SILVER OAK UNIVERSITY
EDUCATION TO INNOVATION

Name : Sapara Soham Rajeshbhai

Enrollmentno: 2202031000137

Subject: OJT Practicals

Course: B.Tech(IT)

Semester: 2nd

NAM

Practical - 1

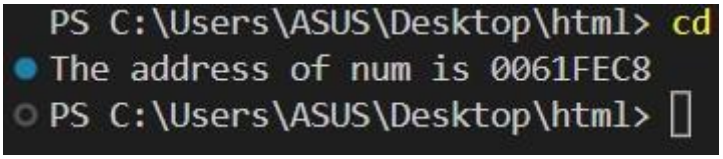
Aim:

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

Code:

```
#include<stdio.h>

int main(){
    int num = 50; int*ptr = &num; printf("The
    address of num is: %p\n",ptr);
    return 0;
}
```

Output:



Practical - 2

Aim:

Write a C program to create a Calculator using a pointer.

Code:

```
#include <stdio.h>

int main() { float num1, num2, *ptr1,
*ptr2; char operator;

printf("Enter two numbers: ");
scanf("%f %f", &num1, &num2);

ptr1 = &num1;
ptr2 = &num2;

printf("Enter an operator (+, -, *, /): ");
scanf(" %c", &operator);

switch(operator) { case '+': printf("%.2f + %.2f = %.2f",
*ptr1, *ptr2, *ptr1 + *ptr2); break; case '-':
printf("%.2f - %.2f = %.2f", *ptr1, *ptr2, *ptr1 - *ptr2);
break; case '*': printf("%.2f * %.2f = %.2f", *ptr1,
*ptr2, *ptr1 * *ptr2); break; case '/':
printf("%.2f / %.2f = %.2f", *ptr1, *ptr2, *ptr1 / *ptr2);
break; default: printf("Error: Invalid operator"); break;
return 0;
}
```

Output:

```
Enter two numbers: 10
20
Enter an operator (+, -, *, /): +
10.00 + 20.00 = 30.00
```

Practical - 3

Aim:

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

Code:

```
#include <stdio.h>

// function declaration for swap by value
void swapByValue(int x, int y);

// function declaration for swap by reference
void swapByReference(int *x, int *y);

int main() {
   int a = 10, b = 20;

   // display the original values
   printf("Before swapping, a = %d and b = %d\n", a, b);

   // swap by value swapByValue(a, b); printf("After swap
   by value, a = %d and b = %d\n", a, b);
```

```
    // swap by reference
    swapByReference(&a, &b); printf("After
    swap by reference, a = %d and b = %d\n",
    a, b);

     return 0;
}

// function definition for swap by value
void swapByValue(int x, int y) { int
temp = x;
    x = y; y
    = temp;
}

// function definition for swap by reference
void swapByReference(int *x, int *y) { int
temp = *x; *x = *y;
    *y = temp;
}
```

Output:

```
Before swapping, a = 10 and b = 20
After swap by value, a = 10 and b = 20
After swap by reference, a = 20 and b = 10
```

Practical - 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.

Code:

```
#include<stdio.h>
#include<string.h>
struct personal
{ char name[50];
    char DOB[50];
    int age;
}; void
 main()
{ struct personal p1,p2,p3,p4;
    strcpy( p1.name,"hardik");
    strcpy( p1.DOB,"30-01-2005");
    p1.age=18;

   strcpy( p2.name,”aniket");
   strcpy( p2.DOB,"15-06-2005");
   p2.age=17;

   strcpy( p3.name,"tith"); strcpy(
   p3.DOB,"22-11-2004");
   p3.age=18;

   strcpy( p4.name,"het");
  strcpy( p4.DOB,"09-09-2003");

  p4.age=18;
```

```
    printf("Person1 name: %s\n",p1.name);
    printf("Person1 DOB: %s\n",p1.DOB);
    printf("Person1 age: %d\n",p1.age);

    printf("Person2 name: %s\n",p2.name);
    printf("Person2 DOB: %s\n",p2.DOB);
    printf("Person2 age: %d\n",p2.age);

    printf("Person3 name: %s\n",p3.name);
    printf("Person3 DOB: %s\n",p3.DOB);
    printf("Person3 age: %d\n",p3.age);

    printf("Person4 name: %s\n",p4.name);
    printf("Person4 DOB: %s\n",p4.DOB);
    printf("Person4 age: %d\n",p4.age);
}
```

Output:

```
Person1 name: hardik
Person1 DOB: 30-01-2005
Person1 age: 18
Person2 DOB: 15-06-2005
Person2 age: 17
Person3 name: tirth
Person3 DOB: 22-11-2004
Person3 age: 18
Person4 name: het
Person4 DOB: 09-09-2003
Person4 age: 18
PS C:\Users\ASUS\Desktop\html>
```

Practical -

5

Aim:

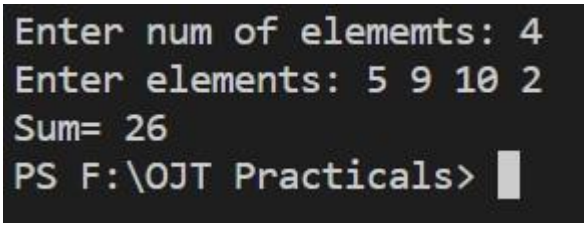
Write a C program to calculate the sum of n numbers entered by the user using dynamic memory allocation.

Code:

```
#include<stdio.h>
int main()
{ int i,n,*ptr, sum=0; printf("Enter
    num of elements: ");
    scanf("%d",&n);
    ptr=(int*) malloc(n*sizeof(int));
   //if memory cannot be allocated
   if(ptr==NULL)
   { printf("Error! memory not allocated");
    exit(0); }
    printf("Enter elements: ");
    for(i=0;i<n;++i)
    { scanf("%d",ptr+i);
        sum+=*(ptr+i);
     }
     printf("Sum= %d",sum);
    //deallocating the memory
   free(ptr);
return 0; }
```

Output:

Practical -


Enter num of elememts: 4
Enter elements: 5 9 10 2
Sum= 26
PS F:\OJT Practicals>

Practical -

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”. Then display the values of files odd and even on the screen.

Code:

```
#include<stdio.h>
int main()
{
    FILE *f1,*f2,*f3;
    int n,i;
    printf("\nWrite the numbers in file.\n Enter -1 to
    stop.\n\n"); f1=fopen("New","w"); for(i=1;i<=10;i++)
{ scanf("%d",&n);
    if(n== -1) break;
   putw(n,f1);
}
fclose(f1);
f1=fopen("New","r");
f2=fopen("ODD","w");
f3=fopen("EVEN","w");
while((n=getw(f1)) !=EOF) {
  if(n%2==0)
       putw(n,f3);
  else
       putw(n,f2)
       ;
}
```

Practical -

```
        fclose(f1);
        fclose(f2);
        fclose(f3);

        f2=fopen("ODD","r");
        f3=fopen("EVEN","r");
        printf("\n\nContents of ODD file\n\n");

      while((n=getw(f2)) != EOF)
            printf("%d\t",n);
       printf("\n\nContents of EVEN file\n\n");

        while((n=getw(f3))
      !=EOF) printf("%d\t",n);
      fclose(f2); fclose(f3); return
      0;
  }
```

Output:

```
Write the numbers in file.
 Enter -1 to stop.

0 1 2 3 4 5 6 7 8 9 -1


Contents of ODD file

1       3       5       7       9

Contents of EVEN file

0       2       4       6       8
PS F:\C\File>
```

7

Aim:

Practical -

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

Code:

```
#include <iostream>
using namespace std;
int main()
{ int n, i, m=0, flag=0; cout << "Enter the Number
      to check Prime: "; cin >> n; m=n/2; for(i = 2;
      i <= m; i++)
      {
         if(n % i == 0)
          { cout<<"Number is not Prime."<<endl;
                flag=1;
                break;
          }
      }
      if (flag==0)
            cout << "Number is Prime."<<endl;
      return 0;
}
```

Output:

```
Enter the Number to check Prime: 12
Number is not Prime.
PS F:\OJT Practicals> cd "f:\OJT Practic
Enter the Number to check Prime: 5
Number is Prime.
PS F:\OJT Practicals>
```

Practical -

8

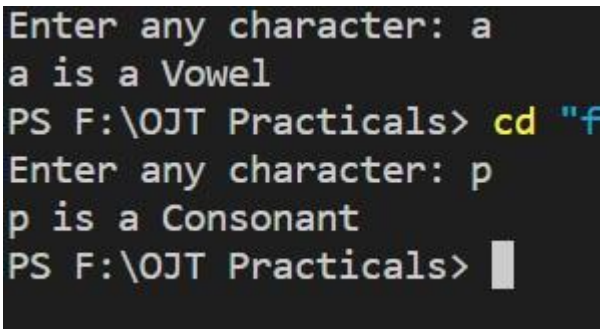
Aim:

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

Code:

```
#include <iostream>
using namespace std;
int main() { char c;
    cout<<"Enter any character: ";
    cin>>c;
    if (c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u'
    ) cout <<c<< " is a Vowel" << endl; else
        cout <<c<< " is a Consonant" << endl;
   return 0;
}
```

Output:



Practical - 9

Aim:

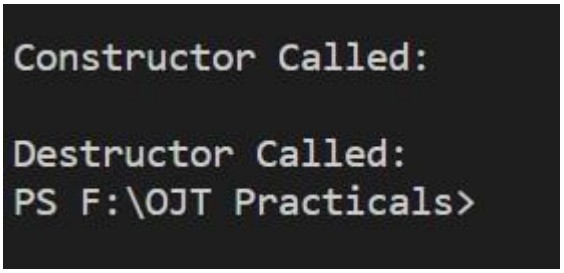
Write a C++ program to demonstrate the concept of constructor and destructor.

