PROGRAM- 1

1. Input three numbers and find the largest.

CODE

```
#include <iostream>
using namespace std;
int main()
{
        int num1,num2,num3,largest;
        cout << "Enter three numbers : ";
        cin >> num1 >> num2 >> num3;
       if (num1 >= num2 && num1 >= num3)
                largest = num1;
        else if (num2 >= num1 && num2 >= num3)
                largest = num2;
        else
                largest = num3;
        cout << "The largest number is = " << largest;
        return 0;
}
```

OUTPUT / LAB- 1

```
/tmp/fnL7rcL2QD.o
Enter three numbers : 23
78
45
The largest number is = 78

=== Code Execution Successful ===
```

PROGRAM- 2

2. Input the principal amount, type of account (C for current a/c or S for SB a/c) and number of years, and display the amount of interest. Rate of interest for Current a/c is 8.5% and that of SB a/c is 6.5%.

CODE

```
#include <iostream>
using namespace std;
int main()
{
	int years;
	char accountType;
	float principal, rate, interest;
	cout << "Enter Principal Amount : ";
	cin >> principal;
	cout << "Enter Account type (C for Current, S for Savings) : ";
	cin >> accountType;
	cout << "Enter Number of years: ";
	cin >> years;
	switch (accountType)
	{
		case 'C': rate = 8.5; break;
		case 'c': rate = 8.5; break;
		case 'S': rate = 6.5; break;
		case 's': rate = 6.5; break;
		default:
		cout << "Invalid Account type.";
		return 1;
	}
	interest = (principal * rate * years) / 100;
	cout << "Interest Earned: " << interest;
	return 0;
}
```

OUTPUT / LAB- 2

```
Output
/tmp/gpOR7KVGHm.o
Enter Principal Amount: 9000
Enter Account type (C for Current, S for Savings) : S
Enter Number of years: 3
Interest Earned: 1755


=== Code Execution Successful ===
```

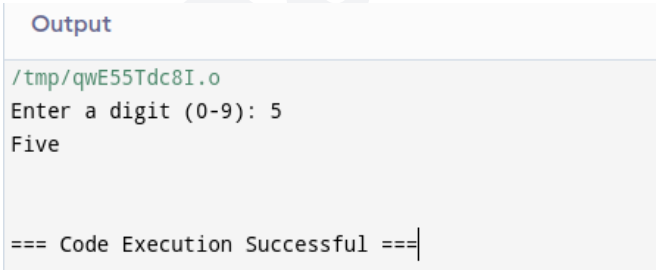
PROGRAM- 3

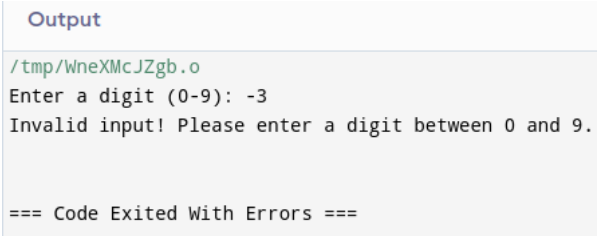
3. Input a digit and display the same word. (Zero for 0, One for 1, …., Nine for 9)

CODE

```
#include <iostream>
using namespace std;
int main()
{
        int digit;
        cout << "Enter a digit (0-9): ";
        cin >> digit;
        switch (digit)
        {
        case 0: cout << "Zero"; break;
        case 1: cout << "One" ; break;
        case 2: cout << "Two"; break;
        case 3: cout << "Three"; break;
        case 4: cout << "Four"; break;
        case 5: cout << "Five"; break;
        case 6: cout << "Six"; break;
        case 7: cout << "Seven"; break;
        case 8: cout << "Eight"; break;
        case 9: cout << "Nine"; break;
        default:cout << "Invalid input! Please enter a digit between 0 and 9.";
        }
        return 0;
}
```

