Name: TRUSH ISAMALIYA RAJESHBHAI

Enrollment:2202030400048

Branch:B.TECH(CE)

Institude: ADITYA SILVER OAK UNIVERSITY

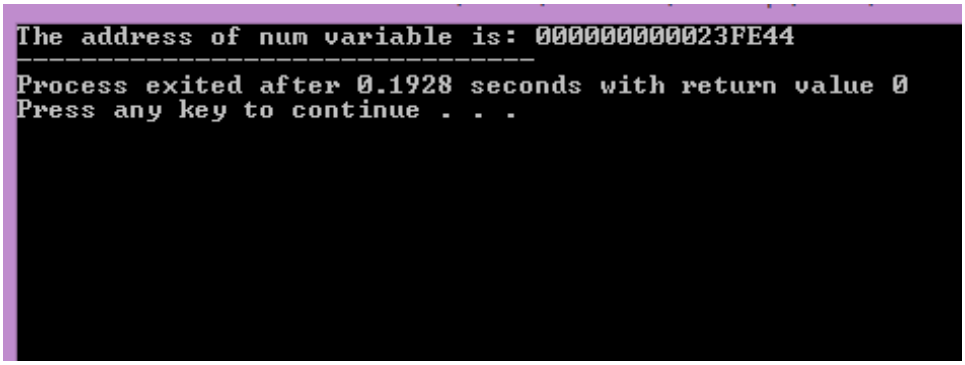
Practical-1

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

Code:

```
#include <stdio.h>

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

    printf("The address of num variable is: %p", ptr);

    return 0;
}
```

Output:



Practical-2

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

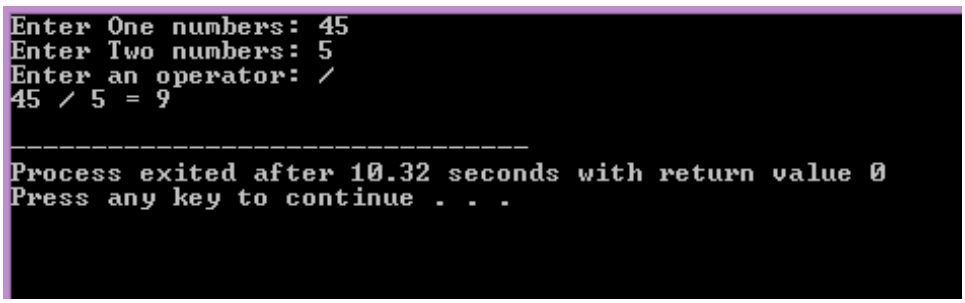
Code:

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

int main() {
    float num1, num2, *ptr;
    char calculator;

    printf("Enter One numbers: ");
    scanf("%f", &num1);
```

```
printf("Enter Two numbers: ");
scanf("%f",&num2);

printf("Enter an operator: ");
scanf(" %c",&calculator);

switch(calculator){
   case '+':
      ptr = &num1;
      printf("%.f + %.f = %.f\n", *ptr, num2, *ptr + num2);
      break;

   case '-':
      ptr = &num1;
      printf("%.f - %.f = %.f\n", *ptr, num2, *ptr - num2);
      break;

   case '*':
      ptr = &num1;
      printf("%.f * %.f = %.f\n", *ptr, num2, *ptr * num2);
      break;

   case '/':
      ptr = &num1;
      printf("%.f / %.f = %.f\n", *ptr, num2, *ptr / num2);
```

```
            break;


        default:
            printf("Invalid number.\n");
    }


    return 0;
}
```

Output:



Practical-3

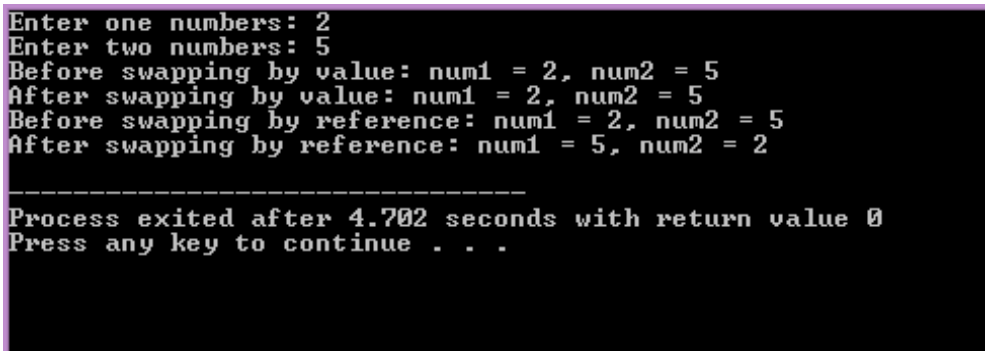
Aim: Write a C program to swap the two values using call by value and call by reference.

Code:

```
#include <stdio.h>

void swap_value(int x, int y);
void swap_reference(int *x, int *y);

int main() {
    int num1, num2;

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

    // Call by value
    printf("Before swapping by value: num1 = %d, num2 = %d\n", num1, num2);
    swap_value(num1, num2);
    printf("After swapping by value: num1 = %d, num2 = %d\n", num1, num2);

    // Call by reference
    printf("Before swapping by reference: num1 = %d, num2 = %d\n", num1,
num2);
    swap_reference(&num1, &num2);
    printf("After swapping by reference: num1 = %d, num2 = %d\n", num1, num2);
```