Practical -

Code:

```
#include<iostream>
using namespace std;
class rectangle
{ int length,width; public:
      rectangle()     //Constructor
            { length=0;
                width=0;
               cout<<"\nConstructor Called: ";
             }
            ~rectangle()   //Destructor
             { cout<<"\n\nDestructor Called: ";
             }
       //other functions for reading, writing and processing can be written here
 }; int
 main()
{ rectangle x;

  //default constructor is called
}
```

Output:



Practical - 10

Aim:

Create a class student that stores roll_no, 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.

Code:

```
#include <iostream>
#include <string>
using namespace std;
class Student {
                    int roll_no;
                    string name;
                    public:
                        void getStudentInfo() { cout <<
                                "Enter roll number: "; cin >>
                                roll_no; cout << "Enter name:
                                "; cin>>name;
                       } void displayStudentInfo() { cout << "Roll
                      number: " << roll_no << endl; cout << "Name: " <<
                      name << endl;
                       }
  }; class Test { int
marks[5]; public:
            void getTestMarks() { cout << "Enter marks obtained
                    in 5 subjects:\n";

                   for (int i = 0; i < 5; i++) { cout <<
                          "Subject " << i+1 << ": ";
                          cin >> marks[i];
                   }
```

```
} void displayTestMarks() { cout
<< "Marks obtained:\n";
        for (int i = 0; i < 5; i++) { cout << "Subject " << i+1 << ":
        " << marks[i] << endl; }
  } int
getTotalMarks() {
int total = 0;
    for (int i = 0; i < 5; i++) {
        total += marks[i];
    }
 return total;
} float
getPercentage() {
    return (getTotalMarks() / 5.0);
}
}; class Result : public Student,
public Test { public:
 void getResult() {
        getStudentInfo();
        getTestMarks();
}
void displayResult() {
       displayStudentInfo();
      displayTestMarks();



    cout << "Total marks: " << getTotalMarks() << endl; cout
    << "Percentage: " << getPercentage() << "%" << endl; }
}; int main() {
Result r;
    r.getResult();
```

```
    r.displayResult();
    return 0;
}
```

Output:

```
Enter marks obtained in 5 subjects:
Subject 1: 80
Subject 2: 90
Subject 3: 58
Subject 4: 99
Subject 5: 45
Roll number: 112
Name: HARDIK
Marks obtained:
Subject 1: 80
Subject 2: 90
Subject 3: 58
Subject 4: 99
Subject 5: 45
Total marks: 372
```

Practical Aim:

11

Write a C++ program to overload binary + operator.

Code:

```
#include<iostream>
using namespace std;

class complex
{ int real,imag;
  public:
  complex()
    { real=0; imag=0; }
    complex(int x,int
    y)
    { real=x;
    imag=y; }
    void
    disp()
    { cout<<"\nReal Value= "<<real<<endl;
      cout<<"\nImag Value=
      "<<imag<<endl;
    }
    complex operator + (complex);
};
complex complex::operator +(complex c)
{ complex tmp;
  tmp.real = this->real + c.real;    //this is operator
tmp.imag = this->imag + c.imag;  //this is operator
return tmp; }
```

Practical Aim:

```
int main()
{ complex c1(4,6),c2(7,9);
   complex c3;
   c3 = c1 + c2;        //(c1 is calling object) (+ is userdefine
   operator) c1.disp(); c2.disp(); c3.disp(); return 0;
}
```

Output:

```
Real Value= 4

Imag Value= 6

Real Value= 7

Imag Value= 9

Real Value= 11

Imag Value= 15
PS F:\C++\Overloading>
```

12

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:

Practical Aim:

```
#include<iostream>
#include<stdlib.h>
using namespace std;
class shape
{ protected:
      double x,y;
  public:
      void getDataofTriangle() {cin>>x>>y;}
      void getDataofRectangle() {cin>>x>>y;}
      virtual float calculateArea()=0;
 }; class Triangle:public
shape
{
   public:
   double calculateArea() {return 0.5*x*y;}
};



class rectangle:public shape
{
    public:
         double calculateArea() {return x*y;}
 }; int
 main()
{ system("cls");
     triangle t;
     rectangle r;
     cout<<"Enter height and base to calculate the area of Triangle: ";
     t.getDataofTriangle();
    cout<<"Area of triangle: "<<t.calculateArea()<<endl;
```

Practical Aim:

```
        cout<<endl<<"Enter length and width to calculate the area of rectangle: ";
        r.getDataofRectangle();
        cout<<"Area of rectangle: "<<r.calculateArea();
        return 0;
    }
```

Output:

```
Enter height and base to calculate the area of triangle: 5 8
Area of triangle: 20

Enter length and width to calculate the area of rectangle: 6 4
Area of rectangle: 24
PS F:\OJT Practicals>
```

13

To study DDL-create and DML-insert commands. Create following Tablea Job (job_id, job_title, min_sal, max_sal)

Code:

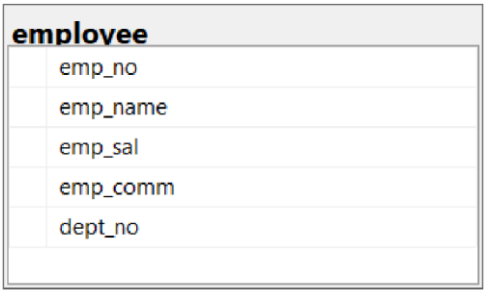
```
create table employee(
                    emp_no int, emp_name
                    varchar(30), emp_sal
                    decimal(8,2),
                    emp_comm
                    decimal(6,1),
                   dept_no int
              );

insert into employee(emp_no,emp_name,emp_sal,dept_no)
                values (101,'Smith',800,20);
```

Practical Aim:

SILVER OAK UNIVERSITY
EDUCATION TO INNOVATION

```
insert into
employee(emp_no,emp_name,emp_sal,emp_comm,dept_no) values
(102,'Snehal',1600,300,25), (103,'Adama',1100,0,20);

 insert into employee(emp_no,emp_name,emp_sal,dept_no)
 values (104,'Aman',3000,15);

insert into employee(emp_no,emp_name,emp_sal,emp_comm,dept_no)
values (105,'Anita',5000,50000,10), (106,'Sneha',2450,24500,10);

insert into employee(emp_no,emp_name,emp_sal,dept_no)
values (107,'Anamika',2975,30);
```

Output:



	emp_no	emp_name	emp_sal	emp_comm	dept_no
1	101	Smith	800.00	NULL	20
2	102	Snehal	1600.00	300.0	25
3	103	Adama	1100.00	0.0	20
4	104	Aman	3000.00	NULL	15
5	105	Anita	5000.00	50000.0	10
6	106	Sneha	2450.00	24500.0	10
7	107	Anamika	2975.00	NULL	30

Practical-14

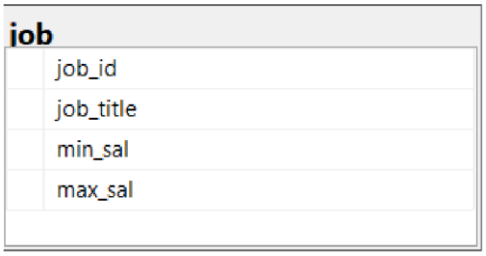
Aim:

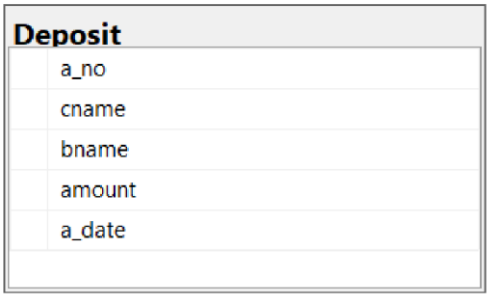
write a query to create job,employee, deposit and borrow table.

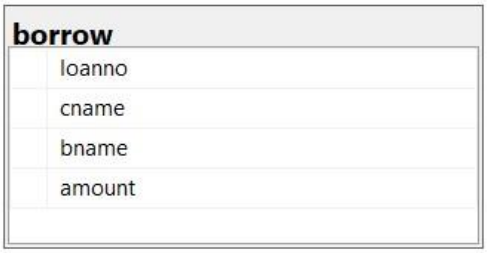
Code:

For job table
```
create table job(
    job_id varchar(15),
    job_title varchar(30),
        min_sal int,
        max_sal int
);
```
For emolyee table
```
create table employee(
    emp_no int, emp_name
    varchar(30), emp_sal
    decimal(8,2), emp_comm
    decimal(6,1),
        dept_no int
);
```
For deposit table
```
create table Deposit( a_no
        int identity(1,1),
        cname varchar(50),
        bname varchar(30),
        amount
        Decimal(4,2), a_date
        date
    );
```
For borrow table
```
create table borrow( loanno
    int, cname varchar(25), bname
    varchar(20), amount
    decimal(6,2)        );
```

Output:







Practical - 15

Aim:

write query to insert values in table employee, job and deposit.

