Practical-1

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

Code:

```
#include<stdio.h>
#include<conio.h>
int main()
{
   int num;
   printf("Enter any number to store in \"num\" variable: ");
scanf("%d", &num);     printf("\nValue of num = %d",
num);
   printf("\nAddress of num = %u",
&num);     getch();     return 0;
}
```

Output:

```
Enter any number to store in "num" variable: 10

Value of num = 10
Address of num = 2770632164
```

Practical-2

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

Code:

```
#include<stdio.h>
#include<stdlib.h>      int main()
 {   int a,b;
 int *p1,*p2;
 char ch;
 p1=&a;
 p2=&b;
   printf("Select the Operation\n");
 printf("Type + for Addition\n");   printf("Type
 - for Subtraction\n");   printf("Type * for
 Multiplication\n");   printf("Type / for
 Division\n");   scanf("%c",&ch);
   printf("Enter any two numbers\n");
   scanf("%d%d",&a,&b);
   switch(ch)
   {   case
 '+':
       printf("%d + %d = %d",a,b,(*p1+*p2));
 break;   case '-':
       printf("%d - %d = %d",a,b,(*p1-
 *p2));       break;   case '*':
       printf("%d * %d =
 %d",a,b,(*p1**p2));       break;   case '/':
       if(*p2==0)
       {
          printf("Sorry, You can not divide a number by 0");
 return 0;
       }
       printf("%d / %d = %0.2f",a,b,(*p1/(float)*p2));
 break;   default:     printf("Sorry, Invalid Choice");
   }
 return 0;
```

Output:

```
Select the Operation
Type + for Addition
Type - for Subtraction
Type * for Multiplication
Type / for Division
+
Enter any two numbers
10
5
10 + 5 = 15
```

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

    printf("Before swapping the values in main a = %d, b = %d\n",a,b);
    swap(&a,&b);
    printf("After swapping values in main a = %d, b = %d\n",a,b);
}
void swap (int *a, int *b)
{
   int temp;
temp = *a;
   *a=*b;
    *b=temp;
    printf("After swapping values in function a = %d, b = %d\n",*a,*b);
}
```

Output:

```
Before swapping the values in main a = 10, b = 20
After swapping values in function a = 20, b = 10
After swapping values in main a = 20, 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>  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 Birth(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 name :Manthan

Person Date of Birth(dd-mm-yyyy) : 19-03-2004

Enter person Age : 19

Enter person name :Mann

Person Date of Birth(dd-mm-yyyy) : 15-12-2003

Enter person Age : 19

Enter person name :Keval

Person Date of Birth(dd-mm-yyyy) : 1-11-2004

Enter person Age : 18

Enter person name :Mayank

Person Date of Birth(dd-mm-yyyy) : 18-08-2004

Enter person Age : 18

Person 1 Detail
Name = Manthan
DOB = 19-03-2004
Age = 19
Person 2 Detail
Name = Mann
DOB = 15-12-2003
Age = 19
```

```
Person 3 Detail
Name = Keval
DOB = 1-11-2004
Age = 18
Person 4 Detail
Name = Mayank
DOB = 18-08-2004
Age = 18
```

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>

#include<stdlib.h>

int main(){

 int n,i,*ptr,sum=0;
   printf("Enter number of elements: ");
scanf("%d",&n);
   ptr=(int*)malloc(n*sizeof(int));  //memory allocated using malloc
if(ptr==NULL)
   {
      printf("Sorry! unable to allocate memory");
exit(0);         }
   printf("Enter elements of array: ");
   for(i=0;i<n;++i)

   {
      scanf("%d",ptr+i);
sum+=*(ptr+i);
   }
   printf("Sum=%d",sum);
free(ptr);
 return 0;
 }
```

Output:

```
Enter number of elements: 5
Enter elements of array: 10
15
30
20
25
Sum=100
```

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>

 void main()

 {

   FILE *f1, *f2, *f3;
  int number, i;

  printf("Contents of DATA file\n\n");
f1 = fopen("DATA", "w");      for(i = 1;
i <= 30; i++)
   {
     scanf("%d", &number);
if(number == -1) break;
putw(number,f1);
   }
fclose(f1);

 f1 = fopen("DATA", "r");

f2 = fopen("ODD", "w");  f3
= fopen("EVEN", "w");
```

```
  while((number = getw(f1)) != EOF)

  {      if(number %2 ==
0)
       putw(number, f3); /* Write to EVEN file */
else
       putw(number, f2); /* Write to ODD file */
 }  fclose(f1);
  fclose(f2);

  fclose(f3);

  f2 = fopen("ODD","r");

  f3 = fopen("EVEN", "r");

 printf("\n\nContents of ODD file\n\n");
while((number = getw(f2)) != EOF)
printf("%4d", number);
    printf("\n\nContents of EVEN file\n\n");
while((number = getw(f3)) != EOF)
printf("%4d", number);
  fclose(f2);

  fclose(f3);

  }
```

Output:

```
Contents of DATA file

111 222 333 444 555 666 777 888 999 000 121 232 343 454 565  Contents of ODD file

111 333 555 777 999 121 343 565

Contents of EVEN file
```

```
222 444 666 888 0 232 454
```

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;

bool check_prime(int);
 int main() {       int n;
   cout << "Enter a positive  integer: ";
 cin >> n;
   if (check_prime(n))
     cout << n << " is a prime number.";
 else
     cout << n << " is not a prime number.";
 return 0;
 }
 bool check_prime(int n)
 {   bool is_prime = true;
 if (n == 0 || n == 1) {
 is_prime = false;
   }
  for (int i = 2; i <= n / 2; ++i)
 {     if (n % i == 0) {
 is_prime = false;       break;
    }
 }
  return is_prime;   }
```

Output:

```
Enter a positive  integer: 47
47 is a prime number.
```

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;
   bool isLowercaseVowel, isUppercaseVowel;

cout << "Enter an alphabet: ";     cin >> c;

   isLowercaseVowel = (c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u');

   isUppercaseVowel = (c == 'A' || c == 'E' || c == 'I' || c == 'O' || c == 'U');
if (!isalpha(c))
     printf("Error! Non-alphabetic character.");

 else if (isLowercaseVowel || isUppercaseVowel)

       cout << c << " is a vowel.";

   else
       cout << c << " is a consonant.";
return 0;
 }
```

Output:

```
Enter an alphabet: Y
Y is a consonant.
```

Practical-9

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

Code:

```
#include <iostream>  using
namespace std;  class
class_name{   private:
    int a,b;
  public:
 class_name(int aa, int bb)
  {       cout<<"Constructor is
called"<<endl;
      a = aa;       b = bb;
cout<<"Value of a: "<<a<<endl;
cout<<"Value of b: "<<b<<endl;
cout<<endl;
    }
   ~class_name()
  {       cout<<"Destructor is
called"<<endl;       cout<<"Value of a:
"<<a<<endl;       cout<<"Value of b:
"<<b<<endl;
  }
   };
```

```
int main()   {
    class_name obj(5,6);
return 0;
}
```

Output:

```
Constructor is called
Value of a: 5
Value of b: 6

Destructor is called
Value of a: 5
Value of b: 6
```

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;
class Student {
private:
        int roll_no;
string name;
      public:
        void get_details() {
           cout<<"Enter Roll No: ";
           cin>>roll_no;

           cout<<"Enter Name: ";
cin>>name;
        }
        void show_details() {
cout<<"\n\n";
           cout<<"Roll No: "<<roll_no<<endl;
            cout<<"Name: "<<name<<endl;
         }
  };
```

```
class Test {

  private:
    int marks[5];
public:
    void get_marks() {
      cout<<"Enter marks in 5 subjects: "<<endl;
for(int i=0; i<5; i++) {
        cout<<"Subject "<<i+1<<": ";
cin>>marks[i];          }
    }
    int get_total_marks() {

      int total = 0;
      for(int i=0; i<5; i++) {
total += marks[i];
      }
      return total;

    }
};



class Result : public Student, public Test {
  private:

    float percentage;
public:
    void calculate_percentage() {
int total_marks = get_total_marks();
percentage = (float)total_marks / 5.0;
    }
```

```
        void show_result() {

          show_details();
          cout<<"Total Marks: "<<get_total_marks()<<endl;
cout<<"Percentage: "<<percentage<<"%"<<endl;
       }
 };


  int main() {
    Result r;

     r.get_details();

     r.get_marks();

    r.calculate_percentage();
    r.show_result();
return 0;
 }
```

Output:

```
Enter Roll No: 44
Enter Name: Manthan
Enter marks in 5 subjects:
Subject 1: 97
Subject 2: 95
Subject 3: 96
Subject 4: 98
Subject 5: 95


Roll No: 44
Name: Manthan
Total Marks: 481
Percentage: 96.2%
```

Practical-11

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

Code:

```
#include<iostream>  Using
namespace std;  class
complex {
    int a, b;  public:
void getvalue() {

        cout << "Enter the value of Complex Numbers a,b:";
       cin >> a>>b;

     }

 complex operator+(complex ob) {
complex t;

       t.a = a + ob.a;
       t.b = b + ob.b;
return (t);

    }
  void display() {        cout << a << "+" <<
  b << "i" << "\n";

    }
  };  int
main() {
    complex obj1, obj2, result;
obj1.getvalue();
   obj2.getvalue();
```

```
     cout << "Input Values:\n";      obj1.display();
obj2.display();      cout << "Result:";      result.display();
return 0;
 }
```

Output:

```
Enter the value of Complex Numbers a,b:4
5
Enter the value of Complex Numbers a,b:6
5
Input Values:
4+5i
6+5i
Result: 10+10i
```

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>

 using namespace std;

class Shape{

public:

      double a,b;

void getData()

      {

        cin>>a>>b;

       }

      virtual void displayArea()

      {

       }

};

class Tringle : public Shape{

void displayArea()

     {

       cout<<"Area Of Tringle: "<<0.5*a*b<<endl;

    }

};
```

```
 class Rectangle: public Shape{

      void displayArea()

