px;
            padding-left: 2%;
            margin-bottom: 10px;
            text-align: center;
        }

        #Intro {
            background-color: gray;
            height: 70px;
            width: 98%;
            padding-top: 60px;
            padding-left: 2%;
            margin-bottom: 10px;
            text-align: center;
```

```
}

#box {
    height: 330px;
    width: 100%;
    margin-bottom: 10px;
}

#box div {
    float: left;
    background-color: gray;
    padding-top: 150px;
    height: 180px;
    text-align: center;
}

#box1 {
    margin-right: 2%;
    width: 32%;
}

#box2 {
    margin-right: 2%;
    width: 32%;
}

#box3 {
    width: 32%;
}

#footer {
    background-color: gray;
    height: 50px;
    text-align: center;
    padding-top: 10px;
```

```
        }
    </style>
</head>

<body>
    <div>
        <div id="logo">Logo Navigation</div>
        <div id="Header">Header/Banner</div>
        <div id="Intro">Intro Text Area</div>
        <div id="box">
            <div id="box1">Box 1</div>
            <div id="box2">Box 2</div>
            <div id="box3">Box 3</div>
        </div>
        <div id="footer">Footer</div>
    </div>
</body>

</html
```

Output:-



(iii)

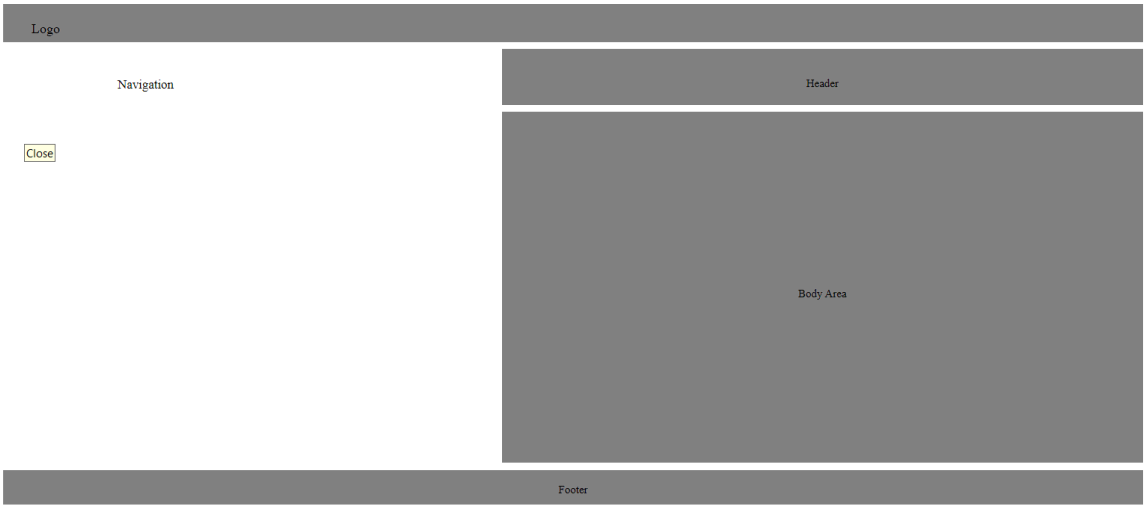
Code:-

```
<!DOCTYPE html>
<html lang="en">

    <head>
```

```
        <meta charset="UTF-8">
        <meta http-equiv="X-UA-Compatible"
content="IE=edge">
        <style>
            #r1 {
                padding-top: 30px;
                height: 30px;
                margin-top: 10px;
                margin-left: 10%;
                width: 78%;
                padding-left: 2%;
                background-color: gray;
                font-size: larger;
            }

            #r2 div {
                float: left;
                margin-left: 10%;
                margin-top: 10px;
                height: 40px;
                padding-top: 40px;
                text-align:
                    center;
            }

            #r2c1 {
                width: 20%;
                margin-right: 5%;
                font-size: large;
            }