Code:

For employee table

```
insert into employee(emp_no,emp_name,emp_sal,dept_no)
values (101,'Smith',800,20);

insert into
employee(emp_no,emp_name,emp_sal,emp_comm,dept_no) values
(102,'Snehal',1600,300,25), (103,'Adama',1100,0,20);

 insert into employee(emp_no,emp_name,emp_sal,dept_no)
 values (104,'Aman',3000,15);

insert into employee(emp_no,emp_name,emp_sal,emp_comm,dept_no)
values (105,'Anita',5000,50000,10), (106,'Sneha',2450,24500,10);
```

```
insert into employee(emp_no,emp_name,emp_sal,dept_no)
values (107,'Anamika',2975,30);
```

For job table

```
insert into job (job_id,job_title,min_sal,max_sal)
values ('IT PROG','Programmer',4000,10000),
       ('MK MGR','Marketing manager',9000,15000),
       ('FI MGR','Finance manager',8200,12000),
       ('FI ACC','Accountant',4200,9000),
       ('LEC','Lecturer',6000,17000),
       ('COMP OP','Computer Operator',1500,3000);
```

For deposit table

```
insert into Deposit(actno,cname,bname,amount,adate)
values(101,'Anil','andheri',7000,'01-jan-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');
```

Output:

	emp_no	emp_name	emp_sal	emp_comm	dept_no
1	101	Smith	800.00	NULL	20
2	102	Snehal	1600.00	300.0	25
3	103	Adama	1100.00	0.0	20
4	104	Aman	3000.00	NULL	15
5	105	Anita	5000.00	50000.0	10
6	106	Sneha	2450.00	24500.0	10
7	107	Anamika	2975.00	NULL	30

	job_id	job_title	min_sal	max_sal
1	IT PROG	Programmer	4000.00	10000.00
2	MK MGR	Marketing manager	9000.00	15000.00
3	FI MGR	Finance manager	8200.00	12000.00
4	FI ACC	Accountant	4200.00	9000.00
5	LEC	Lecturer	6000.00	17000.00
6	COMP OP	Computer Operator	1500.00	3000.00

	actno	cname	bname	amount	adate
1	101	Anil	andheri	7000.00	2006-01-01
2	102	Sunil	virar	5000.00	2006-07-15
3	103	Jay	villeparle	6500.00	2006-03-12
4	104	Vijay	andheri	8000.00	2006-09-17
5	105	Keyur	dadar	7500.00	2006-11-19
6	106	Mayur	borivali	5500.00	2006-12-21

Practical-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.

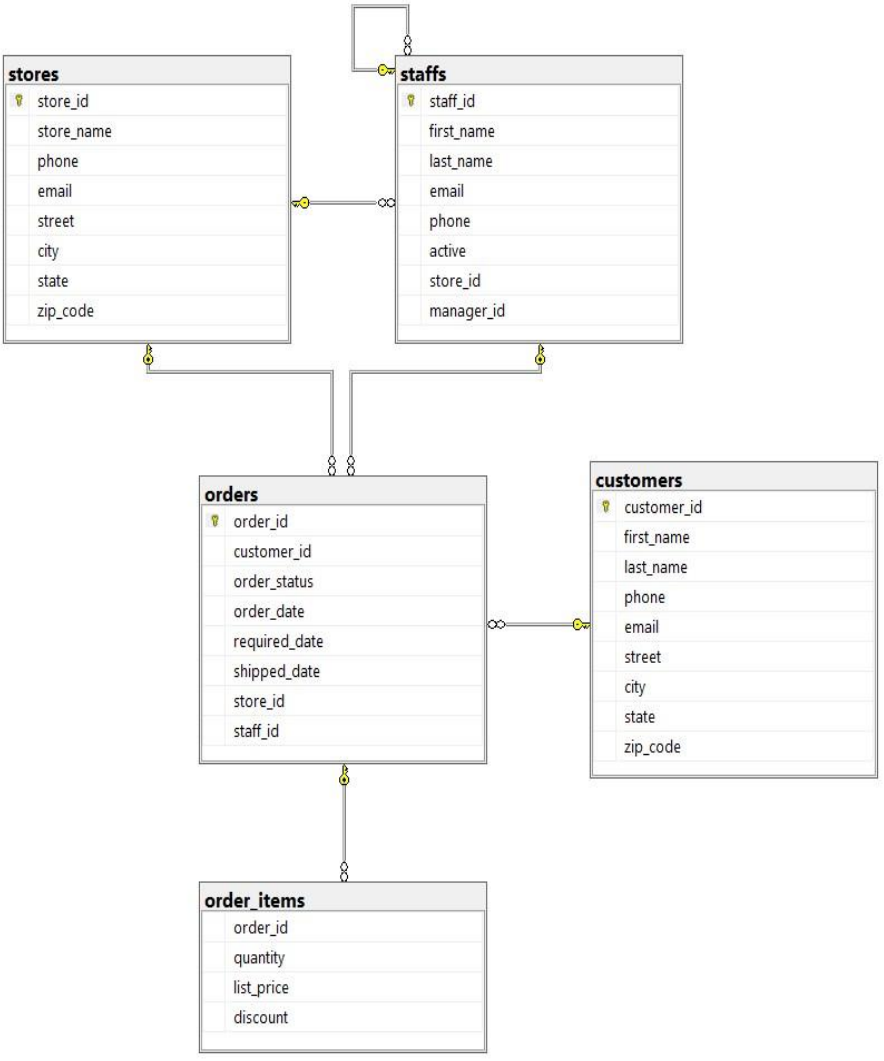
Code:

```
create table customers( customer_id int not null
    primary key identity(1,1), last_name varchar(50)
    not null , first_name varchar(50) not null , phone
    bigint not null , email varchar(50), street
    varchar(50), city varchar(50) not null , state
    varchar(50) not null , zip_code int
);
```

```
create table staffs( staff_id int not null primary key
        identity(1,1), first_name varchar(50) not null,

         last_name varchar(50) not null ,
         email varchar(50), phone bigint
         not null , active binary not null,
         store_id int foreign key references stores(store_id),
         manager_id int foreign key references staffs(staff_id)
 );

 create table stores( store_id int not null primary
        key identity(1,1), store_name varchar(50) not
        null, phone bigint not null, email varchar(50),
        street varchar(50), city varchar(50) not null ,
        state varchar(50) not null ,
        zip_code int
);

create table orders( order_id int not null primary key identity(1,1),
        customer_id int foreign key references customers(customer_id),
        order_status varchar(50) not null, order_date date not null,
        required_date date, shipped_date date not null,

         store_id int foreign key references stores(store_id),
         staff_id int foreign key references staffs(staff_id)
   );
```

```
create table order_items( order_id int foreign key
        references orders(order_id), quantity int not null,
        list_price int not null,
        discount int
);
```

Output:



Practical - 17

Aim:

Write the SQL queries to perform various aggregate functions on table data.

1. List total deposit from deposit.
2. List total amount from andheri branch
3. Count total number of customers
4. Count total number of customer's cities.
5. Update the value dept_no to 10 where second character of emp. name is 'm'.
6. Update the value of employee name whose employee number is 103.
7. Write a query to display the current date. Label the column Date
8. For each employee, display the employee number, salary, and salary increased by 15% and expressed as a whole number. Label the column New Salary
9. Modify your previous query to add a column that subtracts the old salary from the new salary. Label the column Increment.

Code:

1. select *from Deposit;
2. select sum(amount) from Deposit where

   bname='andheri';
3. select count(*) from deposit;
4. select bname,count(*) from deposit
   group by bname;
5. update employee set dept_no=10 where

   emp_name like '_m%';
6. update employee set emp_name='Pujan'

   where emp_no=103;
7. select GETDATE() as Date from
   employee;

8. alter table employee add new_sal varchar(50); update employee set new_sal=emp_sal+(emp_sal*15/100);
9. alter table employee add increment varchar(50); update employee set increment=new_sal-emp_sal;

Output:

	actno	cname	bname	amount	adate
1	101	Anil	andheri	7000.00	2006-01-01
2	102	Sunil	virar	5000.00	2006-07-15
3	103	Jay	villeparle	6500.00	2006-03-12
4	104	Vijay	andheri	8000.00	2006-09-17
5	105	Keyur	dadar	7500.00	2006-11-19
6	106	Mayur	borivali	5500.00	2006-12-21

	(No column name)
1	15000.00

	(No column name)
1	6

	bname	(No column name)
1	andheri	2
2	borivali	1
3	dadar	1
4	villepa...	1
5	virar	1

	Date
1	2023-03-19 22:56:50.937
2	2023-03-19 22:56:50.937
3	2023-03-19 22:56:50.937
4	2023-03-19 22:56:50.937
5	2023-03-19 22:56:50.937
6	2023-03-19 22:56:50.937
7	2023-03-19 22:56:50.937

	emp_no	emp_name	emp_sal	emp_comm	dept_no	new_sal	increment
1	101	Smith	800.00	NULL	10	920.0000000000000	120.00
2	102	Snehal	1600.00	300.0	25	1840.0000000000000	240.00
3	103	Pujan	1100.00	0.0	20	1265.0000000000000	165.00
4	104	Aman	3000.00	NULL	10	3450.0000000000000	450.00
5	105	Anita	5000.00	50000.0	10	5750.0000000000000	750.00
6	106	Sneha	2450.00	24500.0	10	2817.5000000000000	367.50
7	107	Anamika	2975.00	NULL	30	3421.2500000000000	446.25

Practical - 18

Aim:

Write the SQL queries to perform numeric, date and String functions.