OUTPUT / LAB- 3





PROGRAM - 4

4. Find the sum of the first N natural numbers without using formula.

CODE

```
#include <iostream>
using namespace std;
int main()
{
        int n,i,sum = 0;
        cout << "Enter the value of N : ";
        cin >> n;
        for (i = 1; i <= n; i++)
                sum = sum+i;

        cout << "The sum of the first " << n << " natural numbers is: " << sum;
        return 0;
}
```

OUTPUT / LAB- 4

```
/tmp/oS4wpOqb5J.o
Enter the value of N : 6
The sum of the first 6 natural numbers is: 21

=== Code Execution Successful ===
```

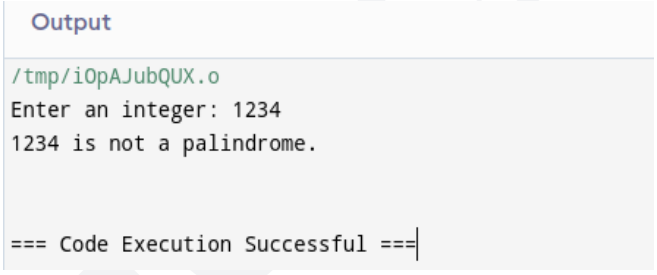
PROGRAM - 5

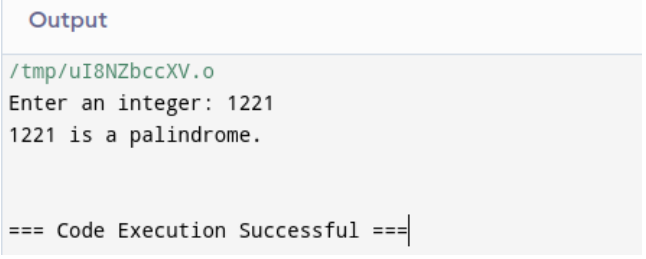
5. Input an integer number and check whether it is palindrome or not.

CODE

```
#include <iostream>
using namespace std;
int main()
{
	int num, rev = 0, oldnum;
	cout << "Enter an integer: ";
	cin >> num;
	oldnum = num;
	while (num != 0)
	{
		rev = rev * 10 + num % 10;
		num = num/10;
	}
	if (oldnum == rev)
		cout << oldnum << " is a palindrome.";
	else
		cout << oldnum << " is not a palindrome.";
	return 0;
}
```

OUTPUT / LAB- 5





PROGRAM - 6

6. Display the multiplication table of a number having 12 rows.

CODE

```
#include <iostream>
using namespace std;
int main()
{
	int num,i;
	cout << "Enter a number : ";
	cin >> num;
	cout << "Multiplication table of " << num << ":" << endl;
	for ( i = 1; i <= 12; i++)
		cout << i << " x " << num << " = " << num * i << endl;
	return 0;
}
```

OUTPUT / LAB- 6

```
/tmp/PCjyqwIivj.o
Enter a number : 5
Multiplication table of 5:
1 x 5 = 5
2 x 5 = 10
3 x 5 = 15
4 x 5 = 20
5 x 5 = 25
6 x 5 = 30
7 x 5 = 35
8 x 5 = 40
9 x 5 = 45
10 x 5 = 50
11 x 5 = 55
12 x 5 = 60
```

PROGRAM- 7

7. Create an array to store the heights of 10 students and find the largest value.

CODE

```
#include <iostream>
using namespace std;
int main()
{
        int i;
        float heights[10],largeH;
        cout <<"Enter the heights of 10 Students :" << endl;
        for ( i = 0; i < 10; i++)
        {
                cout << "Student " << i + 1 << ": ";
                cin >> heights[i];
        }
        largeH = heights[0];
        for ( i = 1; i < 10; i++)
        {
                if (heights[i] > largeH)
                        largeH = heights[i];
        }
        cout << "The largest height is: " << largeH;
        return 0;
  }
```