     {
      cout<<"Area Of Rectangle: "<<a*b<<endl;
     }
};  int
main()
{
    Tringle t;
     Shape *st = &t;
     cout<<"Enter base and altitude: ";
    st->getData();
     st->displayArea();

    Rectangle r;
Shape *sr = &r;
    cout<<"Enter length and breadth: ";
sr->getData();      sr->displayArea();
return 0;
 }
```

Output:

```
Enter base and altitude: 5
6
Area Of Tringle: 15
Enter length and breadth: 5
6
Area Of Rectangle: 30
```

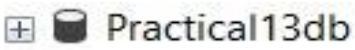
Practical-13

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

Code:

```
CREATE DATABASE Practical13db;
```

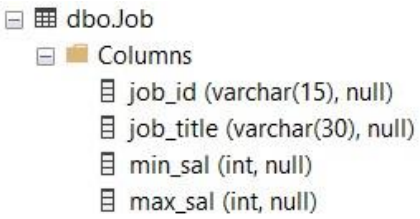
Output:



13.1 Job (job_id, job_title, min_sal, max_sal) **Code:**

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

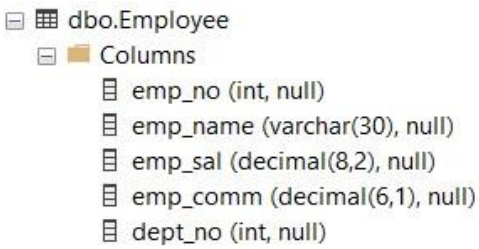
Output:



13.2 Employee (emp_no, emp_name, emp_sal, emp_comm, dept_no) **Code:**

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

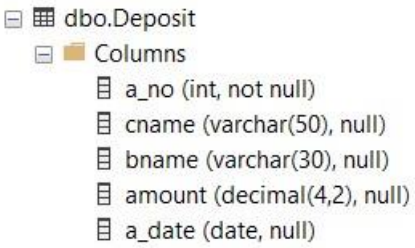
Output:



13.3 Deposit(a_no,cname,bname,amount,a_date) **Code:**

```
CREATE TABLE Deposit( a_no INT IDENTITY (1,1),
cname VARCHAR(50),   bname VARCHAR(30),
amount DECIMAL(4,2),
a_date Date
);
```

Output:

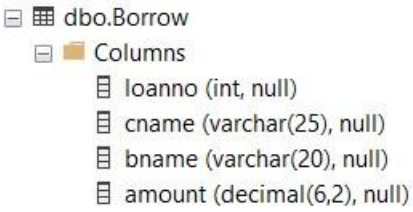


13.4 Borrow(loanno,cname,bname,amount)

Code:

```
CREATE TABLE Borrow(  loanno
 INT,
  cname VARCHAR(25),  bname
 VARCHAR(20),  amount
 DECIMAL(6,2)
);
```

Output:



Practical-14

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

14.1 Insert following values in the table **Employee.**

emp_n	emp_name	emp_sal	emp_comm	dept _no
101	Smith	800		20
102	Snehal	1600	300	25
103	Adama	1100	0	20
104	Aman	3000		15
105	Anita	5000	50000	10
106	Sneha	2450	24500	10
107	Anamika	2975		30

Code:

```
insert into Employee values(101,'Smith',800,Null,20);  insert into Employee
values(102,'Snehal',1600,300,25);  insert into Employee values(103,'Adama',1100,0,20);  insert into
Employee values(104,'Aman',3000,Null,15);  insert into Employee
values(105,'Anita',5000,50000,10); insert into Employee values(106,'Sneha',2450,24500,10);

insert into Employee values(107,'Anamika',2975,Null,30);

select * from Employee;
```

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

14.2 Insert following values in the table **Job.**

job_id	job_name	min_sal	max_sal
IT_PROG	Programmer	4000	10000

MK_MGR	Marketing manager	9000	15000
FI_MGR	Finance manager	8200	12000
FI_ACC	Account	4200	9000
LEC	Lecturer	6000	17000
COMP_OP	Computer Operator	1500	3000

Code:

```
insert into Job values('IT_PROG','Programmer',4000,10000);

insert into Job values('MK_MGR','Marketing manager',9000,15000);
        insert into Job values('FI_MGR','Finance
manager',8200,12000);    insert into Job
values('FI_ACC','Account',4200,9000);        insert into Job
values('LEC','Lecturer',6000,17000);

insert into Job values('COMP_OP','Computer Operator',1500,3000);

select * from Job;
```

Output:

	job_id	job_title	min_sal	max_sal
1	IT_PROG	Programmer	4000	10000
2	MK_MGR	Marketing manager	9000	15000
3	FI_MGR	Finance manager	8200	12000
4	FI_ACC	Account	4200	9000
5	LEC	Lecturer	6000	17000
6	COMP_OP	Computer Operator	1500	3000

14.3 Insert following values in the table **deposit**.

A_no	**cname**	**Bname**	**Amount**	**date**
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

Code:

```
insert into Deposit values(101,'Anil','andheri',7000,'01-jan-06'); insert into Deposit values(102,'sunil','virar',5000,'15-jul-06'); insert into Deposit values(103,'jay','villeparle',6500,'12-mar-06');
insert into Deposit values(104,'vijay','andheri',8000,'17-sep-06'); insert into Deposit values(105,'keyur','dadar',7500,'19-nov-06');
insert into Deposit values(106,'mayur','borivali',5500,'21-dec-06');
select * from Deposit;
```

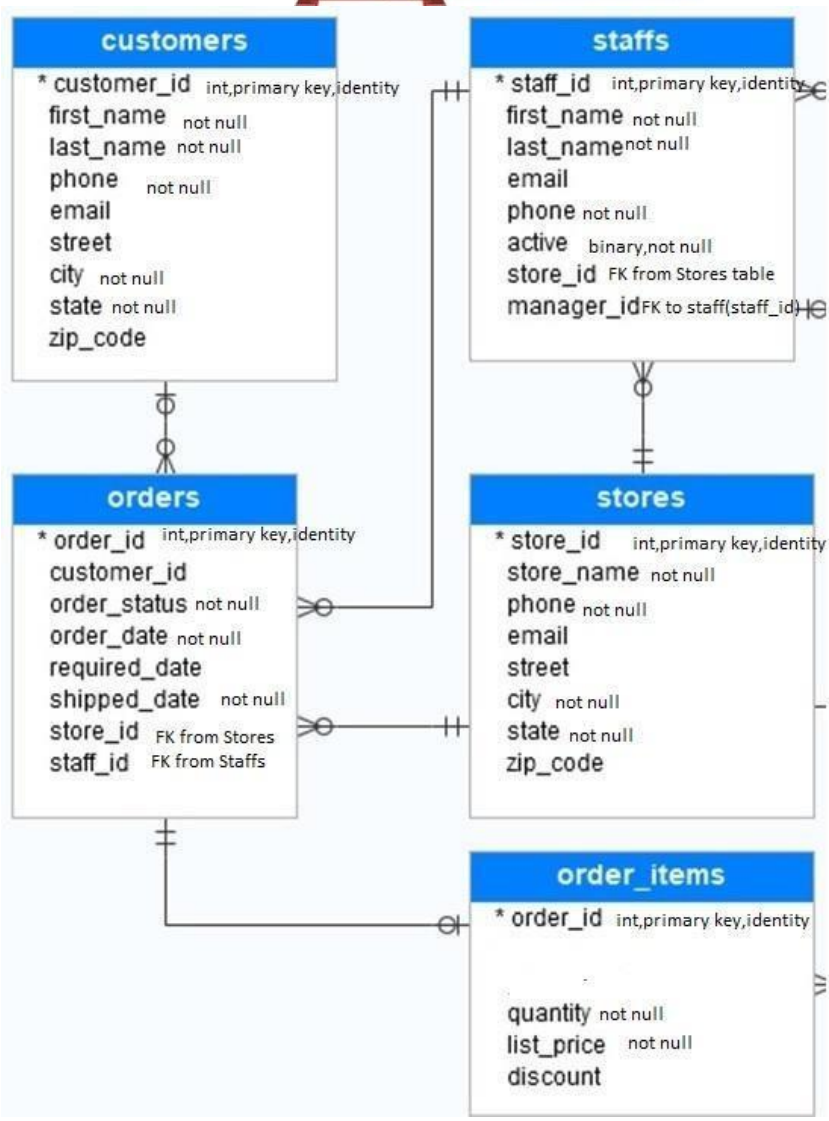
Output:

	a_no	cname	bname	amount	a_date
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-15

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.


customers
* customer_id int,primary key,identity
first_name not null
last_name not null
phone not null
email
street
city not null
state not null
zip_code
staffs
* staff_id int,primary key,identity
first_name not null
last_name not null
email
phone not null
active binary,not null
store_id FK from Stores table
manager_id FK to staff(staff_id)
orders
* order_id int,primary key,identity
customer_id
order_status not null
order_date not null
required_date
shipped_date not null
store_id FK from Stores
staff_id FK from Staffs
stores
* store_id int,primary key,identity
store_name not null
phone not null
email
street
city not null
state not null
zip_code
order_items
* order_id int,primary key,identity
quantity not null
list_price not null
discount

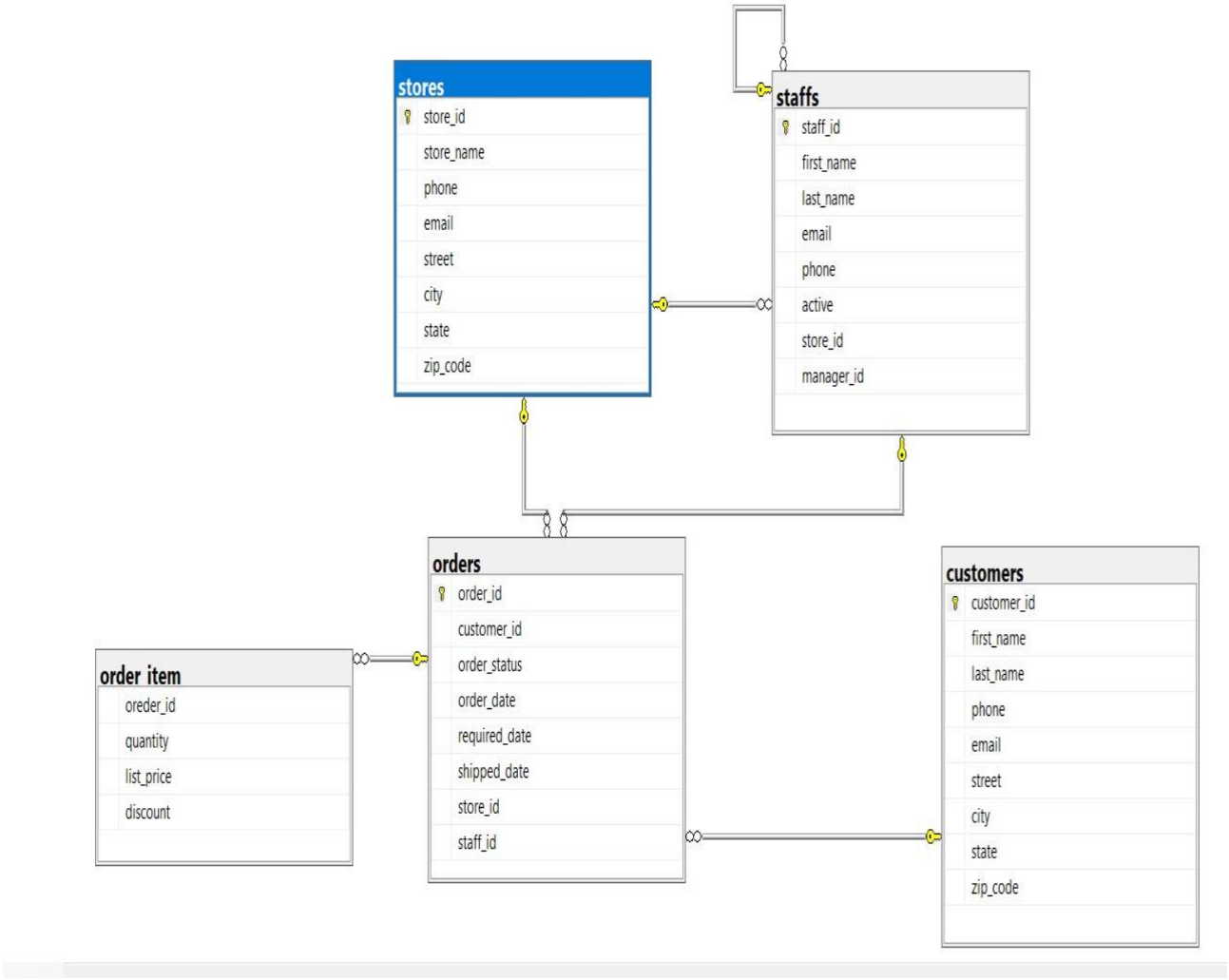
Code:

```
create table customers(
customer_id int not null primary key identity (1,1),
first_name varchar(50) not null,  last_name
varchar(50) not null,  phone int not null,  email
varchar(20),  street varchar(20),  city varchar(20)
not null,
  state varchar(20) not null,
  zip_code int
);
create table
stores(
store_id int not null primary key identity (1,1),
store_name varchar(50) not null,  phone int not
null,  email varchar(20),  street varchar(20),
city varchar(20) not null,
  state varchar(20) 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(20),  phone int not null,  active binary not null,
store_id int foreign key references stores(store_id) not null,

 manager_id int foreign key references staffs(staff_id) not null
);
create table orders(
order_id int primary key identity (1,1) not null,
customer_id int foreign key references customers(customer_id),
order_status varchar(20) 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_item(
 oreder_id int foreign key references orders(order_id) not null,
quantity int not null,
list_price int not null,  discount
int
);
```

Output:

- dbo.customers
  - Columns
    - customer_id (PK, int, not null)
    - first_name (varchar(50), not null)
    - last_name (varchar(50), not null)
    - phone (int, not null)
    - email (varchar(20), null)
    - street (varchar(20), null)
    - city (varchar(20), not null)
    - state (varchar(20), not null)
    - zip_code (int, null)
- dbo.order_item
  - Columns
    - oreder_id (FK, int, not null)
    - quantity (int, not null)
    - list_price (int, not null)
    - discount (int, null)
- dbo.stores
  - Columns
    - store_id (PK, int, not null)
    - store_name (varchar(50), not null)
    - phone (int, not null)
    - email (varchar(20), null)
    - street (varchar(20), null)
    - city (varchar(20), not null)
    - state (varchar(20), not null)
    - zip_code (int, null)
- dbo.staffs
  - Columns
    - staff_id (PK, int, not null)
    - first_name (varchar(50), not null)
    - last_name (varchar(50), not null)
    - email (varchar(20), null)
    - phone (int, not null)
    - active (binary(1), not null)
    - store_id (FK, int, not null)
    - manager_id (FK, int, not null)
- dbo.orders
  - Columns
    - order_id (PK, int, not null)
    - customer_id (FK, int, null)
    - order_status (varchar(20), not null)
    - order_date (date, not null)
    - required_date (date, null)
    - shipped_date (date, not null)
    - store_id (FK, int, null)
    - staff_id (FK, int, null)



Practical-16

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:

```
select *from deposit;

select sum(amount) from deposit;

select amount from deposit where bname='andheri';

select count(cname) from deposit;

select bname,count(cname) from deposit group by bname ;

select *from Employee;

update Employee set dept_no=10 where emp_name like '_m%';

update Employee set emp_name='panda' where emp_no=105;
```

```
select emp_no,emp_name,emp_sal,emp_sal+(emp_sal*15/100) as new_sal from
 Employee;
 select emp_no,emp_name,emp_sal,((emp_sal+(emp_sal*15/100))-emp_sal) as sub_sal
from Employee;


 alter table Employee drop column new_salary;
 select *from deposit;
 select sum(amount) from deposit;
 select amount from deposit where bname='andheri';
 select count(cname) from deposit;
  select bname,count(cname) from deposit group by bname
;

 SELECT CAST( GETDATE() AS Date ) ;
```

Output:

a_no	cname	bname	amount	a_date
101	Anil	Andheri	7000	01-june-06
102	Sunil	virar	5000	15-july-06
103	Jay	villeparle	6500	12-march-06
104	Vijay	andheri	8000	17-sept-06
105	Kayur	dadar	7500	19-nov-06
106	Mayur	borivali	5500	21-dec-06

sum(amount)
39500

amount
8000

count(cname)
6

bname	count(cname)
Andheri	1
andheri	1
borivali	1
dadar	1
villeparle	1
virar	1

emp_no	emp_name	emp_sal	emp_comm	dept_no
101	Smith	800		10
102	Snehal	1600	300	25
103	Adams	1100		20
104	Aman	3000		10
105	panda	5000	50000	10
106	Sneha	2450	24500	10
107	Anamika	2975		30

emp_no	emp_name	emp_sal	new_sal
101	Smith	800	920
102	Snehal	1600	1840
103	Adama	1100	1265
104	Aman	3000	3450
105	panda	5000	5750
106	Sneha	2450	2817
107	Anamika	2975	3421

emp_no	emp_name	emp_sal	sub_sal
101	Smith	800	120
102	Snehal	1600	240
103	Adama	1100	165
104	Aman	3000	450
105	panda	5000	750
106	Sneha	2450	367
107	Anamika	2975	446

Practical-17

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:

```
select *from employee;
select *from job;
select *from deposit;

select a_no,amount from deposit where a_date between '06-01-01'and'06-07-25';

select *from job where min_sal > 4000;

select emp_name from employee  as name_of_employee where dept_no = 20;

select emp_no,emp_name from employee where dept_no in (10,20);

select *from employee where emp_name like '__a%';
```

```
 select emp_no,emp_name,emp_sal from employee where emp_name like 'Ani%' and
length(emp_name)=5;