1. Retrieve all data from employee, jobs and deposit.
2. Give details of account no. and deposited rupees of customers having account opened between dates 01-01-06 and 25-07-06.
3. Display all jobs with minimum salary is greater than 4000.
4. Display name and salary of employee whose department no is 20. Give alias name to name of employee.
5. Display employee no,name and department details of those employee whose department lies in(10,20)
6. Display all employee whose name start with ‘A’ and third character is ‘ ‘a’.
7. Display name, number and salary of those employees whose name is 5 characters long and first three characters are ‘Ani’.
8. Display the non-null values of employees and also employee name second charactershould be ‘n’ and string should be 5 character long.
9. Display the null values of employee and also employee name’s third character should be ‘a’.

Code:

1. select *from employee; select *from job; select *from deposit;
2. select actno,amount from Deposit where adate between '2006-01-01' and '2006-07-25';
3. select job_id from job where min_sal > 4000;
4. select emp_name as 'name of employee',emp_sal from employee where dept_no=20 ;
5. select emp_no,emp_name,dept_no from employee where dept_no between 10 and 20;

6. select emp_name from employee where emp_name like 'A_a%';
7. select emp_name,emp_no,emp_sal from employee where len(emp_name)=5 and emp_name like 'Ani%';
8. select emp_name from employee where emp_comm is not null and emp_name like '_n%' and len(emp_name)=5;
9. select emp_name from employee where emp_comm is null and emp_name like '__a%';

Output:

	emp_no	emp_name	emp_sal	emp_comm	dept_no
1	102	Snehal	1600.00	300.0	25
2	103	Adama	1100.00	0.0	20
3	104	Aman	3000.00	NULL	15
4	105	Anita	5000.00	50000.0	10
5	106	Sneha	2450.00	24500.0	10
6	107	Anamika	2975.00	NULL	30

	job_id	job_title	min_sal	max_sal
1	IT PROG	Programmer	4000.00	10000.00
2	MK MGR	Marketing manager	9000.00	15000.00
3	FI MGR	Finance manager	8200.00	12000.00
4	FI ACC	Accountant	4200.00	9000.00
5	LEC	Lecturer	6000.00	17000.00
6	COMP OP	Computer Operator	1500.00	3000.00

	actno	cname	bname	amount	adate
1	101	Anil	andheri	7000.00	2006-01-01
2	102	Sunil	virar	5000.00	2006-07-15
3	103	Jay	villeparle	6500.00	2006-03-12
4	104	Vijay	andheri	8000.00	2006-09-17
5	105	Keyur	dadar	7500.00	2006-11-19
6	106	Mayur	borivali	5500.00	2006-12-21

	actno	amount
1	101	7000.00
2	102	5000.00
3	103	6500.00

	job_id
1	MK MGR
2	FI MGR
3	FI ACC
4	LEC

	name of employee	emp_sal
1	Adama	1100.00

	emp_no	emp_name	dept_no
1	103	Adama	20
2	104	Aman	15
3	105	Anita	10
4	106	Sneha	10

	emp_name
1	Adama
2	Aman
3	Anamika

	emp_name	emp_no	emp_sal
1	Anita	105	5000.00

	emp_name
1	Anita
2	Sneha

	emp_name
1	Aman
2	Anamika

Practical - 19

Aim:

Make a Resume using the HTML tags without CSS.

Code:

```
<html lang="en">
<head>
```

```
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>resume</title>
</head>
<body>
     <center><h1>RESUME</h1></center>


   <center> <img
src="
AAAAMFBMVEXk5ueutLfn6erGysyrsbTh4+TY29zU19m+w8XJzc+s" alt="" class="i">
    <div class="one">
        <h2>Hardik Rupavatiya</h2>
    </div></center>
    <h3><u>CONTACT</u></h3>
   <p class="p"><b>MOBILE : </b>9316181435</p>
   <p class="p"><b>EMAIL : </b>hardikrupavatiya511@gmail.com</p>



   <h3><u>OBJECTIVE</u></h3>
   <ul>
    <li><p>Looking for a suitable job role in Software Engineering and
Development with ABC Inc. to utilize 2+ years of experience in
technology engineering, software design, client support and servicing
and server maintenance.</p></li>
   </ul>


   <h3><u>EDUCATION</u></h3>
   <table border="collapse">
    <tr>
```

```
        <th>COURSE</th>
        <th>BOARD</th>
        <th>YEAR</th>
<th>MARKS</th>
    </tr>
```

```
    <tr>
        <td>SSC</td>
        <td>GSEB</td>
        <td>2020</td>
        <td>86%</td>
    </tr>
    <tr>
        <td>HSC</td>
        <td>GSEB</td>
        <td>2022</td>
        <td>56%</td>
    </tr>
   </table>
   <h3><u>SKILLS</u></h3>
   <h3>Technical Skills</h3>
   <ul>
    <li>C,C++</li>
    <li>HTML,CSS,JAVASCRIPT</li>
    <li>SQL</li>
    <li>VS STUDIO,WINDOWS</li>
   </ul>
   <h3>Soft Skills</h3>
   <ul>

    <li>ADAPTABILITY</li>
    <li>TEAM WORK</li>


   </ul>


   <h3><u>LANGUAGES</u></h3>
   <ul>
<li>GUJARATI</li>
```

```
    <li>HINDI</li>
    <li>ENGLISH</li>
   </ul>
   <h3><u>HOBBIES</u></h3>
  <ul>
    <li>TRAVEL</li>
    <li>reading</li>
  </ul>
 <h3><u>DECLARATION</u></h3>
 <p>I hereby declare that the details and information given above are
complete and true to the best of my knowledge</p>



</body>
</html>
```

Output:

SILVER OAK UNIVERSITY
EDUCATION TO INNOVATION

RESUME

Hardik Rupavatiya

ring and Development with ABC Inc. to utilize 2+ years of experience in technology engineering, software design, client support and servi

ve are complete and true to the best of my knowledge

Practical - 20

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

Code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Movie Poster Presentation</title>
    <style> table { border-collapse:
        collapse; width: 100%;
        } td, th { border: 1px solid
        black; padding: 8px; text-
        align: center;
        } th { background-color: #f2f2f2;
        font-weight: bold;
        } tr:nth-child(even) {
        background-color: #f2f2f2;
        } caption { font-size: 1.2em;
        font-weight: bold; margin-
        bottom: 10px;
        }
    </style>
</head>
<body>
    <table>
        <caption>Movie Poster Presentation</caption>
        <thead>
            <tr>
                <th>Poster</th>
                <th>Title</th>
                <th>Release Year</th>
                <th>Director</th>

<th>Actors</th>
                <th>Genre</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td><img width="250px" height="300px"
```