OUTPUT / LAB- 7

```
/tmp/mXavLAMmci.o
Enter the heights of 10 Students :
Student 1: 154.7
Student 2: 145.9
Student 3: 160
Student 4: 171.4
Student 5: 159
Student 6: 149
Student 7: 152.5
Student 8: 156
Student 9: 173
Student 10: 166
The largest height is: 173

=== Code Execution Successful ===
```

PROGRAM - 8

8. Find the length of a string without using strlen() function.

CODE

```
#include <iostream>
using namespace std;
int main()
{
        int i,count=0;
        char str[50];
        cout<<"Enter any string  : ";
        cin>>str;
        for(i=0;str[i]!='\0';i++)
                count++;
        cout<<"Length of Word is : "<<count;
        return 0;
}
```

OUTPUT / LAB- 8

```
/tmp/qLuLSYGlC5.o
Enter any string  : teachbook
Length of Word is : 9

=== Code Execution Successful ===
```

PROGRAM - 9

9. Find the factorial of a number with the help of a user-defined function.

CODE

```
#include <iostream>
using namespace std;
int factorial(int n)
{
        int fact=1,i;
        for( i = 1; i <= n; ++i )
                fact = fact * i;
        return fact;
}
int main()
{
        int n;
        cout << "Enter a Number: ";
        cin >> n;
        cout << "Factorial of " << n << " = " << factorial(n);
        return 0;
}
```

OUTPUT

```
/tmp/LEMHBcdpr9.o
Enter a Number: 5
Factorial of 5 = 120

=== Code Execution Successful ===
```

PROGRAM - 10

10. Input two numbers and swap them with the help of a user defined function.

CODE

```
#include <iostream>
using namespace std;
void swap (int& num1, int& num2)
{
        int temp = num1;
        num1 = num2;
        num2 = temp;
}
int main()
{
        int num1, num2;
        cout << "Enter the First number: ";
        cin >> num1;
        cout << "Enter the Second number: ";
        cin >> num2;
        cout << "Before Swapping : \nNumber 1 = "<< num1 << "\nNumber 2 = " << num2;
        swap(num1, num2);
        cout << "\nAfter Swapping : \nNumber 1 = " << num1 << "\nNumber 2 = " << num2;
        return 0;
}
```

OUTPUT

```
/tmp/rDtHgOsd2i.o
Enter the First number: 123
Enter the Second number: 456
Before Swapping :
Number 1 = 123
Number 2 = 456
After Swapping :
Number 1 = 456
Number 2 = 123

=== Code Execution Successful ===
```

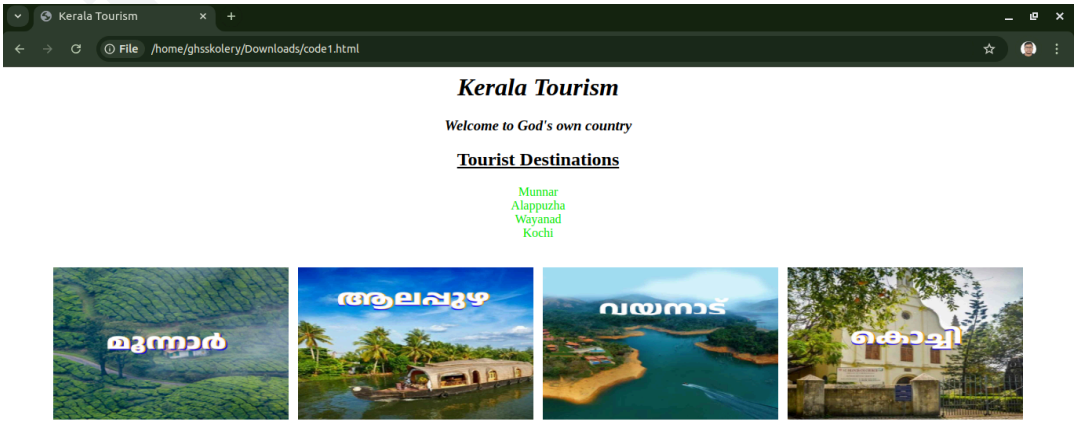
LAB- 11

1. Design a simple and attractive web page for Kerala Tourism. It should contain features like background colour/image, headings, text formatting and font tags, images, etc.