```
    return 0;
}

void swap_value(int x, int y) {
    int temp = x;
    x = y;
    y = temp;
}

void swap_reference(int *x, int *y) {
    int temp = *x;
    *x = *y;
    *y = temp;
}
```

Output:



SILVER OAK UNIVERSITY
EDUCATION TO INNOVATION

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>
struct person
{
  char name[20];
  char dob[10];
  int age;
}p[4];

int main(void)
{
  int i=0;

  for(i=0;i<4;i++)
  {
    printf("\n Enter person name :");
    scanf("%s", p[i].name);
```

```
    printf("\n Person Date of joining(dd-mm-yyyy) : ");
    scanf("%s",p[i].dob);
    printf("\n Enter person age : ");
    scanf("%d",&p[i].age);
  }

  for(i=0;i<4;i++)
  {
    printf("\n Person %d Detail",i+1);
    printf("\n Name = %s",p[i].name);
    printf("\n DOB = %s",p[i].dob);
    printf("\n age = %.d",p[i].age);
  }
  return 0;
}
```

Output:

```
Enter person age : 19

Enter person name :Dhruv

Person Date of joining(dd-mm-yyyy) : 16/8/2004

Enter person age : 17

Person 1 Detail
Name = Trush
DOB = 23/12/2003
age = 18
Person 2 Detail
Name = Darshan
DOB = 7/4/2004
age = 17
Person 3 Detail
Name = Raj
DOB = 29/2/2002
age = 19
Person 4 Detail
Name = Dhruv
DOB = 16/8/2004
age = 17
--------------------------------
Process exited after 90.43 seconds with return value 0
Press any key to continue . . .
```

Practical-5

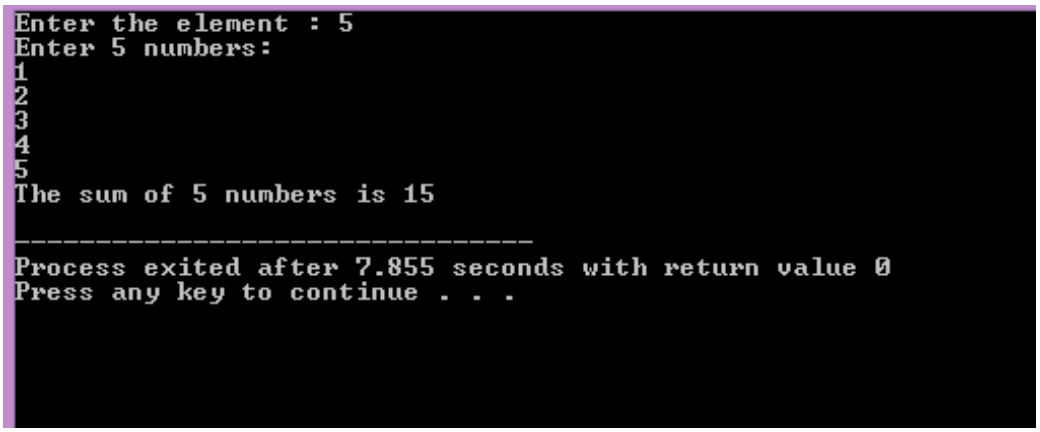
Aim: Write a C program to calculate the sum of n number entered by the user using dynamic memory allocation.

Code:

```
#include <stdio.h>

#include <stdlib.h>
```

```
int main() {
    int n, i, *ptr, sum = 0;

    printf("Enter the element : ");
    scanf("%d", &n);

    // Allocate memory dynamically
    ptr = (int*) malloc(n * sizeof(int));

    // Read input for n numbers
    printf("Enter %d numbers:\n", n);
    for(i = 0; i < n; i++) {
        scanf("%d", ptr + i);
        sum += *(ptr + i);
    }

    printf("The sum of %d numbers is %d\n", n, sum);

    return 0;
}
```

Output:



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

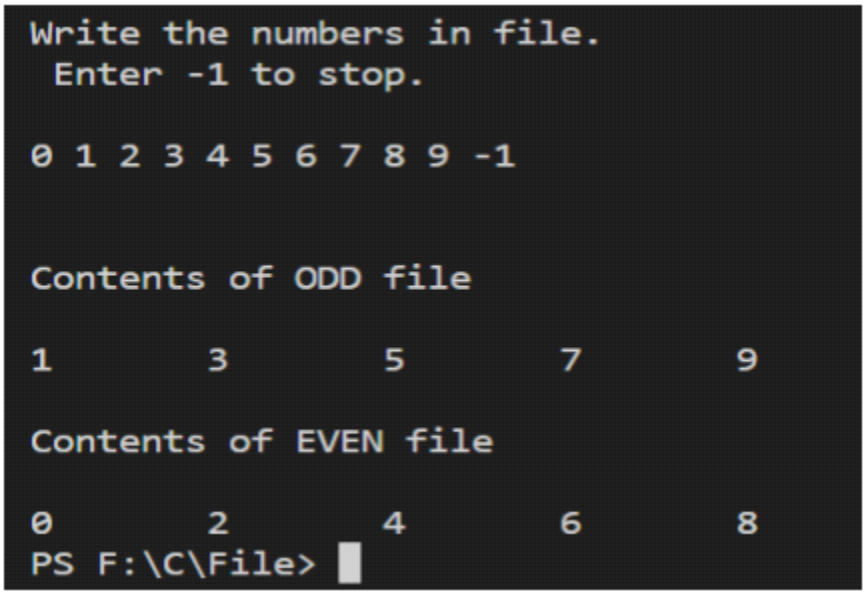
Code:

```
#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);
}
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:



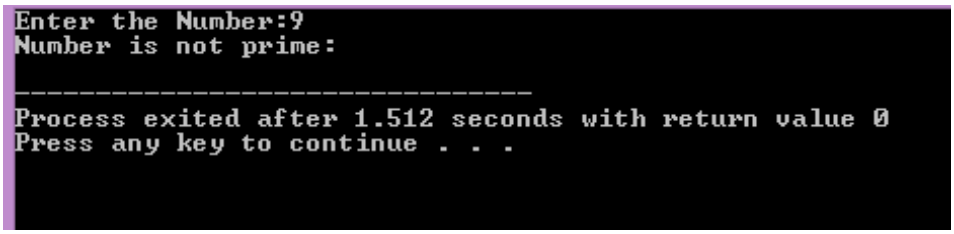
Practical-7

Aim: 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,f=0;
	cout<<"Enter the Number:";
	cin>>n;
	m=n/2;
	for(i=2;i<=m;i++){
		if(n%i==0){
			cout<<"Number is not prime:"<<endl;
			f=1;
			break;
		}
	}
	if(f==0)
	cout<<"Number is prime:"<<endl;
	return 0;

}
```