 select emp_no,emp_name,emp_sal from employee where emp_name like '_n____' and
  emp_comm is not null;
  select emp_no,emp_name,emp_sal from employee where emp_name like '__a%' and emp_comm
is not null;
```

Output:

emp_no	emp_name	emp_sal	emp_comm	dept_no
101	Smith	800		10
102	Snehal	1600	300	25
103	Adams	1100		20
104	Aman	3000		10
105	panda	5000	50000	10
106	Sneha	2450	24500	10
107	Anamika	2975		30

job_id	job_title	min_sal	max_sal
IT PROG	Programmer	4000	10000
MK MGR	Marketing Manager	9000	15000
FI_MGR	Finance Manager	8200	12000
FI ACC	Account	4200	9000
LEC	Lecturer	6000	17000
COMP_OP	Computer Operator	1500	3000

a_no	cname	bname	amount	a_date
101	Anil	Andheri	7000	01-june-06
102	Sunil	virar	5000	15-july-06
103	Jay	villeparle	6500	12-march-06
104	Vijay	andheri	8000	17-sept-06
105	Kayur	dadar	7500	19-nov-06
106	Mayur	borivali	5500	21-dec-06

job_id	job_title	min_sal	max_sal
MK MGR	Marketing Manager	9000	15000
FI_MGR	Finance Manager	8200	12000
FI ACC	Account	4200	9000
LEC	Lecturer	6000	17000

emp_name
Adams

emp_no	emp_name
101	Smith
103	Adams
104	Aman
105	panda
106	Sneha

emp_no	emp_name	emp_sal	emp_comm	dept_no
103	Adams	1100		20
104	Aman	3000		10
107	Anamika	2975		30

emp_no	emp_name	emp_sal
102	Snehal	1600

Practical-18

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

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>DAY-1</title>
</head>
<body>
```

```
    <h2>Manthan Thakar                                                    </h2>
      <h3><u>CONTACT</u></h3>
   <p class="p"><b>MOBILE : </b>8511694954</p>
   <p class="p"><b>EMAIL : </b>manthant555@gmail.com</p>
    <h3><u>OBJECTIVE</u></h3>
   <ul>
    <li><p>Strong in design and integration with intuitive problem-solving skills.        Proficient
in JAVA, CSS, C, C++, and HTML. Passionate about        implementing and launching new
projects. Ability to translate business        requirements into technical solutions. Looking to start
the career as an        entry-level software engineer with a reputed firm driven by
technology.</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>84%</td>

 </tr>
<tr>

    <td>HSC</td>

   <td>GSEB</td>
   <td>2022</td>
   <td>75%</td>
</tr>
</table>
<h3><u>SKILLS</u></h3>
<h3>Technical Skills</h3>
 <ul>
  <li>C,C++,JAVA,Python</li>
  <li>HTML,CSS,JAVASCRIPT</li>
 <li>SQL</li>
 <li>VS STUDIO,WINDOWS</li>
</ul>
 <h3>Soft Skills</h3>
<ul>
  <li>LEADERSHIP</li>
  <li>ADAPTABILITY</li>
```

```
  <li>TEAM WORK</li>

   <li>PROBLEM SOLVING</li>
  </ul>
  <h3><u>PROJECT</u></h3>
  <h3>BANK MANAGEMENT SYSTEM</h3>
  <ul>
   <li>USING C++</li>
  </ul>
 <h3>STUDENT MANAGEMENT SYSTEM</h3>
 <ul>
  <li>USING C</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>SPORTS</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:

Manthan Thakar

CONTACT

MOBILE : 8511694954

EMAIL : manthant555@gmail.com

OBJECTIVE

- Strong in design and integration with intuitive problem-solving skills. Proficient in JAVA, CSS, C, C++, and HTML. Passionate about implementing and launching new projects. Ability to translate business requirements into technical solutions. Looking to start the career as an entry-level software engineer with a reputed firm driven by technology.

EDUCATION

COURSE	BOARD	YEAR	MARKS
SSC	GSEB	2020	84%
HSC	GSEB	2022	75%

SKILLS

Technical Skills

- C,C++,JAVA,Python
- HTML,CSS,JAVASCRIPT
- SQL
- VS STUDIO,WINDOWS

Soft Skills

- LEADERSHIP
- ADAPTABILITY
- TEAM WORK
- PROBLEM SOLVING

PROJECT

BANK MANAGEMENT SYSTEM

- USING C++

STUDENT MANAGEMENT SYSTEM

- USING C

LANGUAGES

- GUJARATI
- HINDI
- ENGLISH

HOBBIES

- TRAVEL
- SPORTS

DECLARATION

I hereby declare that the details and information given above are complete and true to the best of my knowledge

Practical-19

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

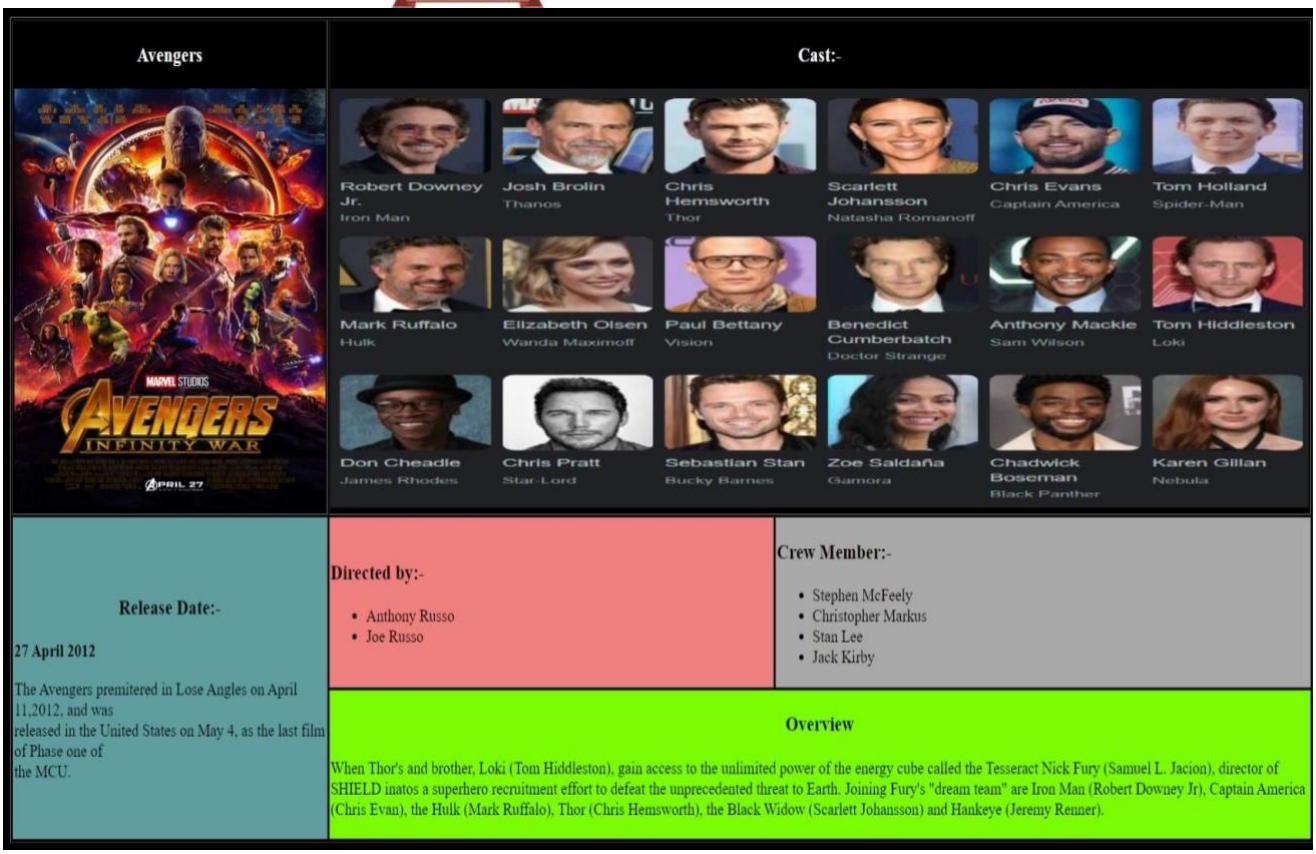
Code:

```
<!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>Avengers Poster</title>
</head>
<style>
    body{

    background-color: black;

 }

 .rd{     background-color:
cadetblue;
} .db{
   background-color: lightcoral;
}
.cm{    background-color:

 darkgray;


 }

 .ov{

    background-color: lawngreen;   }
```

```
 .ov{

     background-color: lawngreen;

}
.av{

    color: white;

}
</style>

<body>
     <table border="dotted">

         <tr>
           <td>

               <h3><b><center><span class="av">Avengers</span> </center> </b></h3>

              <img src="poster a iw.jpg" alt="Poster" height="375" width="350">
           </td>
           <td colspan="10">
              <h3><b><center class="av">Cast:-</center> </b></h3>
              <img src="cast.png" alt="cast" height="375" width="1100">

           </td>

 </tr>


         <tr>

            <td rowspan="5" class="rd">

               <center><h3><b>Release Date:-</b></h3></center>

              <p><b>27 April 2012 </b></p>
              The Avengers premitered in Lose Angles on April 11,2012, and was <br>
released in the United States on May 4, as the last film of Phase one of <br>                the
MCU.

          </td>

          <td class="db">
                <h3><b>Directed by:-</b></h3>              <h3><b>Directed by:-</b></h3>
```

```
            <ul>

                <li>Anthony Russo </li>
                <li>Joe Russo</li>
             </ul>
          </td>
          <td class="cm">
             <h3><b>Crew Member:-</b></h3>
             <ul>

                 <li>Stephen McFeely</li>
                 <li>Christopher Markus</li>
               <li>Stan Lee</li>

               <li>Jack Kirby</li>

             </ul>
             <tr>
                <td colspan="2" class="ov">
                   <h3><b><center>Overview</center></b></h3>
                   <p>When Thor's and brother, Loki (Tom Hiddleston), gain access to the unlimited
  power of the energy cube

                   called the Tesseract Nick Fury (Samuel L. Jacion), director of SHIELD inatos a
 superhero recruitment effort to
                   defeat the unprecedented threat to Earth. Joining Fury's "dream team" are Iron Man
(Robert Downey Jr), Captain
                    America (Chris Evan), the Hulk (Mark Ruffalo), Thor (Chris Hemsworth), the
Black Widow (Scarlett Johansson) and Hankeye (Jeremy Renner).</p>
                </td>
             </tr>
          </td>
             </tr>
    </table>
   </body>
```

Output:



Practical-20

Aim: Create an HTML page table and form.

Code:

```
<!DOCTYPE html>

<html>
 <body>
  <h2>Form</h2>
  <table>
    <form>
      <tr>
        <td>
          <label for="name">
Name
          </label>
        </td>
        <td><input type="text" id="name" />
        </td>
      </tr>
      <tr>
        <td><label for="email">
Email
          </label>
        </td>
         <td><input type="email" id="email" />
        </td>
      </tr>
```

```
        <tr>

            <td><label
for="telnum">                    Tel
No.
              </label>

            </td>

            <td><input type="telnum" id="telnum" />
            </td>
        </tr>
        <tr>
            <td><label for="Roll No.">

Roll No.