SOURCE CODE

```
<HTML>
        <HEAD>
                <TITLE> Kerala Tourism </TITLE>
        </HEAD>
        <BODY background="kerala.jpg">
                <CENTER>
                        <I> <H1> Kerala Tourism </H1>
                            <H3> Welcome to God's own country </H3> </I>
                        <U> <H2> Tourist Destinations </H2> </U>
                        <FONT Face="Arial"  Color="Red" size="5" >
                                Munnar<BR>
                                Alappuzha<BR>
                                Wayanad<BR>
                                Kochi
                        </FONT><BR><BR>
                        <IMG Src="munnar.jpg" width="300" height="200">  
                        <IMG Src="alappuzha.jpg" width="300" height="200">  
                        <IMG Src="wayanad.jpg" width="300" height="200">  
                        <IMG Src="kochi.jpg" width="300" height="200">
                </CENTER>
        </BODY>
</HTML>
```

OUTPUT



LAB- 12

2. Design a simple web page about your school. Create another web page named address.html containing the school address.
   Give links from the school page to address.html.

SOURCE CODE

Create two separate file and save in same location
[Open Geany Editor and save the first file with name **School.html**]

School.html :

```
<HTML>
        <HEAD>
                <TITLE>My School</TITLE>
        </HEAD>
        <BODY Bgcolor= "yellow">
                <Center>
                        <H1 Align =”Center”>Govt. HSS Kolery</H1>
                        <H3>Our School has the following courses</H3>
                            <b>1.Biology Science <BR>
                                2.Computer Application Humanities <BR>
                                3.Computer Application Commerce <BR><BR><BR>
                            </b>
                         <A Href= "address.html"> Address of the School</A>
                </Center>
        </BODY>
</HTML>
```

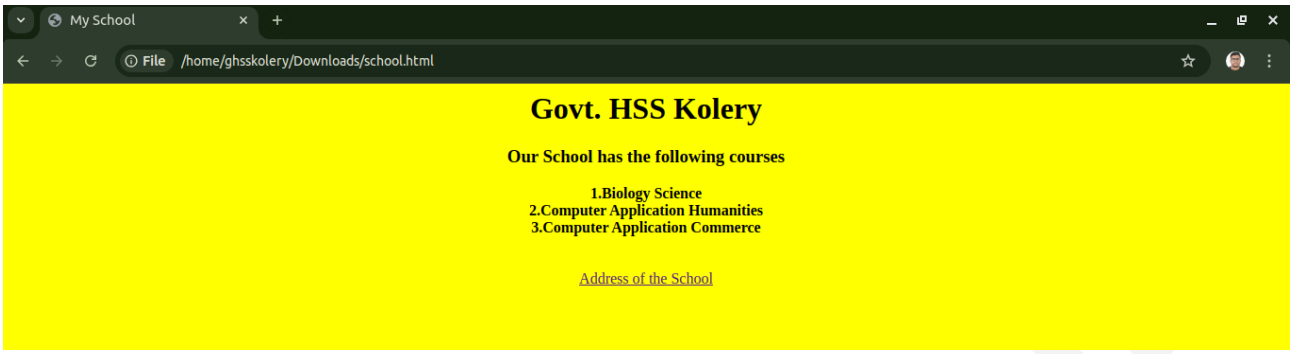
[Save the file again, Open a new File and save with second file name **address.html** in the same folder]

address.html :