```
src="https://i.pinimg.com/originals/84/18/d7/8418d756cc9038b87815a4f2d4f2a16f.jpg" alt="Movie Poster"></td>
                        <td>The Dark Knight</td>
                        <td>2008</td>
                        <td>Christopher Nolan</td>
                        <td>Christian Bale, Heath Ledger, Aaron Eckhart</td>
                        <td>Action, Crime, Drama</td>
                    </tr>
                    <tr>
                        <td><img width="250px" height="300px"
src="https://w0.peakpx.com/wallpaper/1/639/HD-wallpaper-forrest-gump-love-movie.
jpg" alt="Movie Poster"></td>
                        <td>Forrest Gump</td>
                        <td>1994</td>
                        <td>Robert Zemeckis</td>
                        <td>Tom Hanks, Robin Wright, Gary Sinise</td>
                        <td>Drama, Romance</td>
                    </tr>
                    <tr>
                        <td><img width="250px" height="300px"
src="https://wallpaperaccess.com/full/2235018.jpg" alt="Movie Poster"></td>
                        <td>KGF2</td>
                        <td>2022</td>
                        <td>Prashanth Neel</td>
                        <td>Yash, Srinidhi Shetty, Sanjay Dutt, Raveena Tandon,
Prakash Raj, Ramachandra Raju</td>
                        <td>Action, Drama</td>
                    </tr>
                </tbody>
        </table>
</body>
</html>
```

Output:

Movie Poster Presentation

Poster	Title	Release Year	Director	Actors	Genre
	The Dark Knight	2008	Christopher Nolan	Christian Bale, Heath Ledger, Aaron Eckhart	Action, Crime, Drama
	Forrest Gump	1994	Robert Zemeckis	Tom Hanks, Robin Wright, Gary Sinise	Drama, Romance
	KGF2	2022	Prashanth Neel	Yash, Srinidhi Shetty, Sanjay Dutt, Raveena Tandon, Prakash Raj, Ramachandra Raju	Action, Drama

Practical-20

Aim: Create an HTML page table and form.

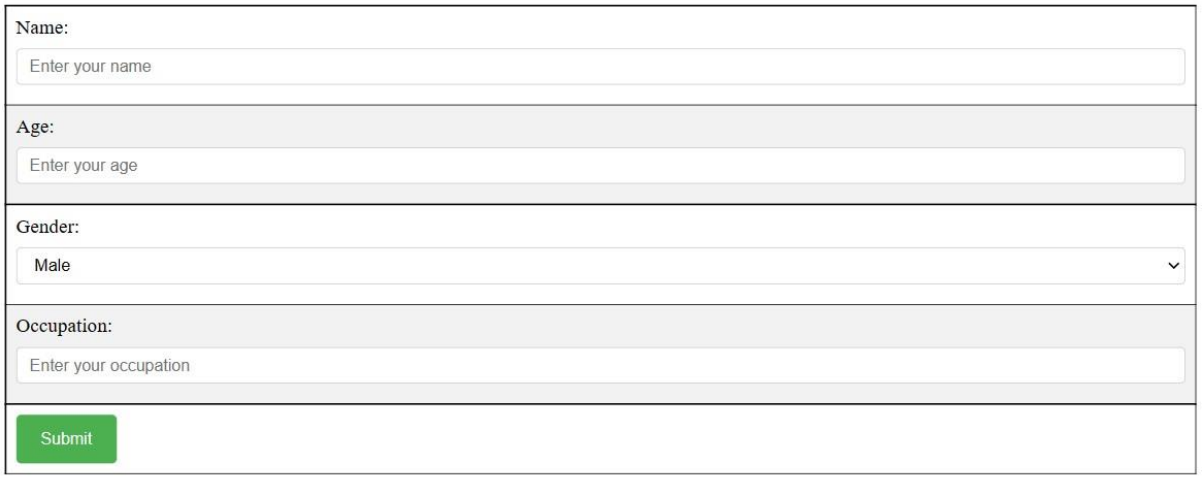
Code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Table and Form Example</title>
```

```
        <style> table { border-collapse: collapse;
                width: 100%; } td, th { border:
                1px solid black; padding: 8px; }
                th { background-color: #f2f2f2;
                font-weight: bold;
                } tr:nth-child(even) {
                background-color: #f2f2f2;
                } form { margin-top:
                20px;
                } label { display: block;
                margin-bottom: 8px;
                }
                input[type="text"], select { padding:
                6px 10px; border: 1px solid #ccc;
                border-radius: 4px; box-sizing:
                border-box; margin-bottom: 8px;
                width: 100%; }
                input[type="submit"] { background-
                color: #4CAF50;
                        color: white;
padding: 12px 20px;

                        border: none; border-
                        radius: 4px;
                        cursor: pointer;
                }
                input[type="submit"]:hover { background-
                        color: #45a049;
                }
        </style>
</head>
<body>


        <form>
      <table border="1" width="100%">
         <tr>
            <td><label for="name">Name:</label>
               <input type="text" id="name" name="name" placeholder="Enter your
name">
            </td></tr>
            <tr>
```

```
            <td>
              <label for="age">Age:</label>
              <input type="text" id="age" name="age" placeholder="Enter your age">
            </td>
            </tr>
            <tr>
            <td>
              <label for="gender">Gender:</label>
              <select id="gender" name="gender">
            <option value="male">Male</option>
            <option value="female">Female</option>
            <option value="other">Other</option>
          </select>
            </td>
          </tr>
          <tr>
            <td>
              <label for="occupation">Occupation:</label>
              <input type="text" id="occupation" name="occupation"
placeholder="Enter your occupation">
            </td>
          </tr>
            <tr><td><input type="submit" value="Submit"></td>
          </tr>
        </table>
          </form>
</body>
</html>
```

Output:



Practical-21

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

Code:

```
!DOCTYPE html>

<html>
<head>
	<title>Registration Form</title>
	<style> label { display: block;
		margin-bottom: 8px;
		}
		input[type="text"], input[type="email"], input[type="password"]
			{ padding: 6px 10px; border: 1px solid #ccc; border-
			radius: 4px; box-sizing: border-box; margin-bottom: 8px;
			width: 100%;
		}
		input[type="submit"] { background-
			color: #4CAF50;
			color: white;
			padding: 12px 20px;
			border: none;
			border-radius: 4px;
			cursor: pointer;
		}
	</style>
</head>
<body>
	<h1>Registration Form</h1>
	<form>
		<label for="name">Name:</label>
		<input type="text" id="name" name="name" placeholder="Enter your name"
required>

		<label for="email">Email:</label>
		<input type="email" id="email" name="email" placeholder="Enter your
email" required>

		<label for="password">Password:</label>
		<input type="password" id="password" name="password" placeholder="Enter
your password" pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,}" required>
```

```
            <small>Password must contain at least one number, one lowercase letter, one uppercase letter, and be at least 8 characters long.</small>

            <label for="confirm-password">Confirm Password:</label>
            <input type="password" id="confirm-password" name="confirm-password" placeholder="Confirm your password" pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,}" required>

            <input type="submit" value="Register">
        </form>
</body>
</html>
```

Output:

Practical-22

Aim: Make a Resume using the HTML tags with CSS.

Code:

```
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
```

```
initial-scale=1.0">
    <link rel="stylesheet" href="file.css">
</head>



<body>
    <div class="full">
        <div class="left">
            <div class="image">
                <img src="E:\CSE B Tech\Sem
1\photo_2022-2-01_16-59-45.jpg" alt="profile picture"
                    style="width:100px;height:100px;">
            </div>
            <div class="Contact">
                <h2>Contact</h2>
                <p><b>Email id:</b>hardikrupavatiya511@gmail.com</p>
                <p><b>Mobile no :</b>9316181435</p>
            </div>
            <div class="Skills">
```

```
                <h2>Skills</h2>
                <ul>
                    <li><b>Programming Languages :
                            C++,HTML,css,c</b></li>
                    </b></li>


                </ul>
```

```
        </div>
        <div class="Language">
            <h2>Language</h2>
            <ul>
                <li>English</li>
                <li>Hindi</li>
                <li>Gujarati</li>
            </ul>
        </div>
        <div class="Hobbies">
            <h2>Hobbies</h2>
            <ul>
                <li>Travelling</li>
                <li>Reading</li>
            </ul>
        </div>
    </div>
    <div class="right">
        <div class="name">
            <h1>Hardik Rupavatiya</h1>
        </div>
```

```
        <div class="title">
            <p>Student</p>

        </div>
```

```
<div class="Personal details">

    <h2>Personal details</h2>

    <p><b>DOB:</b>02 december 2004</p>

    <p><b>Gender:</b>Male</p>
</div>

  



<div class="Education">

    <h2>Education</h2>

    <table>

        <tr>
            <th>Qualification   </th>

            <th>Board    </th>

            <th>Passing year     </th>

            <th>percentage</th>

        </tr>




        <tr>
```

```
                <td>SSC</td>

                <td>GSEB</td>

                <td>2020</td>

                <td>86%</td>

            </tr>
```