Output:



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 letter;

   cout << "Enter a letter:";
   cin >> letter;

   switch (letter) {
      case 'a':
```

```
        case 'e':
        case 'i':
        case 'o':
        case 'u':
        case 'A':
        case 'E':
        case 'I':
        case 'O':
        case 'U':
            cout << letter << " is a vowel." << endl;
            break;
        default:
            cout << letter << " is a consonant."<< endl;
            break;
    }

    return 0;
}
```

Output:

```
Enter a letter:o
o is a vowel.

--------------------------------
Process exited after 4.085 seconds with return value 0
Press any key to continue . . .
```

Practical-9

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

Code:

```
#include <iostream>

using namespace std;

class Example {
   private:
      int value;
   public:
      Example() {
         cout << "Constructor called" << endl;
         value = 0;
      }
      ~Example() {
         cout << "Destructor called" << endl;
      }
      void setValue(int v) {
         value = v;
```

```
    }
    int getValue() {
        return value;
    }
};

int main() {
    Example obj;
    obj.setValue(4);

    cout << "Value of obj is " << obj.getValue() << endl;
    cout << "Value of obj is " << obj.getValue() << endl;
    return 0;
}
```

Output:

```
Constructor called
Value of obj is 4
Value of obj is 4
Destructor called

--------------------------------
Process exited after 0.1337 seconds with return value 0
Press any key to continue . . .
```

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>
using namespace std;

#define MAX 10

class student {
private:
   char name[30];
   int rollNo;
   int total;
   float perc;

public:
   //member function to get student's details
   void getDetails(void);
   //member function to print student's details
   void putDetails(void);
};
```

```
//member function definition, outside of the class
void student::getDetails(void)
{
   cout << "Enter name: ";
   cin >> name;
   cout << "Enter roll number: ";
   cin >> rollNo;
   cout << "Enter total marks outof 500: ";
   cin >> total;

   perc = (float)total / 500 * 100;
}

//member function definition, outside of the class
void student::putDetails(void)
{
   cout << "Student details:\n";
   cout << "Name:" << name << ",Roll Number:" << rollNo << ",Total:" << total
<< ",Percentage:" << perc;
}

int main()
{
   student std[MAX]; //array of objects creation
```

```
    int n, loop;

    cout << "Enter total number of students: ";
    cin >> n;

    for (loop = 0; loop < n; loop++) {
        cout << "Enter details of student " << loop + 1 << ":\n";
        std[loop].getDetails();
    }

    cout << endl;

    for (loop = 0; loop < n; loop++) {
        cout << "Details of student " << (loop + 1) << ":\n";
        std[loop].putDetails();
    }

    return 0;
}
```

Output:

```
Enter total number of students: 2
Enter details of student 1:
Enter name: Treush
Enter roll number: 048
Enter total marks outof 500: 350
Enter details of student 2:
Enter name: Darshan
Enter roll number: 040
Enter total marks outof 500: 413

Details of student 1:
Student details:
Name:Treush,Roll Number:48,Total:350,Percentage:70Details of student 2:
Student details:
Name:Darshan,Roll Number:40,Total:413,Percentage:82.6
--------------------------------
Process exited after 58.74 seconds with return value 0
Press any key to continue . . .
```

Practical-11

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

Code:

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

class Number {
  private:
    int num;

  public:
```

```
    Number() {
      num = 0;
    }

    Number(int n) {
      num = n;
    }

    Number operator+(Number const &obj) {
      Number res;
      res.num = num + obj.num;
      return res;
    }

    void display() {
      cout << num << endl;
    }
};

int main() {
  Number num1(10);
  Number num2(20);
  Number sum = num1 + num2;
  sum.display();
  return 0;
```

```
}
```

Output:

```
30

--------------------------------
Process exited after 0.0928 seconds with return value 0
Press any key to continue . . .
```

Practical-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:

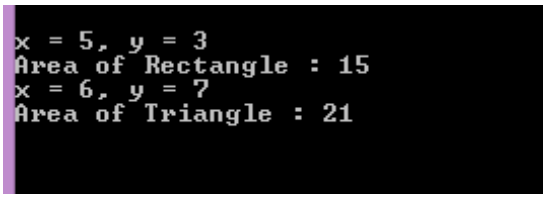
```
#include<iostream>

#include<conio.h>

using namespace std;

class shape

{

protected:
```

```
        double x,y;
public:
        void get(int a,int b)
        {
                x=a;
                y=b;
        }

        virtual void display_area()=0;
};

class rec:public shape
{
public:
        void display_area()
        {
                cout<<"\nx = "<< x <<", y = "<< y <<" ";
                cout<<"\nArea of Rectangle : ";
                cout<<x*y;
        }
};

class tri:public shape
{
public:
```

```
        void display_area()
        {
                cout<<"\nx = "<< x <<", y = "<< y <<" ";
                cout<<"\nArea of Triangle : ";
                cout<<0.5*x*y;
        }
};

int main()
{

        shape *ptr;
        rec r1;
        tri t1;

        r1.get(5,3);
        t1.get(6,7);

        ptr=&r1;
        ptr->display_area();

        ptr=&t1;
        ptr->display_area();