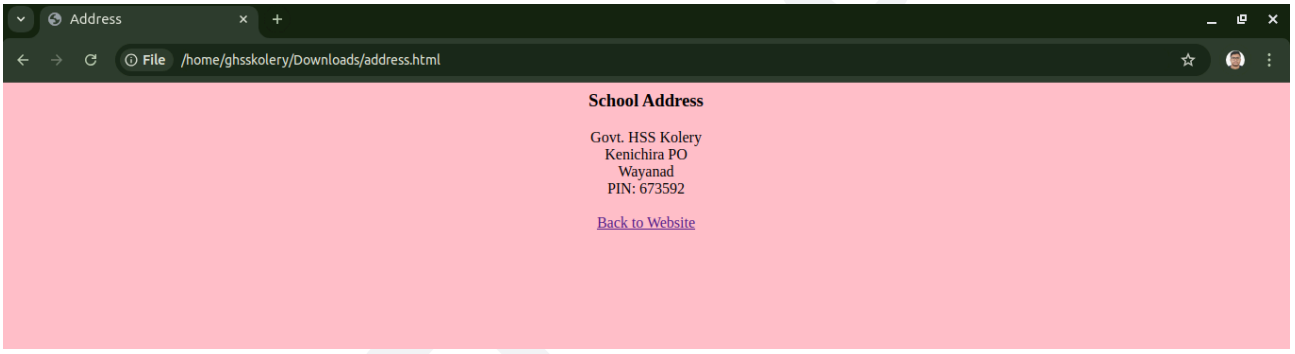
```
<HTML>
        <HEAD>
                <TITLE>Address</TITLE>
        </HEAD>
        <BODY Bgcolor= "pink">
           <CENTER>
                <H3>School Address</H3>
                        Govt. HSS Kolery<BR> Kenichira PO<BR>
                        Wayanad<BR> PIN: 673592<BR><BR>
                        <A Href="school.html"> Back to Website</A>
           </CENTER>
        </BODY>
</HTML>
```

OUTPUT

School.html :



address.html :



LAB- 13

3. Design a webpage as shown below using appropriate list tags.

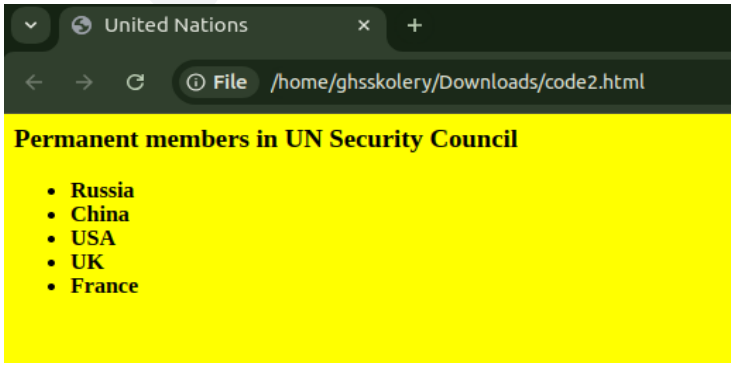
Permanent members in the UN Security Council

- Russia
- China
- USA
- UK
- France

SOURCE CODE

```
<HTML>
        <HEAD>
                <TITLE> United Nations </TITLE>
        </HEAD >
        <BODY Bgcolor= "yellow">
                <H3>Permanent members in UN Security Council </H3>
                <B>
                <UL>
                        <LI>Russia</LI>
                        <LI>China</LI>
                        <LI>USA</LI>
                        <LI>UK</LI>
                        <LI>France</LI>
                </UL>
                </B>
        </BODY>
</HTML>
```