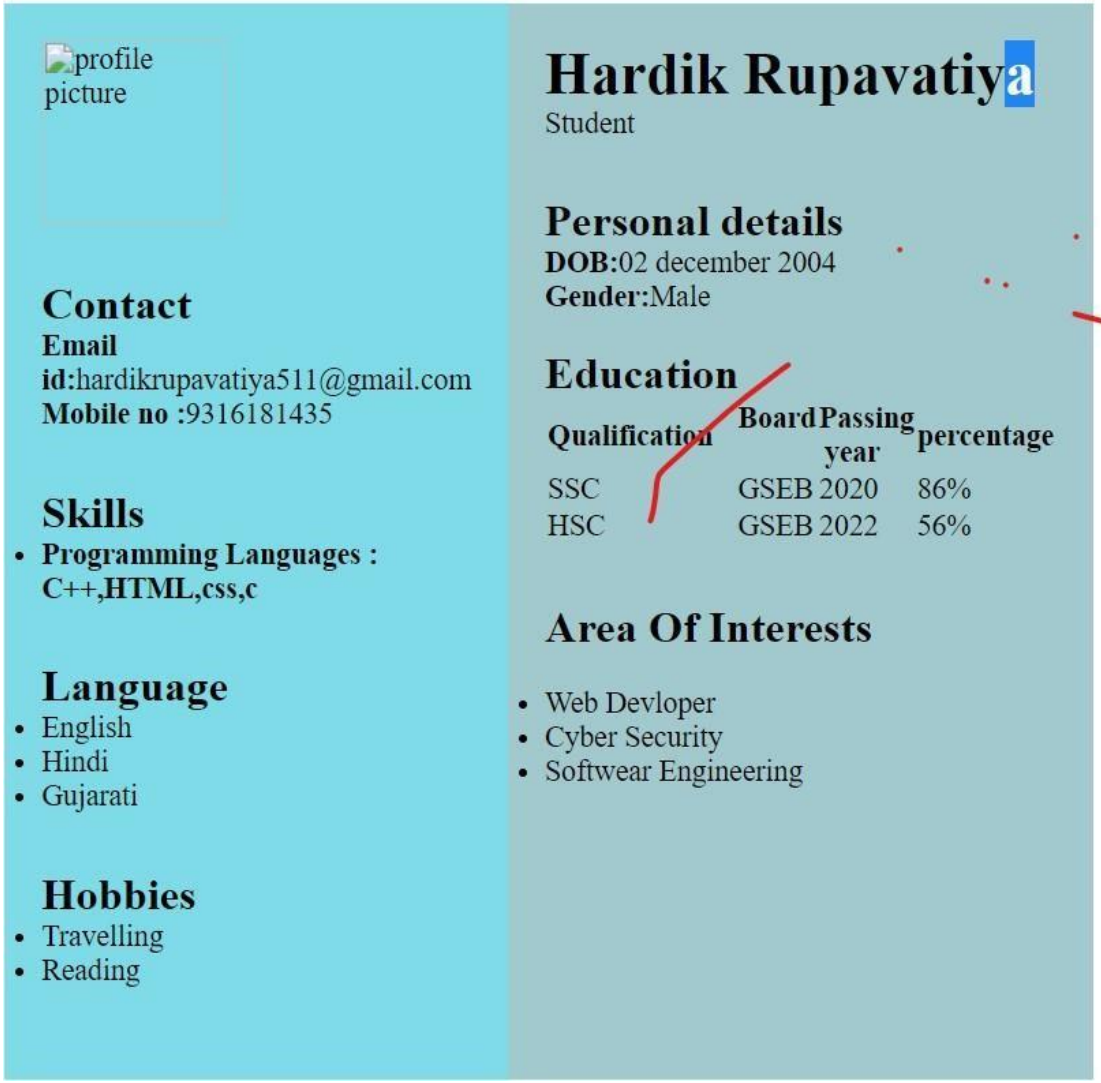
```
            <tr>
                <td>HSC</td>
                <td>GSEB</td>
                <td>2022</td>
                <td>56%</td>
            </tr>
        </table>
    </div>
    <div class=" Interests">
        <h2>Area Of Interests</h2>
        <ul>
              <li>Web Devloper</li>
            <li>Cyber Security</li>
            <li>Softwear Engineering</li>
        </ul>
        <div>
            </li>
            </ul>
        </div>
```

```
            </div>

        </div>

</body>



</html>

* {

    margin: 0;

    padding: 0;

    box-sizing: border-box;
```

```
}
body
{

    background-color: rgb(253, 254, 255);

    display: flex;

    justify-content: center;

    align-items: center;

}

.full { width:
    50%;


    max-width: 1000px;

    min-height: 100px;

    background-color: rgb(245, 239, 231);

    margin: 0px;

    display: grid;

    grid-template-columns: 2fr 4fr;

}

.left {
```

```
    position: initial;

    background-color: rgb(126, 219, 231);

    padding: 20px;



}
.right
{

    position: initial;

    background-color: rgb(162, 202, 206);

    padding: 20px;



}

.image, .Contact, .Skills, .Language, .Hobbies, .title,

 .Experience, .Education, .project {

    margin-bottom: 30px;

}

.h2 {

    background-color: rgb(4, 96, 150);

}
```

Output:



Practical-23

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

Code:

1.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <style type="text/css">
      * {
        font-size: 22px; font-
        weight: bold;
```

```
    }
    #r1 { background-color:
      gray; height: 50px;
      padding-top: 20px;
      width: 98%; padding-
      left: 2%; margin-
      bottom: 10px;
    }
    #r2,
    #r5 { background-color:
      gray; height: 33px;
      padding-top: 7px; /*
      width: 100%; */
      padding-left: 2%; text-
      align: center; margin-
      bottom: 10px;
    }
    #r3 { background-color:
      gray; height: 90px;
      width: 100%; padding-
      top: 60px; text-align:
      center; margin-bottom:
      10px;
    }
  #r4 { height: 600px;
      width: 100%;
      margin-bottom:
      10px;
    }
    #r4c1 { width:
      28%; margin-
      right: 2%;
    }
    #r4c2 {
      width: 70%;
    }
    #r4 div {
      float: left; height:
      320px; padding-top:
      280px; text-align:
      center; background-
      color: gray;
    }
  </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="r4c1">Side bar</div>
        <div id="r4c2">Body Area</div>
      </div>
      <div id="r5">Footer</div>
    </div>
  </body>
</html>
```

2.

<!DOCT

YPE

html>

```
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Document</title>
   <style type="text/css">
      *{ font-size: 22px;
         font-weight: bold;
}
      #r1{ background-color:
         gray; height: 50px;
         padding-top: 20px;
         width: 98%; padding-
         left: 2%; margin-
         bottom: 10px;
      }
      #r2,#r5{ background-
         color: gray; height:
         33px; padding-top: 7px;
         /* width: 100%; */
         padding-left: 2%; text-
         align: center; margin-
         bottom: 10px;
```

```
        }
        #r3{ background-color:
            gray; height: 90px;
            width: 100%; padding-
            top: 60px; text-align:
            center; margin-bottom:
            10px;
        }
        #r4{ height: 600px;
             width: 100%; margin-
             bottom: 10px;
        }
        #r4c1{ width: 32%;
            margin-right: 2%;
        }
        #r4c2{ width: 32%;
            margin-right: 2%;
        }
        #r4c3{ width:
            32%;
        }
        #r4 div{
            float: left; height:
            320px; padding-top:
            280px; text-align:
            center; background-
            color: gray;
    }
    </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="r4c1"> box-
              1
            </div>
```

```
        <div id="r4c2"> box-
           2
        </div> <div
        id="r4c3">
        box-3 </div>
      </div>
      <div id="r5">
        Footer
      </div>
   </div>
</body>
</html>
```

3.

```
<html>
<head>
   <title>Demo Layout 3</title>
   <style type="text/css">
      * {
         font-size: 22px; font-
         weight: bold;
      }

      #R1 {
         background-color: gray;
         height: 50px; padding-
         top: 20px
 width: 98%;
         padding-left: 2%; margin-
         bottom: 10px;
      }

      #R5 {
         background-color: gray;
      height: 33px; width:
      100%; padding-top: 7px;
      text-align: center; margin-
      bottom: 10px; margin-top:
      1%; }

      table { width:
         100%;
```

```
        }

        .s { height:
            600px; width:
            28%;
        }

        .b { background-color:
            gray; height: 100px;
            text-align: center;
        }

        .bl { height:
            10px;
        }

        .ba { background-color:
            gray; height: 490px;
            text-align: center;
        }
    </style>
</head>

<body>
    <div>
        <div id="R1">
            Logo
        </div>
        <table>
 <tr>
                <td rowspan="3" class="s">
                    Side Bar Navigation
                </td>
                <td class="b">
                    Header/Banner
                </td>
            </tr>
            <tr>
                <td>
                    <div class="bl"></div>
                </td>
            </tr>
            <tr>
                <td class="ba">
                    Body Area
                </td>
```

```
        </tr>
      </table>
      <div id="R5">
        Footer
      </div>
    </div>
</body>

</html>
```

4.

```
<html>
```

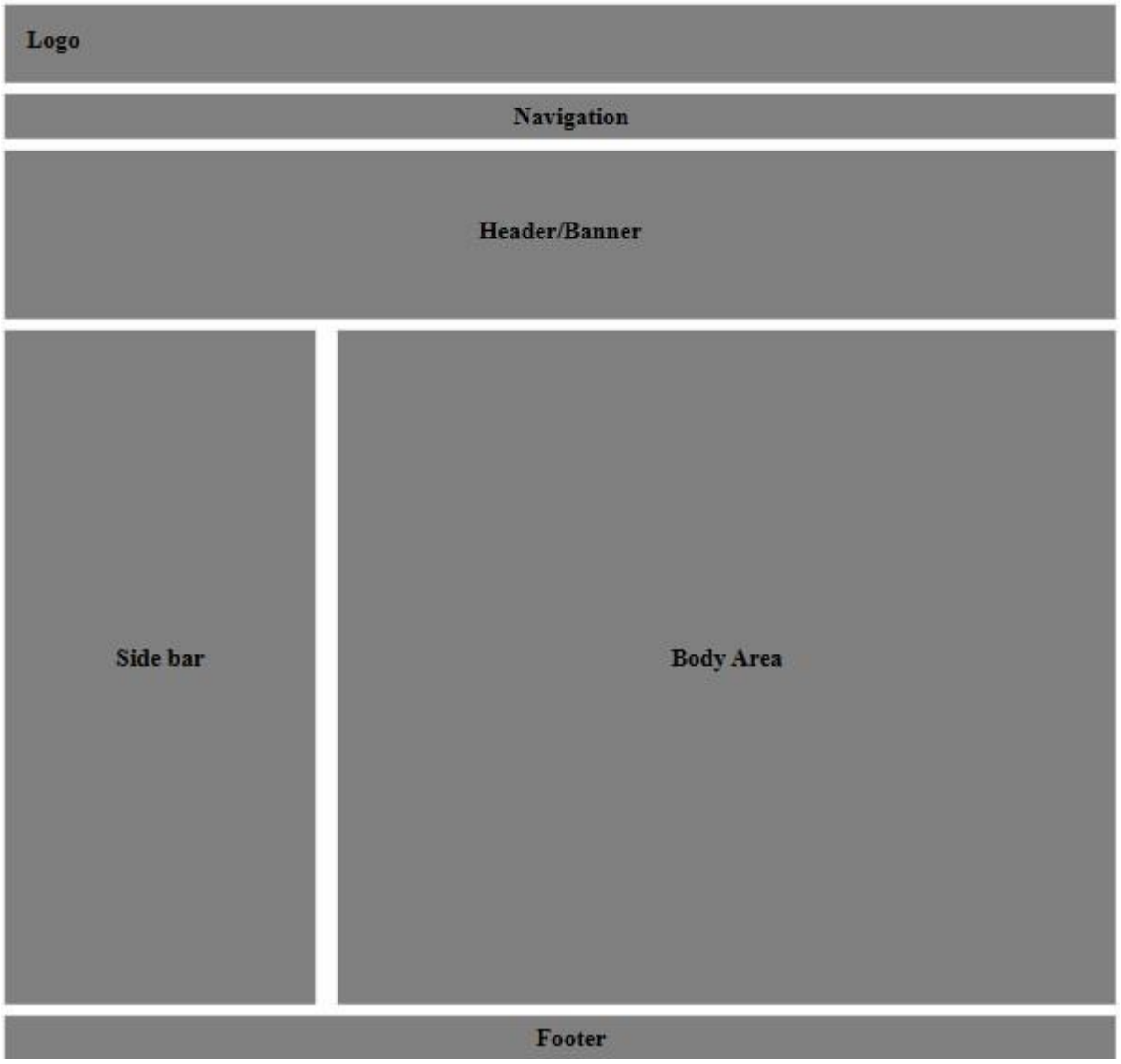
```
<head>
        <title>Demo Layout 1</title>
        <style type="text/css">
                * {
                        font-size: 22px; font-
                        weight: bold;
                }