            #r2c2 {
                width: 45%;
                background-color: gray;
            }
```

```
            #r3 {
                margin-bottom: 10px;
                margin-left: 45%;
                width: 45%;
                height: 250px;
                padding-top: 250px;
                text-align: center;
                background-color: gray;
                margin-top: 100px;
            }

            #r4 {
                margin-left: 10%;
                width: 80%;
                text-align: center;
                padding-top: 20px;
                height: 30px;
                background-color: gray;
            }
        </style>
    </head>

    <body>
        <div id="r1">Logo</div>
        <div id="r2">
            <div id="r2c1">Navigation</div>
            <div id="r2c2">Header</div>
        </div>
        <div id="r3">Body Area</div>
        <div id="r4">Footer</div>
    </body>

</html>
```

Output:-



(iv)

Code:-

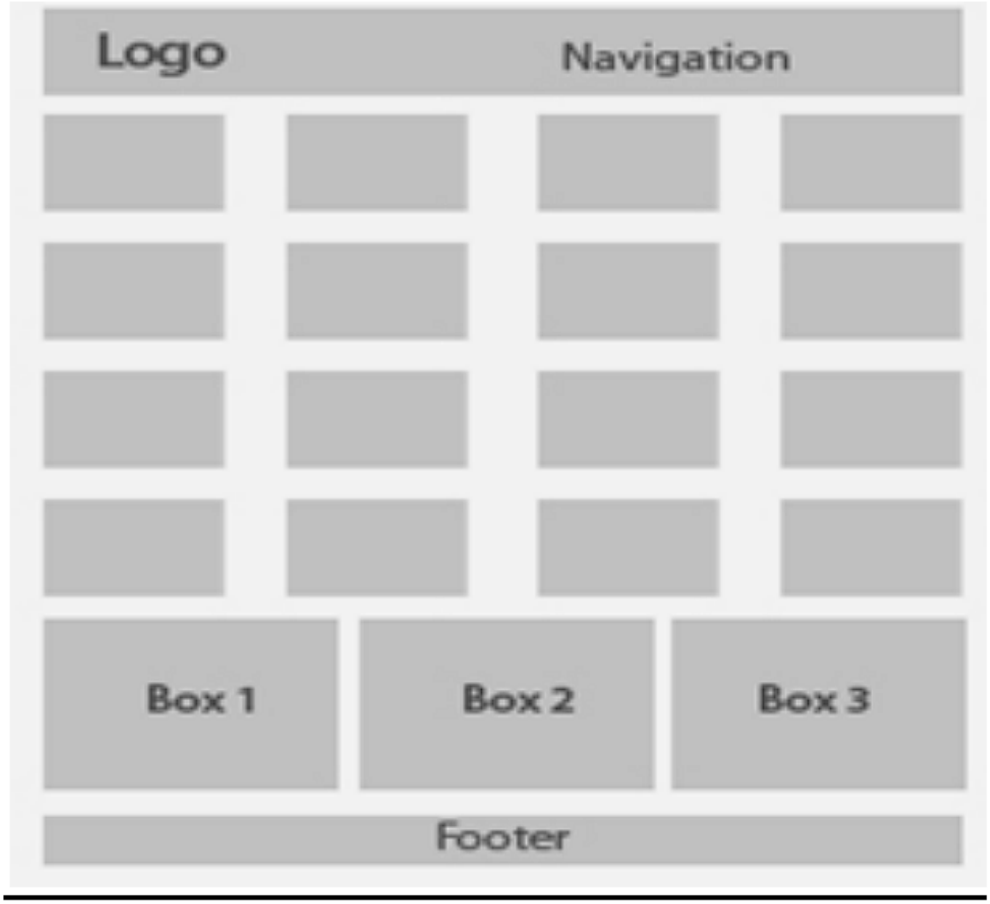
```
<!DOCTYPE html>
<html lang="en">