OUTPUT



LAB- 14

4. Design a webpage as shown below using appropriate list tags.

Wildlife Sanctuaries in Kerala

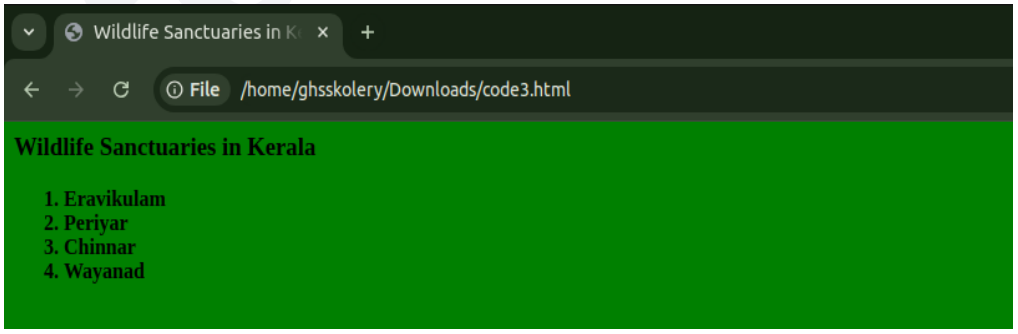
1. Eravikulam
2. Periyar
3. Chinnar
4. Wayanad

SOURCE CODE

```
<HTML>
    <HEAD>
        <TITLE> Wildlife Sanctuaries in Kerala </TITLE>
    </HEAD >

    <BODY Bgcolor= "green">
        <H3>Wildlife Sanctuaries in Kerala</H3>
        <B>
            <OL>
                <LI>Eravikulam</LI>
                <LI>Periyar</LI>
                <LI>Chinnar</LI>
                <LI>Wayanad</LI>
            </OL>
        </B>
    </BODY>
</HTML>
```

OUTPUT



LAB- 15

5. Design a web page containing a table as shown below.

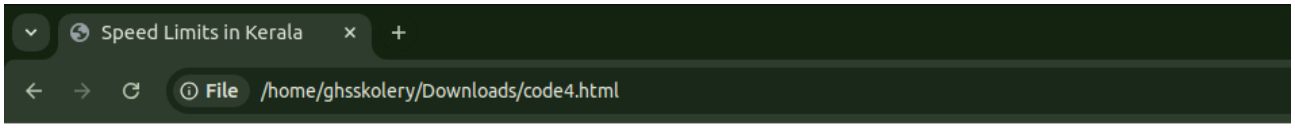
Speed Limits in Kerala

Vehicles	Near School *(In Km/hour)*	Within Corporation/ Municipality *(In Km/hour)*	State Highway *(In Km/hour)*
Motor Cycle	30	50	50
Motor Car	30	50	80
Light motor vehicles	30	50	80
Heavy motor vehicles	30	40	65

SOURCE CODE

```
<HTML>
	<HEAD>
		<TITLE>Speed Limits in Kerala</TITLE>
	</HEAD>
	<BODY>
		<H2 ALIGN="CENTER">
		<TABLE BORDER=1 ALIGN="CENTER">
			<CAPTION>Speed Limits in Kerala</CAPTION>
			<TR>
				<TH >Vehicles </TH>
				<TH>Near School <br> (In Km/hour)</TH>
				<TH>Within Corporation/<br>Municipality<br>(InKm/hour)</TH>
				<TH>State Highway <br> (In Km/hour)</TH>
			</TR>
			<TR ALIGN="CENTER">
				<TD>Motor Cycle</TD>
				<TD> 30</TD>
				<TD> 50</TD>
				<TD> 50</TD>
			</TR>
			<TR ALIGN="CENTER">
				<TD>Motor Car</TD>
				<TD> 30</TD>
				<TD> 50</TD>
				<TD> 80 </TD>
			</TR>
			<TR ALIGN="CENTER">
				<TD>Light Motor Vehicles</TD>
				<TD> 30</TD>
				<TD> 50</TD>
				<TD> 80 </TD>
			</TR>
			<TR ALIGN="CENTER">
				<TD>Heavy Motor Vehicles</TD>
				<TD> 30</TD>
				<TD>40</TD>
				<TD> 65</TD>
			</TR>
		</TABLE>
	</BODY>
</HTML>
```