        getch();
```

```
    return 0;
}
```

Output:



Practical-13

Aim: 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);
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

SILVER OAK UNIVERSITY
EDUCATION TO INNOVATION

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(

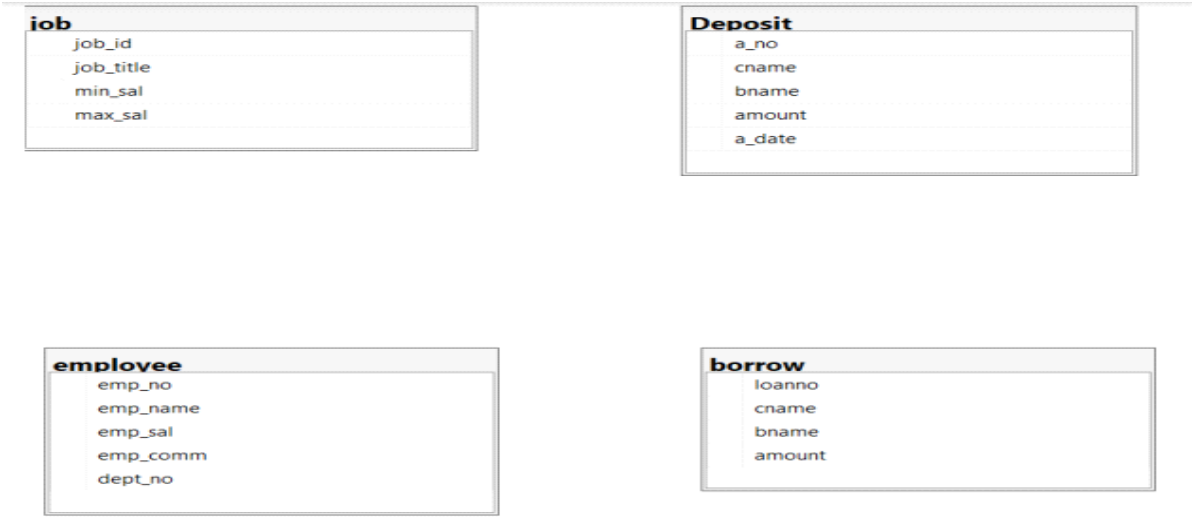
loanno int,

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:

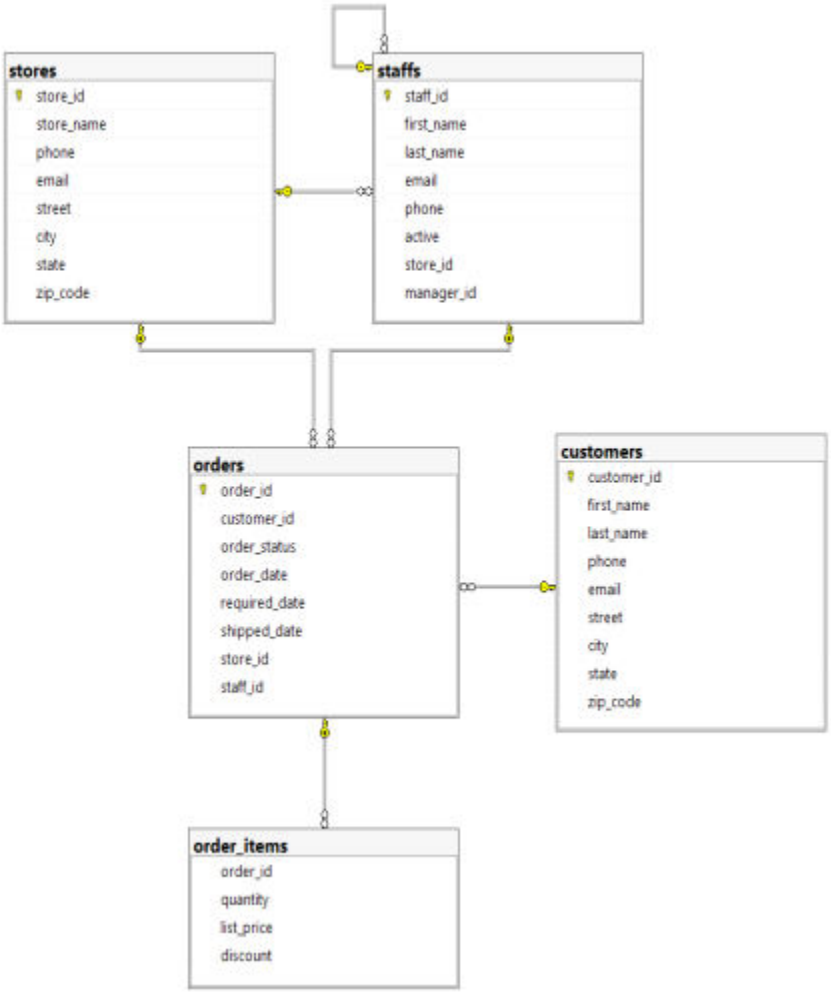
```
create table customers(
customer_id int not null primary key identity(1,1),
last_name varchar(50) not null ,
first_name varchar(50) not null ,
phone bigint not null ,
email varchar(50),
street varchar(50),
city varchar(50) not null ,
state varchar(50) not null ,
zip_code int
);
create table staffs(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

```
<!DOCTYPE html>

<html>

 <head>

  <title>Resume</title>

 </head>

 <body>

  <h1>Trush Isamaliya</h1>

  <p>Email: Trush@email.com</p>
```

```
    <p>Phone: 7201935377</p>
    <h2>Objective</h2>
    <p>To obtain a position as a web developer in a  company.</p>
    <h2>Education</h2>
<h3>College</h3>
    <p>B.Tech in Computer Engeneering</p>
    <p>Aditya Sliver Oak University, Ahmedabad</p>
    <p>1 Year -Running</p>

<h3>12th</h3>
    <p>Science</p>
    <p>School of Science, Rajkot</p>
    <p>Percentage-60%</p>
<h3>10th</h3>
    <p>J.M.Ajmera English Medium School, Damnagar</p>
    <p>Percentage-60%</p>
    <h2>Skills</h2>
 <ul>
      <li>HTML</li>
      <li>CSS</li>
      <li>JavaScript</li>
      <li>MySQL</li>
    </ul>

    <h2>Hobbies</h2>
```

```
<ul>
  <li>Cricket</li>
  <li>Chess</li>
  <li>Football</li>
  <li>Badminton</li>
</ul>

<h2>Declaration</h2>
<p>I do hereby declare that all the details mentioned above are accurate to the best of my familiarity and confidence.</p>
<p>18 March 2023</p>
</body>
</html>
```