              </label>

            </td>

            <td><input type="number" id="rollno" />

            </td>

        </tr>
        <tr>
            <td>
              <input type="submit" value="Submit">
            </td>

            <td>

              <input type="reset" value="Reset">

            </td>

         </tr>

      </form>

   </table>

 </body>

 </html>
```

Output:

Form

Name

Email

Tel No.

Roll No.

Submit Reset

Practical-21

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

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>Practical 5</title>  </head>
<body bgcolor="powderblue">

       <form action="" method="post">

       <h1 align="center">Registration Form</h1>

       <hr>

      <table border="2" width="50%" height="50%" align="center" >
         <tr>
            <th><b>First Name :</b></th>
            <td><input type="text"  required placeholder="Enter First Name"        title="Fill the
correct data" maxlength="20" ></td>
         </tr>
         <tr>
            <th> <b>Last Name : </b></th>
            <td><input type="text" required placeholder="Enter Last Name"
maxlength="20"></td>
         </tr>
```

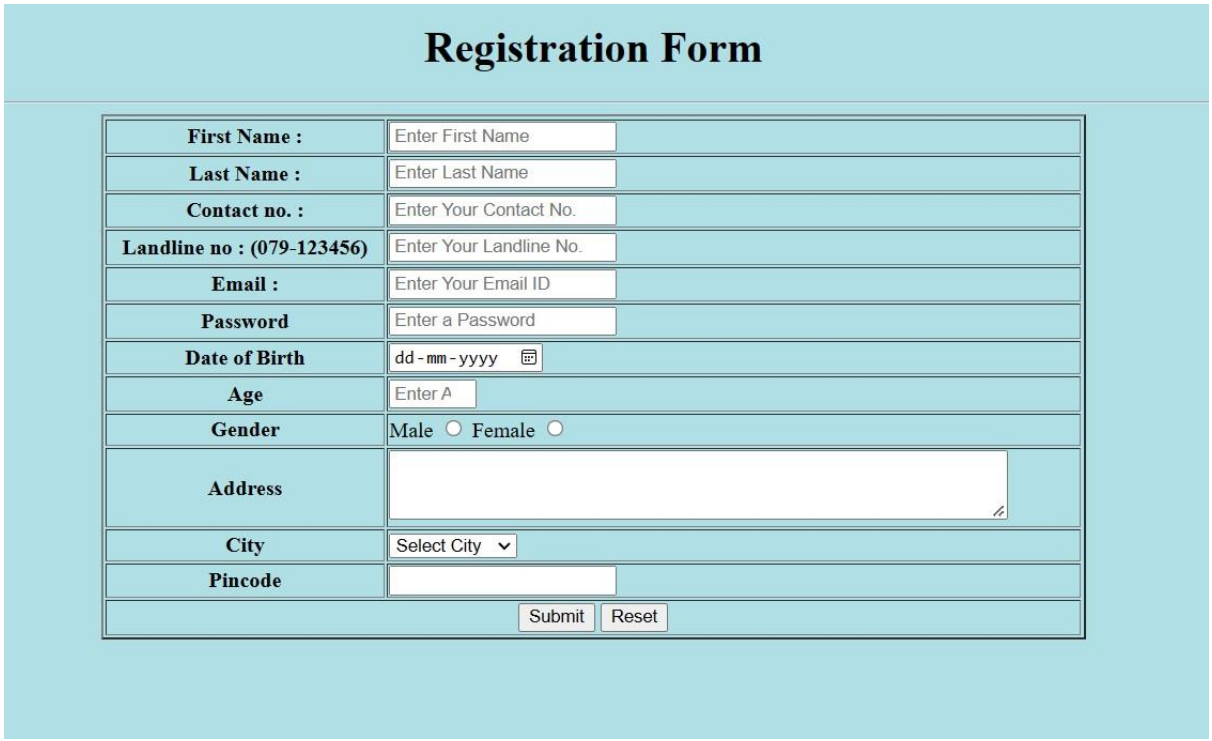
```
    <tr>
        <th> <b>Contact no. :</b></th>

        placeholder="Enter Your Contact No."></td><td><input type="text" pattern="\d{10}"
minlength="10" maxlength="10" required

      </tr>
      <tr>
        <th> <b>Landline no : (079-123456)</b></th>
        <td><input type="text" pattern="[^0-9]{3}-\d{6}" maxlength="10" required
 placeholder="Enter Your Landline No.">

      </tr>

      <tr>
        <th><b>Email :</b></th>
        <td><input type="email" required placeholder="Enter Your Email ID"
 pattern="[^@\t\r\n]+@[^@\t\r\n]+\.[^@ \t\r\n]+"></td>

      </tr>
      <tr>

        <th>Password</th>

        <td><input type="password" required placeholder="Enter a Password"
pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8}"></td>

      </tr>

      <tr>

        <th><b>Date of Birth</b></th>

        <td><input type="date" required></td>

      </tr>

      <tr>
        <th>Age</th>
        <td><input type="Number" min="18" max="40" width="600%" placeholder="Enter
 Age Between 18 to 40" required></td>

      </tr>
```

```
<tr>
    <th> <b>Gender</b></th>
```

```
            <td>Male
                <input type="radio" name="gender">
                Female
                <input type="radio" name="gender">
            </td>
        </tr>
        <tr>
            <th>Address</th>
            <td><textarea name="Address" id="" cols="60" rows="3"></textarea></td>
        </tr>
        <tr>
            <th>City</th>
            <td><select name="City" id="">
                    <option value="" disabled selected>Select City</option>
                    <option value="">Ahmedabad</option>
                    <option value="">Vadodara</option>
                    <option value="">Morbi</option>
                </select>
            </td>
        </tr>
        <tr>
            <th>Pincode</th>
            <td><input type="text" pattern="\d*" maxlength="6"></td>
        </tr>
        <tr>
            <td colspan="2" align="center"> <input type="submit"> <input type="reset"></td>



    </form>
        </tr>
  </table>  </body>
```

```
</html>
```

Output:



Practicl-22

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

Code:

HTML code:

```
<html>
  <head>
 <link type="text/css" rel="stylesheet" href="style4.css"/>
 <title>Resume</title>
 </head>
 <body>
 <div id="header">
 <p id="name">Manthan Thakar</p>
        <a href="mailto:manthant555@gmail.com" target="_blank"><p
id="email">manthant555@gmail.com</p></a>
    </div>
    <div class="left">
    </div>
    <div class="right">
          <h3>CV Highlights</h3>
          <p>
          <ul>
             <li>Currently going through CS253 – Web Application Engineering and CS61A –
Structure & Interpretation of Computer Programs</li>
             <li>Learnt basic Python, JavaScript, HTML, CSS</li>
             <li>An independent, dedicated, efficient person. These attributes are proved through
the series of courses I have taken or I am taking as of now independently through online
platforms.</li>
             <li>Good Communication Skills, Presentation Skills, attitude towards leadership,
authorisation and delegation, conflict resolution and negotiation and a very good team
worker.</li></ul>  </p>
```

```
          <ul></ul>
             <li>Interact with our engineering team to get software issues and bugs resolved</li>
             <li>Occasionally interact with customers over Skype or telephone</li>
             <li>Contribute ideas to the team on how customers can be delighted</li></ul>
```

```
        </p>
        <h3>Educational Qualifications</h3>
        <table>
          <tr id="heading">
            <td>Qualification</td>
            <td>Board</td>
            <td>Percentage / Grades</td>
            <td>Year</td>
          </tr>
          <tr>
            <td>S.S.C</td>
            <td>G.S.E.B India</td>
            <td>84.16%</td>
            <td>2020</td>
          </tr>
          <tr>
            <td>H.S.C (Science Stream)</td>
            <td>G.S.E.B India</td>
            <td>75.40%</td>
            <td>2022</td>
          </tr>
        </table>
       <p> <h3>Technical Skills</h3>
        <p>
        <ul>
          <li>
          <span id="course-name">Operating Systems:</span> DOS, Windows 98, Windows 2000, Windows XP, Windows NT, Windows Server 2003, Windows Vista, Windows 7, Macintosh Computers (OS X), Linux (Ubuntu, Fedora)</li>
```

```
            <li><span id="course-name">Programming Skills:</span>HTML, CSS, Python, JavaScript, learning C and C++</li></ul>
          </p>
        <h3>Personal Information:</h3>
          <p>
          <ul>
             <li>
          A young, determined hard and smart working person. I believe in task based roles and complete ownership of work.
             <li>
             <span id="course-name">Languages Known:</span>English, Hindi, Gujarati</li>
             <li>
             <span id="course-name">Hobbies:</span>I love reading Finance and IT related books / magazines, playing Chess, swimming, listening music, surfing Internet, self-learning through ecourses.</li> </ul>
          </p>
          <h3>Other Information</h3>
          <p>
          <ul>
             <li>
             <span id="course-name">Expected Salary:</span>As per company standards</li>
             <li>
             <span id="course-name">Area of Interest:</span>Software Development, Programming, Start-ups, Coding, App Development, Technical Support, Support Engineer, Customer Happiness, Client service, Investment Banking, Corporate Finance, Hedge Funds, Mergers & Acquisitions, Analyst, Equity Research, Business Analysis</li>
```

```
            </li>
        </p>
        <h3>Declaration</h3>
        <p>
        I hereby declare that the details furnished above are true and correct to the best of my knowledge and belief.</p>
    </div>
    <div id="footer"></div>
   </body>
</html>
```