    <head>
        <title>Gray Layout 4</title>
        <style>
            #r1 {
                background-color: gray;
                width: 78%;
                padding-top: 15px;
                padding-left: 2%;
                margin-left: 10%;
                height: 40px;
            }

            #r1c1 {
                float: left;
                font-size: larger;
            }

            #r1c2 {
                margin-left: 80%;
                padding-top: 5px;
            }

            #r3 {
                margin-bottom: 10px;
                height: 100px;
                margin-left: 10%;
                margin-top: 10px;
```

```
}

#c1,
#c2,
#c3 {
    float: left;
    background-color: gray;
    width: 10%;
    margin-right: 15%;
    text-align:
        center;
}

#c5,
#c6,
#c7 {
    float: left;
    background-color: gray;
    width: 10%;
    margin-right: 15%;
    text-align: center;
}

#c9,
#c10,
#c11 {
    float: left;
    background-color: gray;
    width: 10%;
    margin-right: 15%;
    text-align:
        center;
}

#c13,
#c14,
#c15 {
    float: left;
    background-color: gray;
    width: 10%;
    margin-right: 15%;
    text-align:
        center;
}

#r4 div {
    float: left;
    background-color: gray;
```

```
            font-weight: bolder;
        }

        #r4c1 {
            margin-left: 10%;
            text-align: center;
            width: 25%;
            height: 100px;
            margin-top: 10px;
            padding-top: 100px;
        }

        #r4c2 {
            margin-left: 2.5%;
            text-align: center;
            width: 25%;
            height: 100px;
            margin-top: 10px;
            padding-top: 100px;
        }

        #r4c3 {
            margin-left: 2.5%;
            text-align: center;
            width: 25%;
            height: 100px;
            margin-top: 10px;
            padding-top: 100px;
        }

        #r5 {
            width: 80%;
            margin-left: 10%;
            background-color: graY;
            margin-top: 250px;
            height:
                30px;
            padding-top: 20px;
            text-align: center;
        }
    </style>
</head>

<body>
    <div id="r1">
        <div id="r1c1">Logo</div>
        <div id="r1c2">Navigation</div>
    </div>
```

```
        <div>
            <div></div>
        </div>
        <div id="r3">
            <div id="c1">1</div>
            <div id="c2">2</div>
            <div id="c3">3</div>
            <div id="c4">4</div><br>
            <div id="c5">5</div>
            <div id="c6">6</div>
            <div id="c7">7</div>
            <div id="c8">8</div><br>
            <div id="c9">9</div>
            <div id="c10">10</div>
            <div id="c11">11</div>
            <div id="c12">12</div><br>
            <div id="c13">13</div>
            <div id="c14">14</div>
            <div id="c15">15</div>
            <div id="c16">16</div>
        </div>
        <div id="r4">
            <div id="r4c1">Box 1</div>
            <div id="r4c2">Box 2</div>
            <div id="r4c3">Box 3</div>
        </div>
        <div id="r5">Footer</div>
    </body>

</html>
```

Output:-



24.

Aim:- Demonstrate JavaScript Form Validation with proper examples

Code:-

```
<!DOCTYPE html>
<html>

    <head>
        <title>Basic Form Validation</title>
        <script>
            function validateForm() {
                var name = document.forms["myForm"]["name"].value;
                var email = document.forms["myForm"]["email"].value;
                var password = document.forms["myForm"]["password"].value;
                if (name == "" || email == "" || password == "") {
                    alert("Please fill out all fields");
                    return false;
                }
            }
        </script>
    </head>
```

```
    <body>
        <form name="myForm" onsubmit="return validateForm()">
            <label for="name">Name:</label>
            <input type="text" id="name" name="name">
            <br><br>
            <label for="email">Email:</label>
            <input type="email" id="email" name="email">
            <br><br>
            <label for="password">Password:</label>
            <input type="password" id="password" name="password">
            <br><br>
            <input type="submit" value="Submit">
        </form>
    </body>