Output:

SILVER OAK UNIVERSITY
EDUCATION TO INNOVATION

Trush Isamaliya

Email: Trush@email.com

Phone: 7201955377

Objective

To obtain a position as a web developer in a company.

Education

College

B.Tech in Computer Engeneering

Aditya Silver Oak University, Ahmedabad

1 Year -Running

12th

Science

School of Science, Rajkot

Percentage-60%

10th

J.M.Ajmera English Medium School, Damnagar

Percentage-60%

Skills

- HTML
- CSS
- JavaScript
- MySQL

Hobbies

- Cricket
- Chess
- Football
- Badminton

Declaration

I do hereby declare that all the details mentioned above are accurate to the best of my familiarity and confidence.

18 March 2023

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
.jp
g" 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:

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>
</head>
<body>
        <h1>Table and Form Example</h1>

        <h2>Table</h2>
        <table border=1>
                <thead>
                        <tr>
                                <th>Name</th>
                                <th>Age</th>
                                <th>Gender</th>
                        </tr>
                </thead>
                <tbody>
                        <tr>
                                <td>John</td>
                                <td>35</td>
                                <td>Male</td>
                        </tr>
                        <tr>
                                <td>Jane</td>
                                <td>27</td>
                                <td>Female</td>
                        </tr>
```

```
                <tr>
                    <td>Bob</td>
                    <td>42</td>
                    <td>Male</td>
                </tr>
            </tbody>
        </table>

        <h2>Form</h2>
        <form>
            <label for="name">Name:</label>
            <input type="text" id="name" name="name"><br>

            <label for="email">Email:</label>
            <input type="email" id="email" name="email"><br>

            <label for="message">Message:</label><br>
            <textarea id="message" name="message" rows="5" cols="30"></textarea><br>

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

Output:

Table and Form Example

Table

Name	Age	Gender
John	35	Male
Jane	27	Female
Bob	42	Male

Form

Name:
Email:
Message:

Submit

Practical-21

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

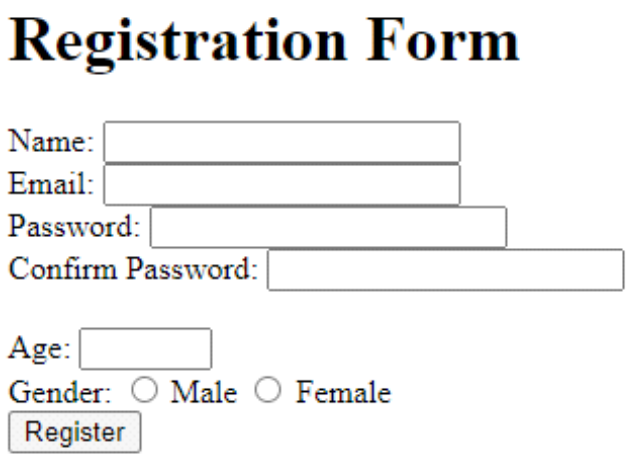
Code:

```
<!DOCTYPE html>

<html>

<head>
```

```
        <title>Registration Form</title>
</head>
<body>
        <h1>Registration Form</h1>
        <form>
                <label for="name">Name:</label>
                <input type="text" id="name" name="name" required minlength="3"
maxlength="50"><br>

                <label for="email">Email:</label>
                <input type="email" id="email" name="email" required><br>

                <label for="password">Password:</label>
                <input type="password" id="password" name="password" required
minlength="6"><br>

                <label for="confirm_password">Confirm Password:</label>
                <input type="password" id="confirm_password" name="confirm_password" required
minlength="6" oninput="checkPasswordMatch()"><br>

                <span id="password_match_error" style="color: red; display: none;">Passwords do not
match</span><br>

                <label for="age">Age:</label>
                <input type="number" id="age" name="age" required min="18" max="99"><br>
```

```
            <label for="gender">Gender:</label>
            <input type="radio" id="male" name="gender" value="male" required>
            <label for="male">Male</label>
            <input type="radio" id="female" name="gender" value="female" required>
            <label for="female">Female</label><br>

            <input type="submit" value="Register">
        </form>

        <script>
            function checkPasswordMatch() {
                var password = document.getElementById("password");
                var confirm_password = document.getElementById("confirm_password");
                var error = document.getElementById("password_match_error");

                if (password.value != confirm_password.value) {
                    error.style.display = "block";
                } else {
                    error.style.display = "none";
                }
            }
        </script>
</body>
</html>
```

Output:



Practical-22

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

Code:

```
<!DOCTYPE html>

<html>

<head>

        <title>Resume</title>

        <style>

                body {

                        font-family: Arial, sans-serif;

                        background-color: #F0F0F0;

                }
```