CSS

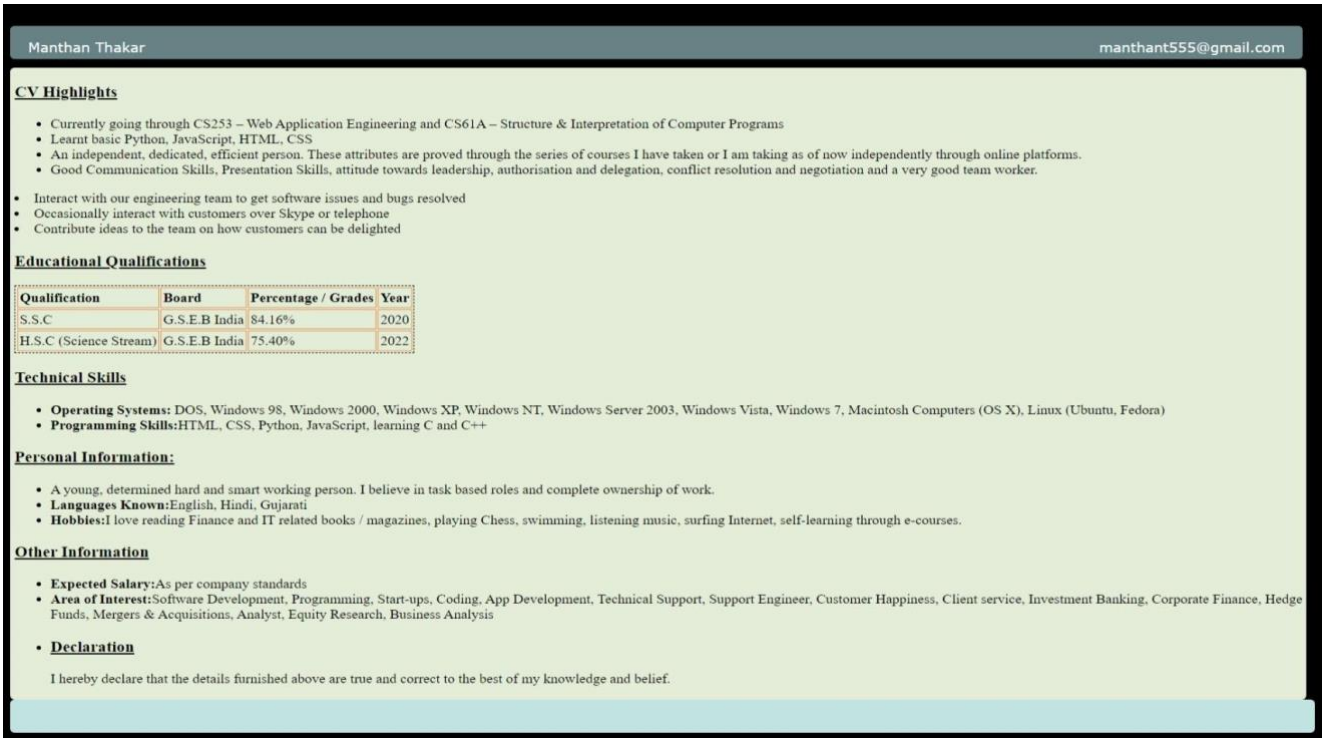
Code: div

```
{
   border-radius: 5px;
   }
#header
{
   position: fixed;
z-index: 1;
height:40px;    width:
98%;
   background-color: #668284;    margin-
bottom: 10px
   }
#name {
   float:left;    margin-
left: 20px;    padding-
bottom: 10px;    font-
size: 16px;
   font-family: Verdana, sans-serif;
```

```
    color: #ffffff;
}
#email{
    float:right;    margin-right: 20px;    padding-bottom: 10px;    font-size: 16px;
    font-family: Verdana, sans-serif;    color: #ffffff;
}
#contact
{
    margin-left:45%;    padding-bottom: 10px;    font-size: 16px;
    font-family: Verdana, sans-serif;    color: #ffffff;
    }
a:hover {
    font-weight: bold;
}
.right {
    float: left;    margin-top: 50px;    padding-left:5px;    height: auto;    width:
99%;
```

```
    background-color: #E3EDD8;
    }
#footer {
    height:40px;
clear:both;     position:
relative;
    background-color: #C1E3E1;
    }
h3 {
    text-decoration: underline;
    }
table
{
    border: 1px dashed black;
    }
td {
    padding: 2px;
    border: 1px solid #E88741;
    }
#course-name
{
    font-weight:bold;
    }
#heading
{
    font-weight:bold;   }




    }
```

Output:



Manthan Thakar manthant555@gmail.com

CV Highlights

- Currently going through CS253 – Web Application Engineering and CS61A – Structure & Interpretation of Computer Programs
- Learnt basic Python, JavaScript, HTML, CSS
- An independent, dedicated, efficient person. These attributes are proved through the series of courses I have taken or I am taking as of now independently through online platforms.
- Good Communication Skills, Presentation Skills, attitude towards leadership, authorisation and delegation, conflict resolution and negotiation and a very good team worker.

- Interact with our engineering team to get software issues and bugs resolved
- Occasionally interact with customers over Skype or telephone
- Contribute ideas to the team on how customers can be delighted

Educational Qualifications

Qualification	Board	Percentage / Grades	Year
S.S.C	G.S.E.B India	84.16%	2020
H.S.C (Science Stream)	G.S.E.B India	75.40%	2022

Technical Skills

- **Operating Systems:** DOS, Windows 98, Windows 2000, Windows XP, Windows NT, Windows Server 2003, Windows Vista, Windows 7, Macintosh Computers (OS X), Linux (Ubuntu, Fedora)
- **Programming Skills:**HTML, CSS, Python, JavaScript, learning C and C++

Personal Information:

- A young, determined hard and smart working person. I believe in task based roles and complete ownership of work.
- **Languages Known:**English, Hindi, Gujarati
- **Hobbies:**I love reading Finance and IT related books / magazines, playing Chess, swimming, listening music, surfing Internet, self-learning through e-courses.

Other Information

- **Expected Salary:**As per company standards
- **Area of Interest:**Software Development, Programming, Start-ups, Coding, App Development, Technical Support, Support Engineer, Customer Happiness, Client service, Investment Banking, Corporate Finance, Hedge Funds, Mergers & Acquisitions, Analyst, Equity Research, Business Analysis

- **Declaration**

  I hereby declare that the details furnished above are true and correct to the best of my knowledge and belief.

Practical-23

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

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>
    <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>
```

```
<!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;
```

```
            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>
```

```
<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>
```

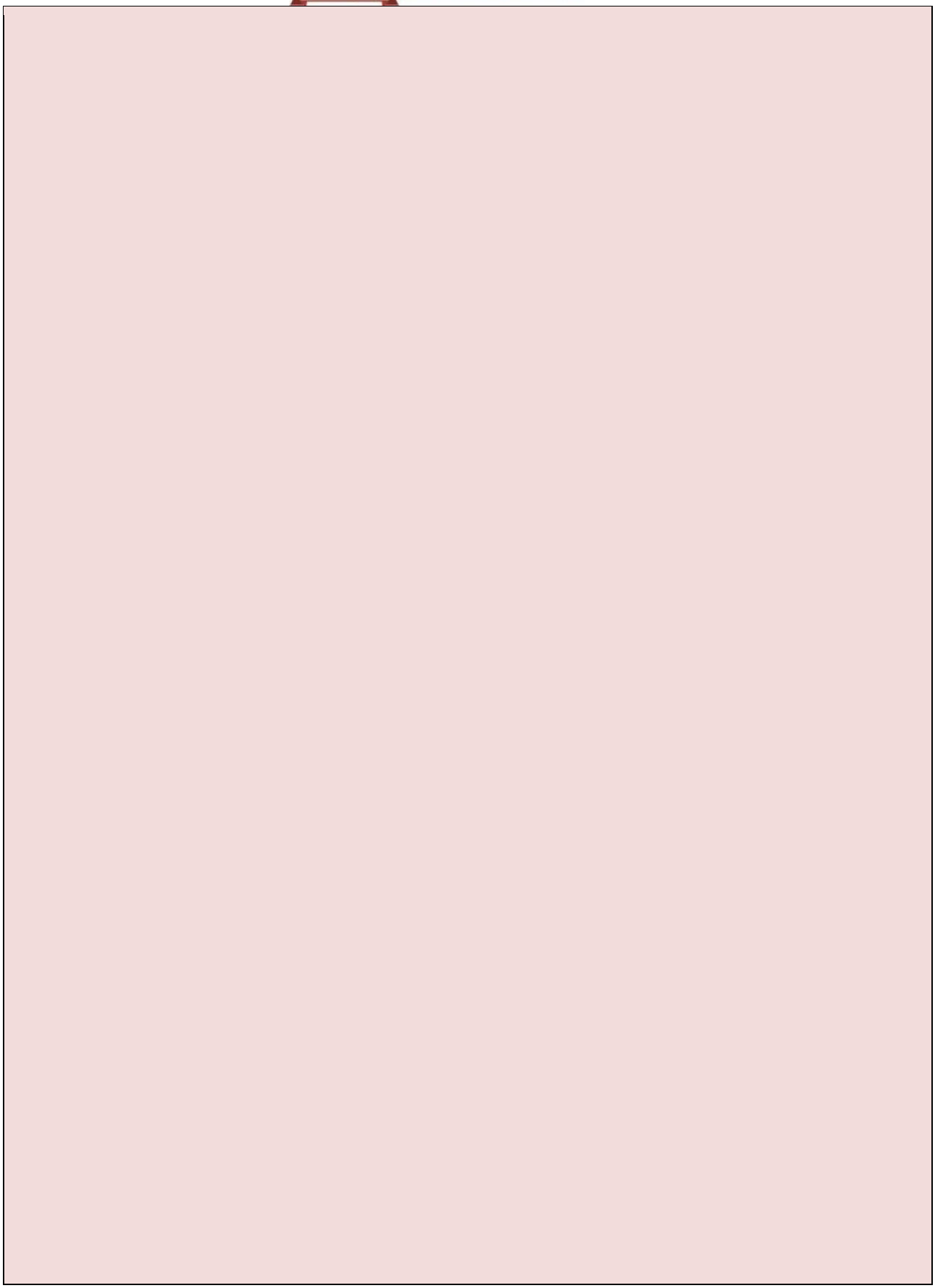
<td class="b">

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>
```

```
<html>
<head>
        <title>Demo Layout 4</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%;
```

```
                        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%;
                }
```