                #R1 {
```

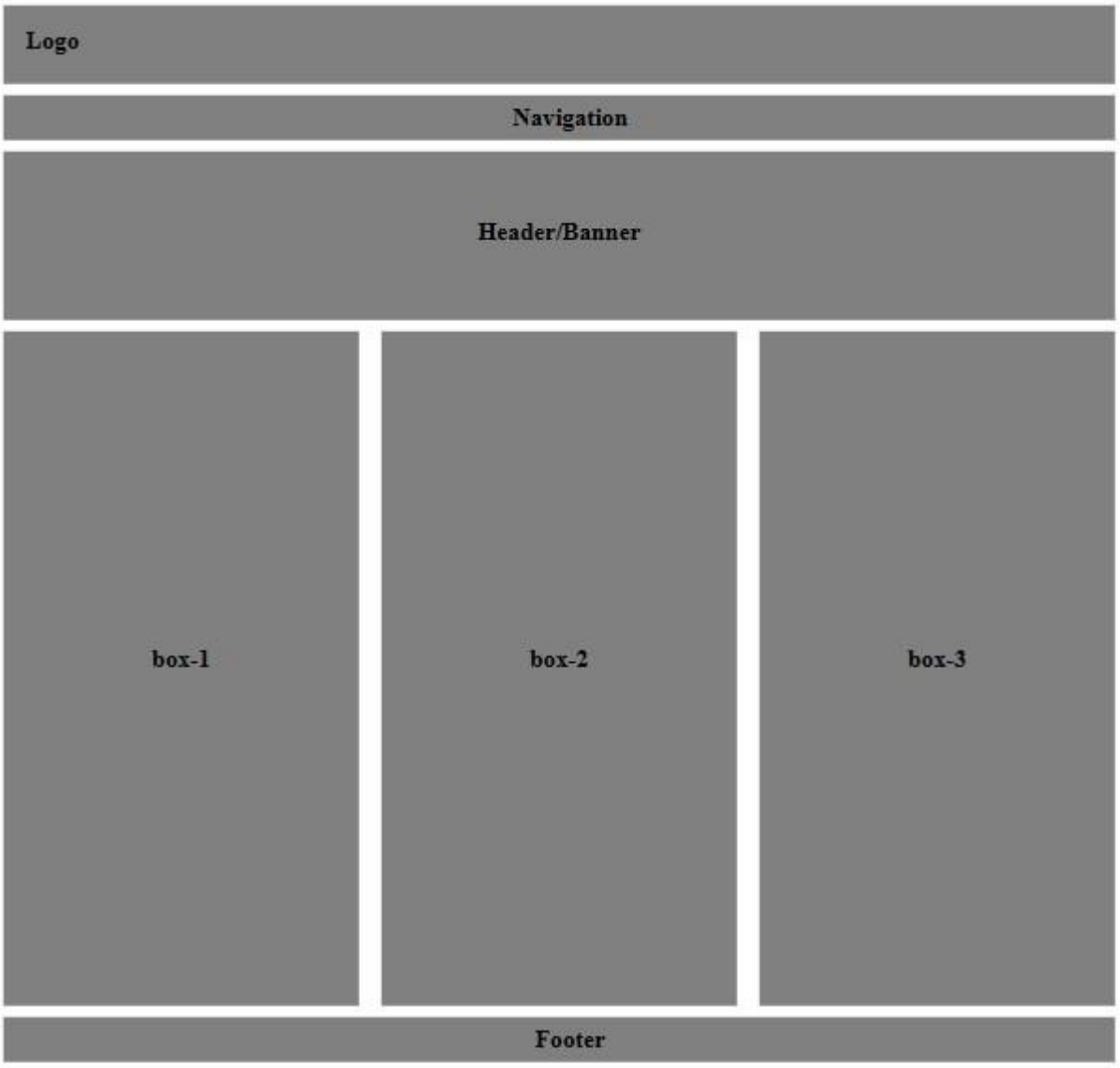
```
                background-color: gray;
                height: 50px; padding-
                top: 20px; width: 98%;
                padding-left: 2%;
                margin-bottom: 10px;
        }
#R2 { width: 100%; margin-bottom: 10px;
                height: 620px;

        }

        #R2 div {
                float: left;
                margin-bottom: 1%;
        }

        #R5 {
                background-color:
                gray; height: 33px;
                width: 100%; padding-
                top: 7px; text-align:
                center; margin-bottom:
                10px;
        }

        #R3 {
                background-color:
                gray; height: 90px;
                width: 100%; padding-
                top: 60px; text-align:
                center;

        } .B1,.B2,.B3,.B5,.B6,.B7,.B9,.B10,.B11,.B13,.B14,.B15 { background-color:
        grey; height: 150px; width: calc(25% - 1%); margin-right: 1%;
        }

        .B4,.B8,.B12,.B16 {
                background-color:
                grey; height: 150px;
                width: 25%;

        }

        #R3 {
                width: 100%; background-
                color: white;
```

```
                height: 222px; padding-
                top: 0px;

                margin-top: 2%;
        }

        #R3 div {
                float: left;
                margin-bottom: 0%;
        }

        #c1 {
                background-color:
                grey; height: 220px;
                width: 32%; margin-
                right: 1%;

        }

        #c2 {
                background-color:
                grey; height: 220px;
                width: 34%; margin-
                right: 1%;
        }

        #c3 {
                background-color:
                grey; height: 220px;
                width: 32%;
        }

        .r4 {

                height: 50px; background-
                color: grey; margin-top:
                1%; text-align: center;
                padding-top: 20px;
        }
    </style>
</head>