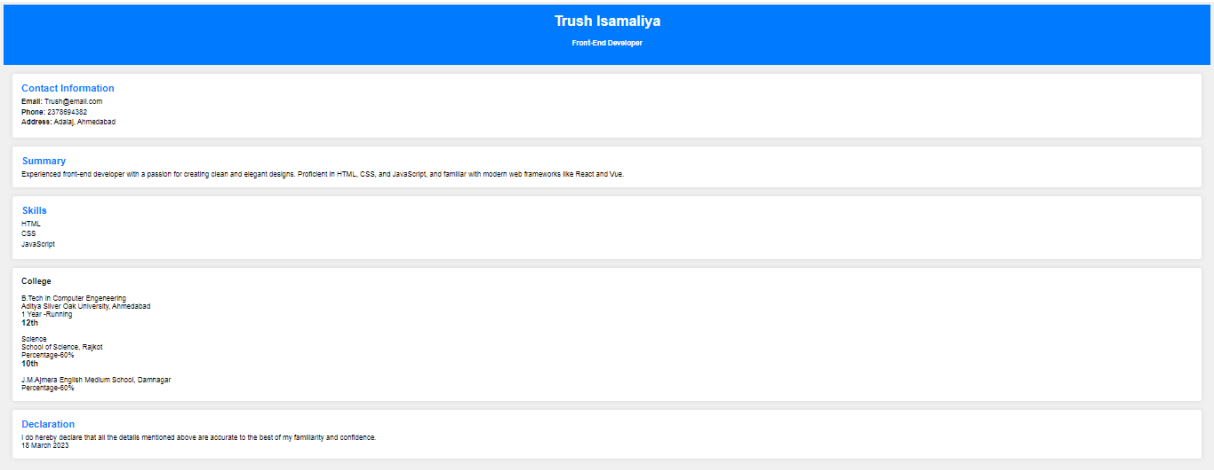
```
h1, h2, h3, h4, h5, h6 {
        margin-top: 0;
}
header {
        background-color: #007BFF;
        color: #FFF;
        padding: 20px;
        text-align: center;
}
section {
        background-color: #FFF;
        padding: 20px;
        margin: 20px;
        border-radius: 5px;
        box-shadow: 0 0 10px rgba(0,0,0,0.1);
}
section h2 {
        color: #007BFF;
        font-size: 22px;
        margin-bottom: 10px;
}
section ul {
        margin: 0;
        padding: 0;
        list-style: none;
```

```
            }
            section ul li {
                margin-bottom: 5px;
            }
            section p {
                margin: 0;
            }
        </style>
</head>
<body>
        <header>
            <h1>Trush Isamaliya</h1>
            <h4>Front-End Developer</h4>
        </header>
        <section>
            <h2>Contact Information</h2>
            <ul>
                <li><strong>Email:</strong> Trush@email.com</li>
                <li><strong>Phone:</strong> 2378694382</li>
                <li><strong>Address:</strong> Adalaj, Ahmedabad</li>
            </ul>
        </section>
        <section>
            <h2>Summary</h2>
```

```
        <p>Experienced front-end developer with a passion for creating clean and elegant designs. Proficient in HTML, CSS, and JavaScript, and familiar with modern web frameworks like React and Vue.</p>

    </section>

    <section>

        <h2>Skills</h2>

        <ul>

            <li>HTML</li>

            <li>CSS</li>

            <li>JavaScript</li>

                </ul>

    </section>
<section>
<h3>College</h3>
  <p>B.Tech in Computer Engeneering</p>
  <p>Aditya Sliver Oak University, Ahmedabad</p>
  <p>1 Year -Running</p>


<h3>12th</h3>
  <p>Science</p>
  <p>School of Science, Rajkot</p>
  <p>Percentage-60%</p>
<h3>10th</h3>
  <p>J.M.Ajmera English Medium School, Damnagar</p>
  <p>Percentage-60%</p>
```

```
</section>

<section>

<h2>Declaration</h2>

 <p>I do hereby declare that all the details mentioned above are accurate to the best of my familiarity and confidence.</p>

<p>18 March 2023</p>

</section>

                </body>

</html>
```

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.

```
<!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: 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%;
```

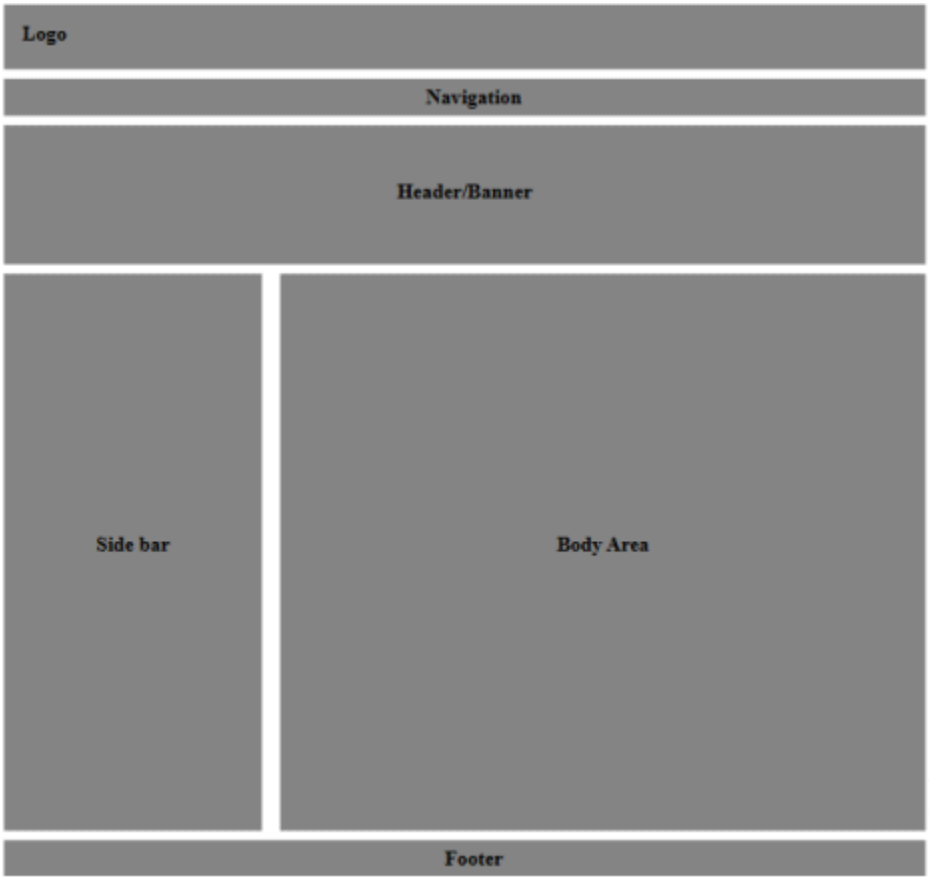
```
padding-top: 60px;
text-align: center;
}
.B1,.B2,.B3,.B5,.B6,.B7,.B9,.B10,.B11,.B13,.B14,.B15 {
background-color: grey;
height: 150px;
width: calc(25% - 1%);
margin-right: 1%;
}
.B4,.B8,.B12,.B16 {
background-color: grey;
height: 150px;
width: 25%;
}
#R3 {
width: 100%;
background-color: white;
height: 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>
```