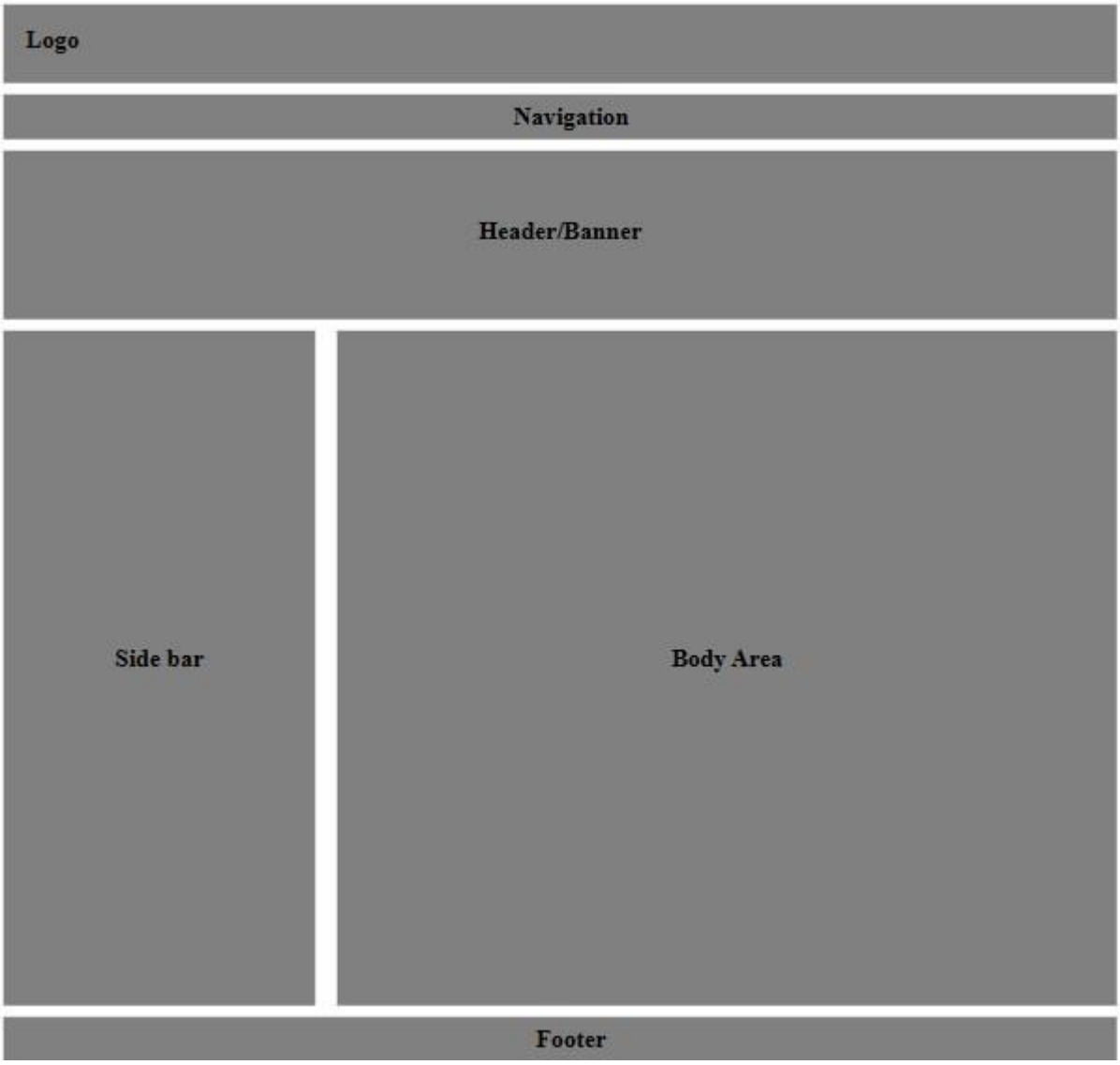


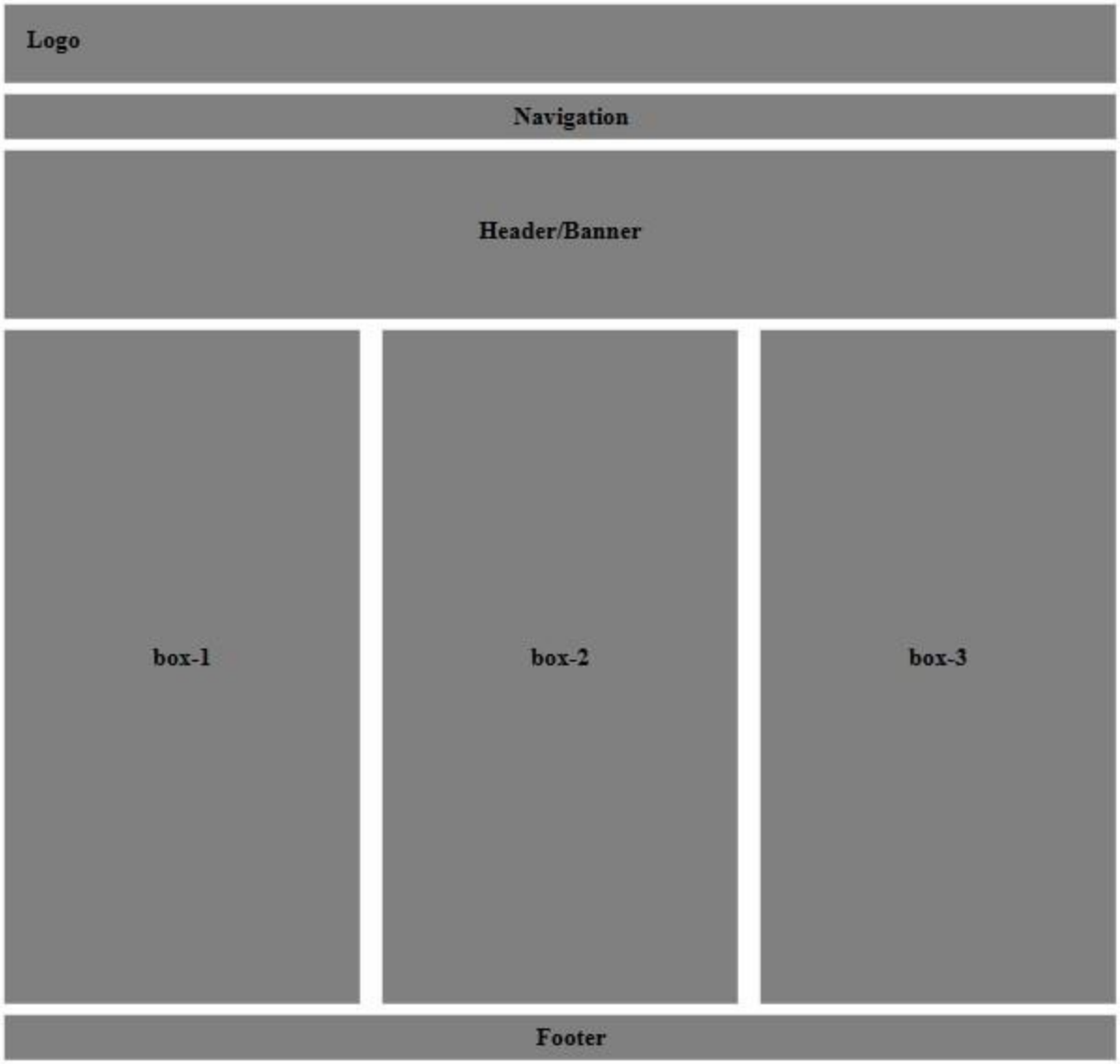
#c1 {

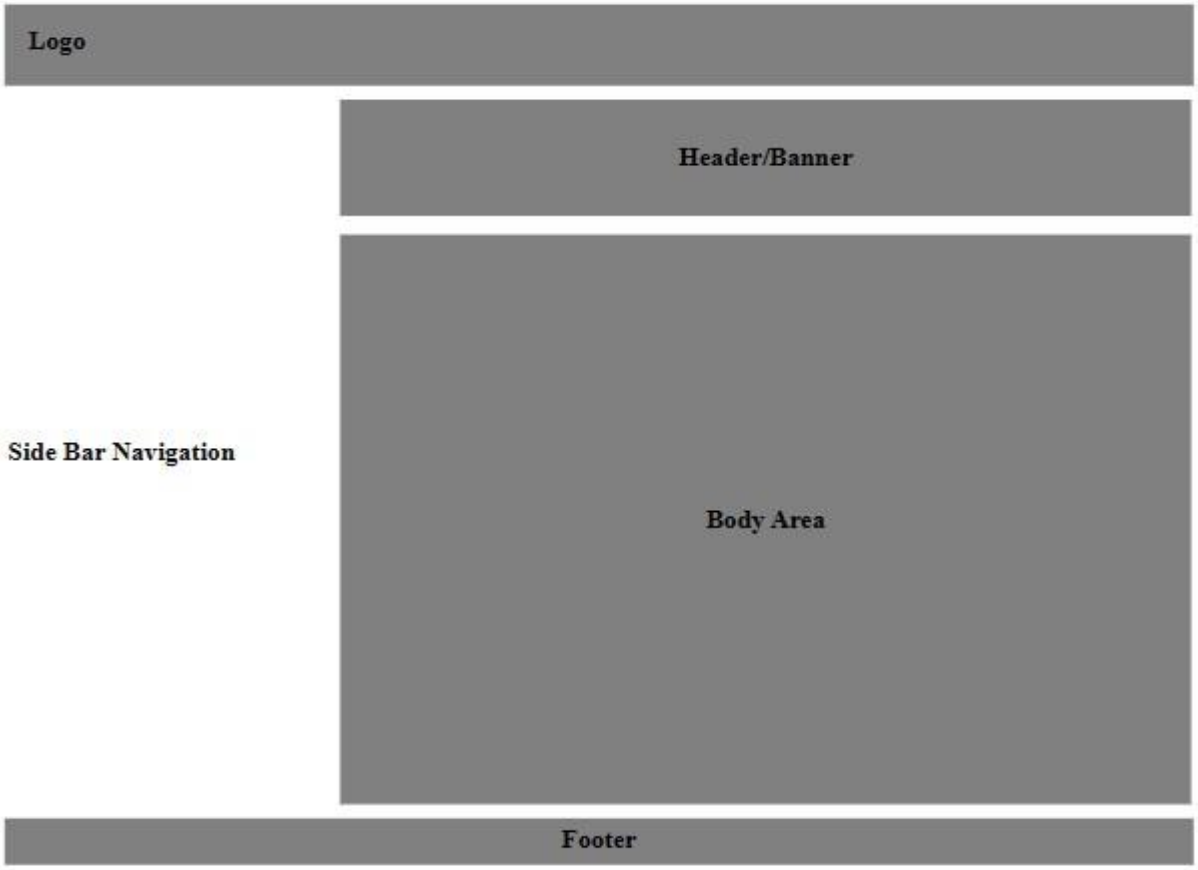
```
            #c1 {
                background-color: grey;                 height: 220px;
    width: 32%;
                  margin-right: 1%;

            }

            #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>
```