</html
```

Output:-

25.

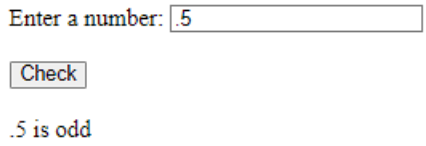
Aim:- : Write a javascript to check if the number is even or odd.

Code:-

```
<!DOCTYPE html>
<html>

    <head>
        <title>Even or Odd Checker</title>
        <script>
            function checkEvenOrOdd() {
                var number = document.getElementById("number").value;
                if (number % 2 == 0) {
                    document.getElementById("result").innerHTML = number + "
is even";
                } else {
                    document.getElementById("result").innerHTML = number + "
is odd";
                }
            }
        </script>
```

```
    </head>

    <body>
        <label for="number">Enter a number:</label>
        <input type="number" id="number">
        <br><br>
        <button onclick="checkEvenOrOdd()">Check</button>
        <br><br>
        <div id="result"></div>
    </body>

</html>
```

Output:-



26.

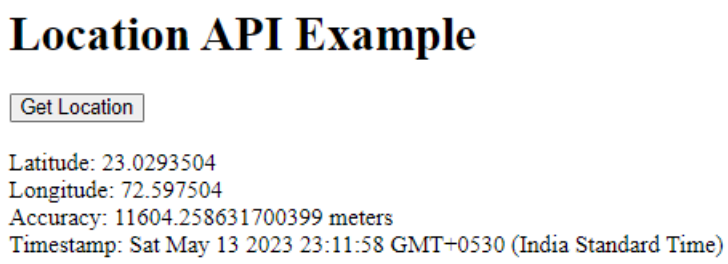
Aim:- Create a page and access the LocationAPI

Code:-

```
<!DOCTYPE html>
<html>

    <head>
        <title>Location API Example</title>
        <script>
            function getLocation() {
                if (navigator.geolocation) {
                    navigator.geolocation.getCurrentPosition(showPosition);
                }
                else {
                    alert("Geolocation is not supported by this browser.");
}
            }
            function showPosition(position) {
                var latitude = position.coords.latitude;
                var longitude = position.coords.longitude;
                var accuracy = position.coords.accuracy;
                var timestamp = new Date(position.timestamp);
                document.getElementById("latitude").innerHTML = "Latitude: " +
latitude;
                document.getElementById("longitude").innerHTML =
                    "Longitude: " + longitude;
                document.getElementById("accuracy").innerHTML = "Accuracy: "
```

```
                + accuracy + " meters";
                document.getElementById("timestamp").innerHTML =
                    "Timestamp: " + timestamp;
            }
        </script>
    </head>

    <body>
        <h1>Location API Example</h1>
        <button onclick="getLocation()">Get Location</button>
        <br><br>
        <div id="latitude"></div>
        <div id="longitude"></div>
        <div id="accuracy"></div>
        <div id="timestamp"></div>
    </body>

</html>
```

Output:-



27.

Aim:-

Create a simple XMLHTTPRequest,and retrieve the data from the text file.

Code:-

```
<!DOCTYPE html>
<html>

    <head>
        <title>XMLHTTPRequest Example</title>
        <script>
            function loadData() {
                var xhr = new XMLHttpRequest();
                xhr.open('GET', 'example.txt');
                xhr.onreadystatechange = function () {
                    if (xhr.readyState === XMLHttpRequest.DONE) {
```

```
                        if (xhr.status === 200) {
                            var response = xhr.responseText;
                            document.getElementById('output').innerHTML =
response;
                        }
                        else {
                            document.getElementById('output').innerHTML =
'Error: ' + xhr.status;
                        }
                    }
                };
                xhr.send();
            }
        </script>
    </head>

    <body>
        <h1>XMLHTTPRequest Example</h1>
        <button onclick="loadData()">Load Data</button>
        <br><br>
        <div id="output"></div>
    </body>

</html>
```

Output:-