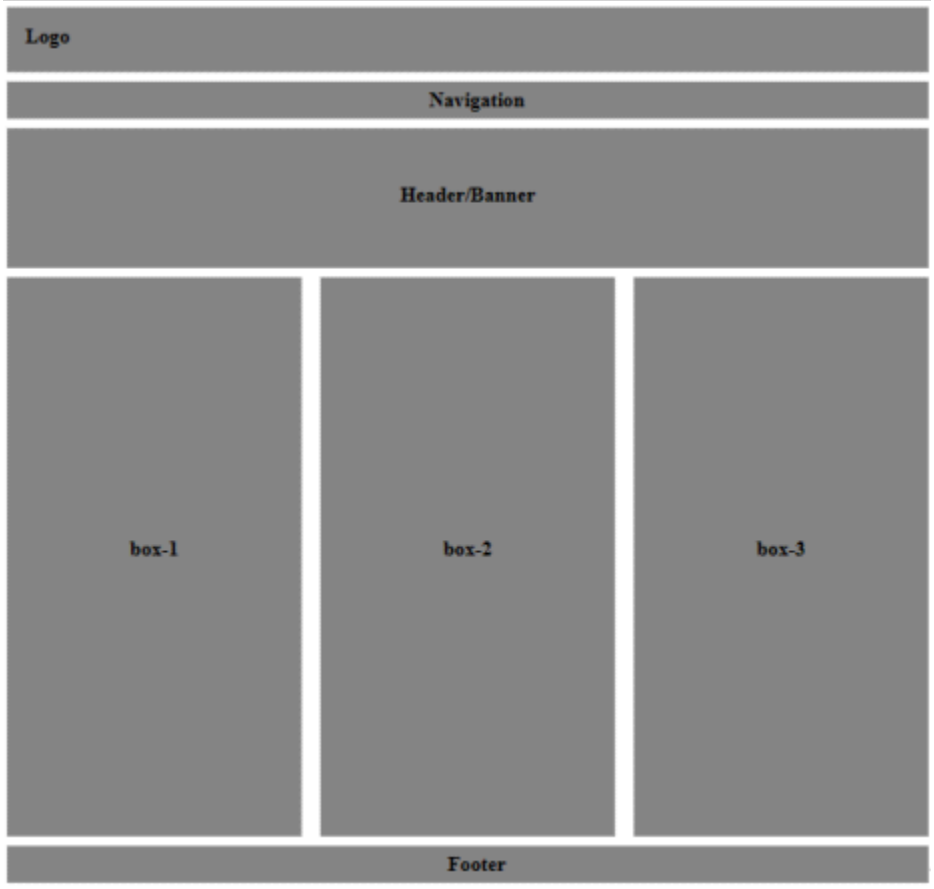
```
<div>
<div id="R1">
Logo
</div>
<div id="R2">
<div class="B1"></div>
<div class="B2"></div>
<div class="B3"></div>
<div class="B4"></div>
<div class="B5"></div>
<div class="B6"></div>
<div class="B7"></div>
<div class="B8"></div>
<div class="B9"></div>
<div class="B10"></div>
<div class="B11"></div>
<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>
```