<body>
    <div>
        <div id="R1">
            Logo
        </div>
```

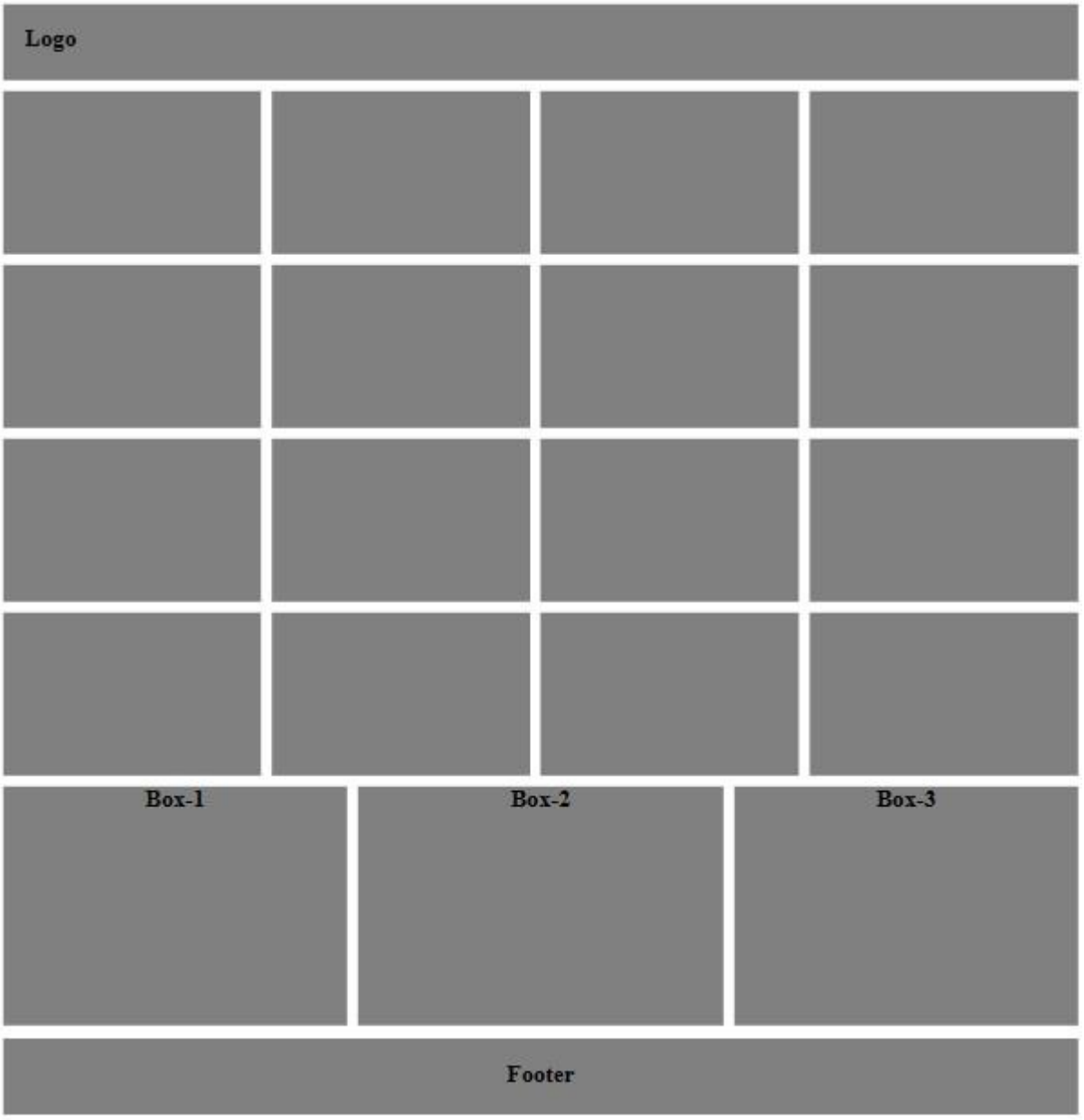
```
        <div id="R2">
                <div class="B1"></div>
                <div class="B2"></div>
                <div class="B3"></div>
                <div class="B4"></div>
                <div class="B5"></div>
                <div class="B6"></div>
                <div class="B7"></div>
                <div class="B8"></div>
                <div class="B9"></div>
                <div class="B10"></div>
                <div class="B11"></div>
                <div class="B12"></div>
                <div class="B13"></div>
                <div class="B14"></div>
                <div class="B15"></div>
                <div class="B16"></div>
        </div>
        <div id="R3">
                <div id="c1">Box-1</div>
                <div id="c2">Box-2</div>
                <div id="c3">Box-3</div>
        </div>

        <div class="r4">
                Footer
        </div>
</body>

</html>
```

Output:


Logo
Navigation
Header/Banner
Side bar
Body Area
Footer


Logo
Navigation
Header/Banner
box-1
box-2
box-3
Footer


Logo
Header/Banner
Side Bar Navigation
Body Area
Footer



Practical-24

Aim: Demonstrate JavaScript Form Validation with proper examples.

Code:

```
<!DOCTYPE html>
<html>

<head>
   <title>Table and Form Example</title>
   <style>
      .container{ height:
         100vh; padding: 6px
```

```
        10px; display: flex;
        justify-content: center;
        align-items: center;
    } form
    {
        margin-top: 20px;
    }

    label {
        display: block; margin-
        bottom: 8px;
    }

    input[type="text"],
    select {
        padding: 6px 10px;
        border: 1px solid #ccc;
        border-radius: 4px; box-
        sizing: border-box;
        margin-bottom: 8px;
        width: 100%;
    }
    input[type="password"],
    select {
        padding: 6px 10px;
        border: 1px solid #ccc;
        border-radius: 4px; box-
        sizing: border-box;
        margin-bottom: 8px;
        width: 100%; }

    input[type="submit"] {
        background-color: #4CAF50;
        color: white;
        padding: 12px 20px;
        border: none;
        border-radius: 4px;
        cursor: pointer;
    }

    input[type="submit"]:hover {
        background-color: #45a049;
    }
  </style>
</head>
```

```
<body>
   <script> function
      verifyPassword() {
         var pw = document.getElementById("pswd").value;
         if (pw == "") {
            document.getElementById("message").innerHTML =
               "**Fill the password
         please!"; return false; }

         if (pw.length < 8) {
            document.getElementById("message").innerHTML =
               "**Password length must be atleast 8
         characters"; return false; }

         if (pw.length > 15) {
            document.getElementById("message").innerHTML =
               "**Password length must not exceed 15 characters";
            return false;
         } else { alert("Password is
            correct");
         }
         if (pw.charAt(0) != pw.charAt(0).toUppercase()) {
            alert("First letter must be Uppercase");
            return false;
         }
         const specialchars = /[`!@#$%^&*()_+\-=\[\]{};':"\\|,.<>\/?~]/;
         if (!specialchars.test(pw)) {
            document.getElementById("message").innerHTML =
"**include Atleat one special character";
            return false;
         }
      }
   </script>


  <div class="container">
   <form onsubmit="return verifyPassword()">
      <label for="name">Name:</label>
      <input type="text" id="name" name="name" placeholder="Enter your name">

      <label for="occupation">Password:</label>
      <input type="password" id="pswd" name="password" placeholder="Enter your
password" required>
      <span id="message" style="color: rgb(255, 0, 0) ;"> </span>
      <br>
      <input type="submit" value="Submit">
```

```
  </form>
 </div>
</body>

</html>
```

Output:

Practical-25

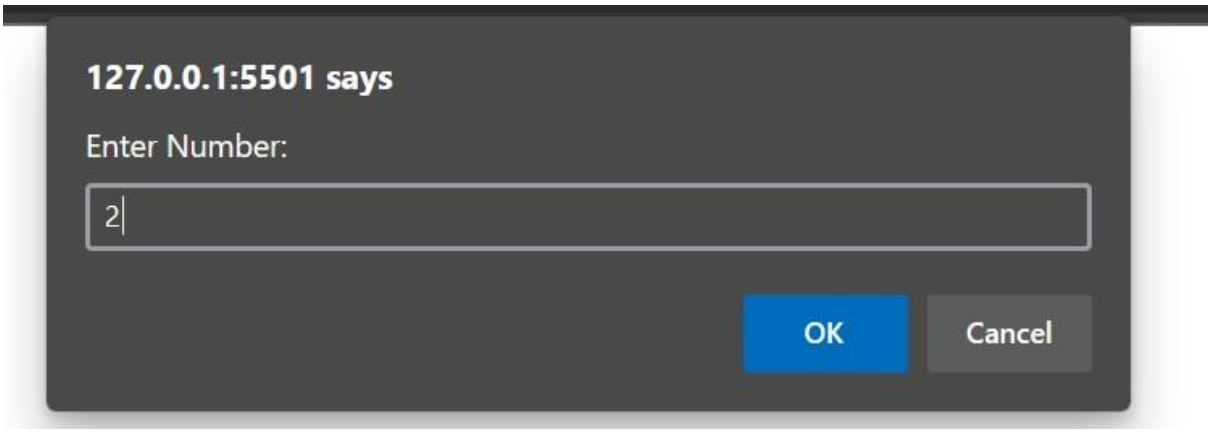
Aim: Write a javascript to check if the number is even or odd.

Code:

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

<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Document</title>


</head>

<body>
   <script> function check(a) {
      const ans = Number(a);
```

```
        if (ans % 2 == 0)
        { return 1; } else {
            return 0;
        }
    }
    inp = prompt("Enter Number: ");
    temp = check(inp); if (temp)
        document.write("Even Number");
    else document.write("Odd
        number");
  </script>
</body>

</html>
```

Output:



Practical-26

Aim: Create a page and access the LocationAPI.

Code:

```
<!DOCTYPE html>
<html>

<body>

   <p>Click the button to get your coordinates.</p>

   <button onclick="getLocation()">Try It</button>

   <p id="demo"></p>

   <script> var x =
      document.getElementById("demo");

      function getLocation() {
         if (navigator.geolocation) {
            navigator.geolocation.getCurrentPosition(showPosition);
         } else {
            x.innerHTML = "Geolocation is not supported by this browser.";
         }
      }

      function showPosition(position) {
         x.innerHTML = "Latitude: " + position.coords.latitude +
            "<br>Longitude: " + position.coords.longitude;
      }
   </script>

</body>

</html>
```

Output:

Click the button to get your coordinates.

Try It

Latitude: 23.0302
Longitude: 72.5772

Practical-27

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

Code:

```
<!DOCTYPE html>
<html>

<body>

   <div id="demo">
      <h2>Let AJAX change this text</h2>
   </div>

   <button type="button" onclick="loadDoc()">Change Content</button>

   <script> function
      loadDoc() {
         var xhttp = new XMLHttpRequest();
         xhttp.onreadystatechange = function () {
            if (xhttp.readyState == 4 && xhttp.status == 200) {
               document.getElementById("demo").innerHTML = xhttp.responseText;
            }
         };
         xhttp.open("GET", "ajax_info.txt", true);
         xhttp.send();
      }
   </script>

</body>

</html>
```

Output:

\

AJAX

AJAX is not a programming language.

AJAX is a technique for accessing web servers from a web page.

AJAX stands for Asynchronous JavaScript And XML.