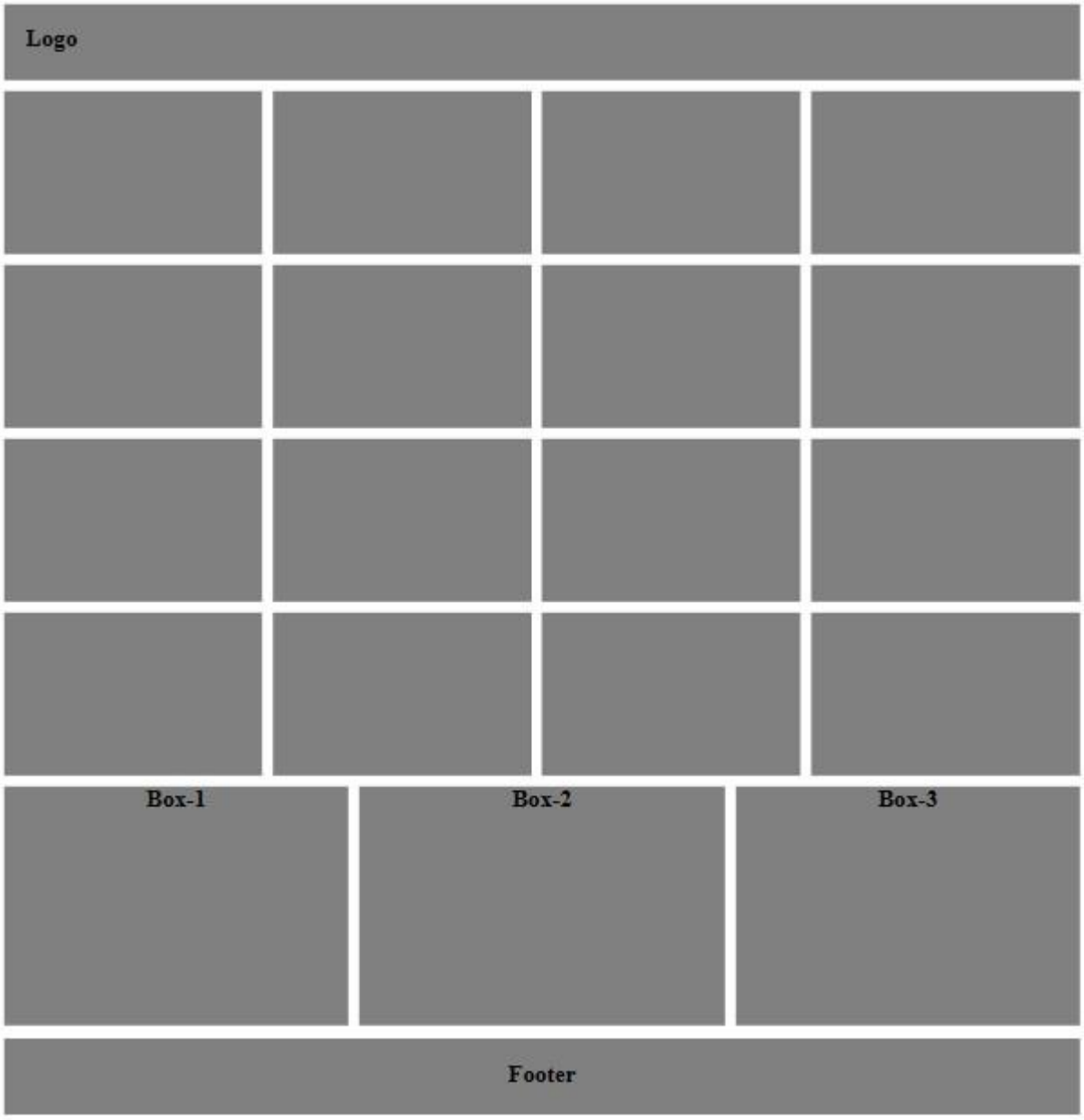
```
                <div class="B9"></div>
                <div class="B10"></div>
                <div class="B11"></div>
                <div class="B12"></div>
                <div class="B13"></div>
                <div class="B14"></div>
                <div class="B15"></div>
                <div 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.'

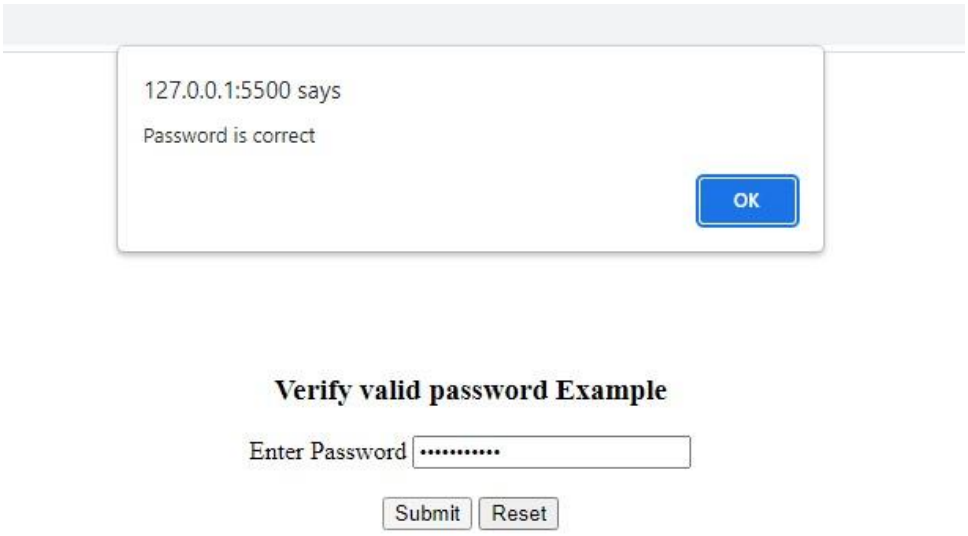
Code:

```
<html>
 <head>
<title> Verification of valid Password </title>
</head>
<script type="text/javascript">   function
verifyPassword() {
  var pw = document.getElementById("pswd").value;
  //check empty password field
if(pw == "") {
    document.getElementById("message").innerHTML = "**Fill the password please!";
return false;
  }
  const specialChars=/[`!@#$%^&*()_+\-=\[\]{};':"\\|,.<>\/?~]/;
if(!specialChars.test(pw)){
   document.getElementById("Message").innerHTML="**Include atlest one
Special Character!";    return false;
  }
  if(pw.charAt(0)!=pw.charAt(0).toUpperCase()){
alert("First Latter nMust be upper case:");    return
false;
  }
 //minimum password length validation
if(pw.length < 8) {
    document.getElementById("message").innerHTML = "**Password length must be atleast 8
characters";
```

```
characters";
return false;
  }
  //maximum length of password validation
if(pw.length > 15) {
    document.getElementById("message").innerHTML = "**Password length must not exceed 15 characters";
return false;
  }
else {
    alert("Password is correct");
  }
}
</script>
<style>    h3{
     margin-top: 15%;
   }
  </style>

<body>
<center>
<h3> Verify valid password Example </h3>

<form onsubmit ="return verifyPassword()">
<!-- Enter Password -->
<td> Enter Password </td>
<input type = "password" id = "pswd" value = "">
<span id = "message" style="color:red"> </span> <br><br>

<!-- Click to verify valid password -->
```

```
<input type = "submit"                                    value = "Submit">
  <!-- Click to reset fields -->
<button type = "reset" value = "Reset" >Reset</button>
</form>
</center>
</body>
</html>
```

Output:

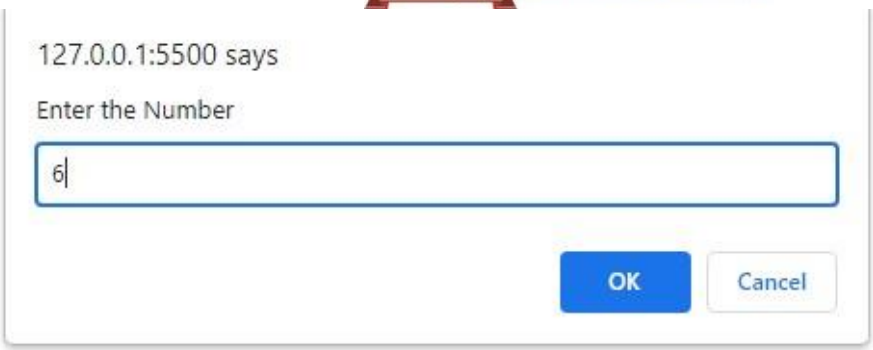


Practical-25

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>
        let a=prompt("Enter the Number");
if(a%2==0){
        document.write("<h1>Number is Even");
        }
else{
        document.write("<h1>Number is Odd");
        }
    </script>
</body>
</html>
```



Number is Even

Output:

Practical-26

Aim: Create a page and access the LocationAPI. **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>Location</title>
</head>
<body>
   <p>Click  On the Try it Button
   </p>
   <button onclick="getLocation()">Try it</button>
   <p id="demo"></p>
   <script>
      function getLocation(){
         var x=document.getElementById("demo");
if(navigator.geolocation){
            navigator.geolocation.getCurrentPosition(showPosition);
function showPosition(position){

x.innerHTML="Latitude:"+position.coords.latitude+"<br>Longitude:"+position.coords.longitud
e;
            }
         }
      }
   </script>

</body>
```

Click On the Try it Button

Try it

Latitude:23.03622847856012
Longitude:72.65729987786963

Output:

Practical-28

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>
```

The XMLHttpRequest Object

Change Content

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.