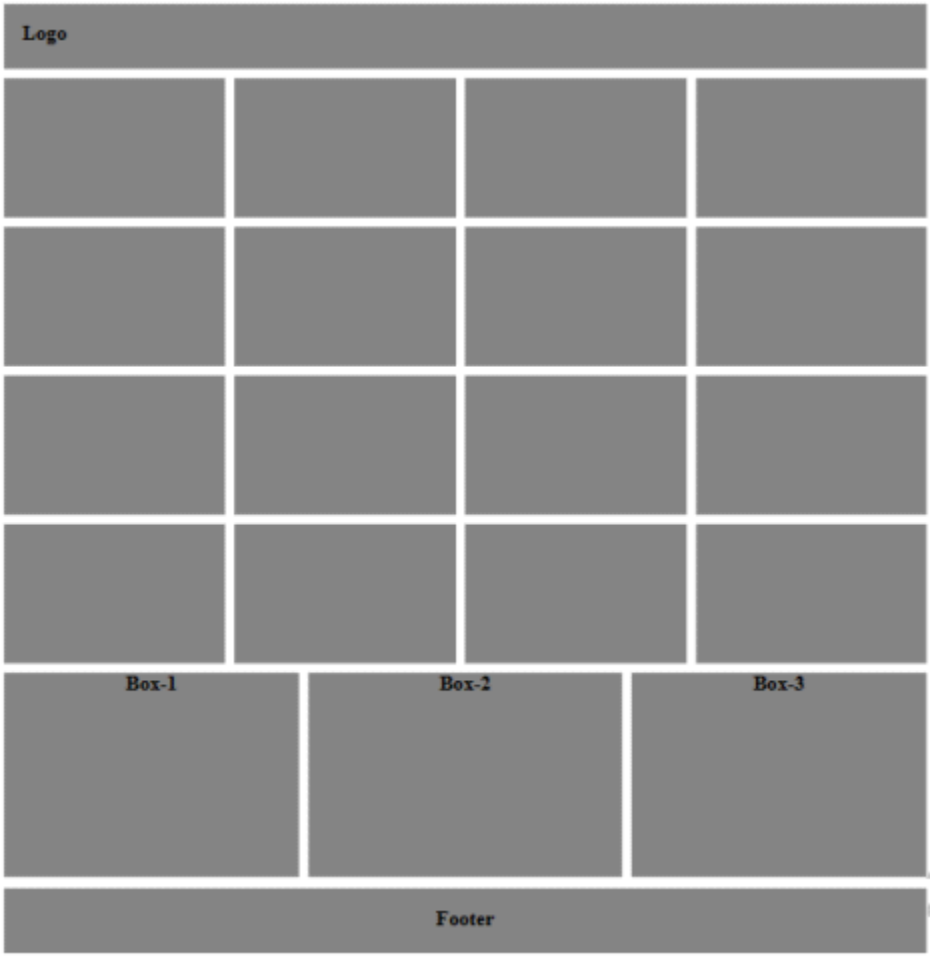
```
</div>
<div class="r4">
Footer
</div>
</body>
</html>
```

Output:



SILVER OAK UNIVERSITY
EDUCATION TO INNOVATION





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

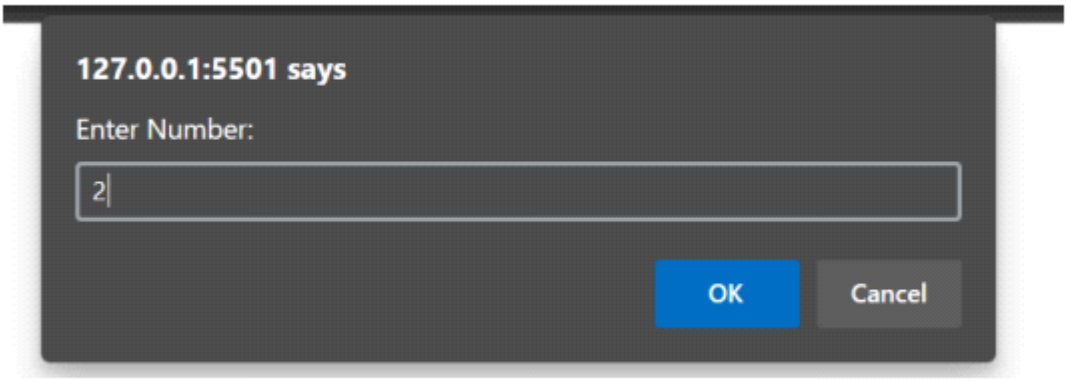
Aim: Write a javascript to check if the number is even or odd.

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

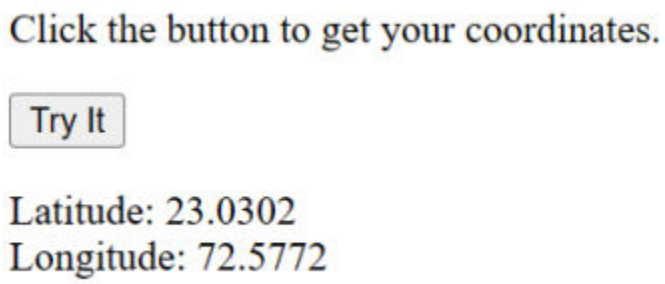
Aim: Create a page and access the LocationAPI**.**

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:



Practical-27

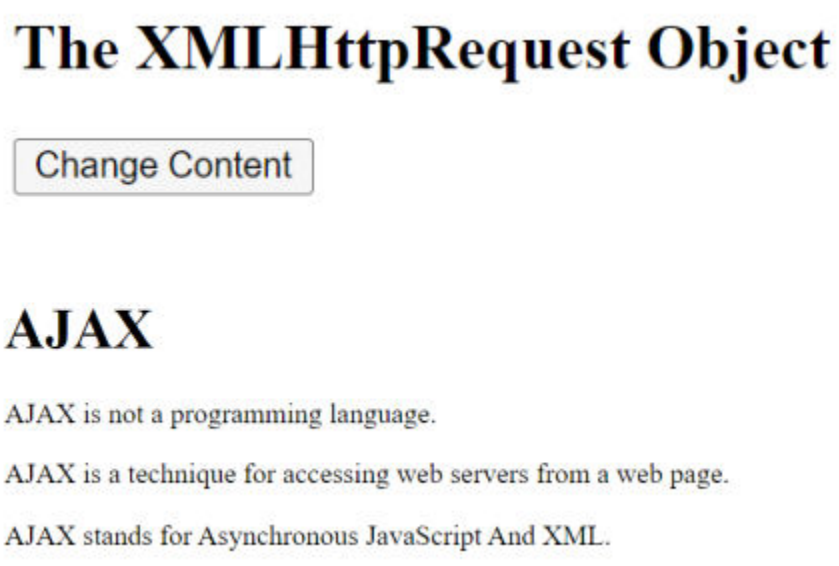
Aim: Create a simple XMLHTTPRequest,and retrieve the data from the text file.

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:



SILVER OAK UNIVERSITY
EDUCATION TO INNOVATION