OUTPUT

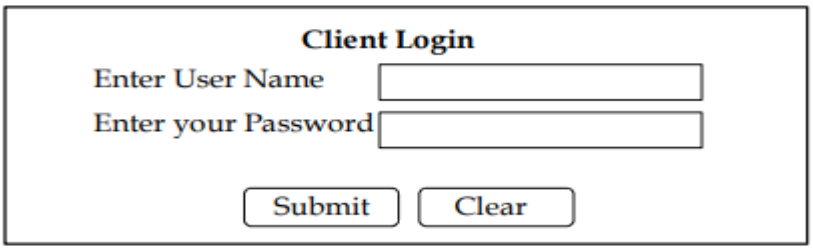


Speed Limits in Kerala

Vehicles	Near School (In Km/hour)	Within Corporation/ Municipality (InKm/hour)	State Highway (In Km/hour)
Motor Cycle	30	50	50
Motor Car	30	50	80
Light Motor Vehicles	30	50	80
Heavy Motor Vehicles	30	40	65

LAB- 16

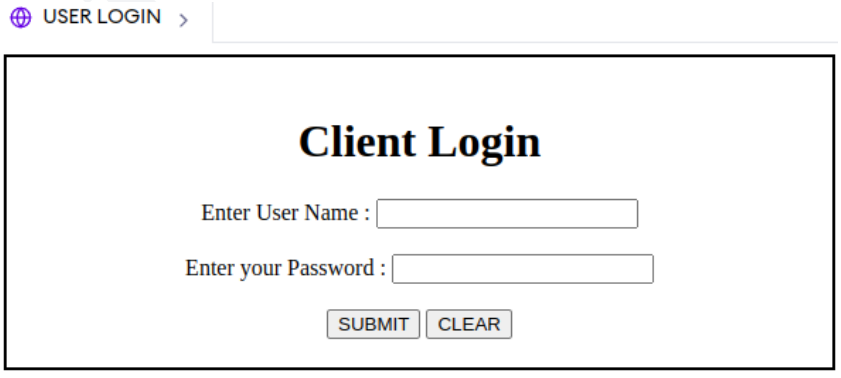
6. Design a simple web page as shown below.



SOURCE CODE

```
<HTML>
<HEAD>
   <TITLE>USER LOGIN</TITLE>
</HEAD>
<BODY>
<FORM STYLE="BORDER: 2PX SOLID;PADDING: 20PX;">
        <CENTER>
        <H1>Client Login</H1>
        Enter User Name :
        <INPUT TYPE="TEXT" ID="USERNAME" NAME="USERNAME">
        <BR><BR>
        Enter your Password :
        <INPUT TYPE="PASSWORD" ID="PASSWORD" NAME="PASSWORD" REQUIRED>
        <BR><BR>
        <INPUT TYPE="SUBMIT" VALUE="SUBMIT">
        <INPUT TYPE="RESET" VALUE="CLEAR">
       </CENTER>
</FORM>
</BODY>
</HTML>
```

OUTPUT



Part - C

Database queries using MySQL

LAB- 17

1. Create a table Student with the following fields and insert at least 5 records into the table except for the column Total.

Roll_Number	Integer	Primary key
Name	Varchar (25)	
Batch	Varchar (15)	
Mark1	Integer	
Mark2	Integer	
Mark3	Integer	
Total	Integer	

a. Update the column Total with the sum of Mark1, Mark2 and Mark3.
b. List the details of students in the Commerce batch.
c. Display the name and total marks of students who have failed (Total < 90).
d. Display the name of students in alphabetical order and in batch based.

SQL CODE

➤ CREATE TABLE Student (Roll_Number INT PRIMARY KEY, Name VARCHAR(25), Batch VARCHAR(15), Mark1 INT, Mark2 INT, Mark3 INT, Total INT);

➤INSERT INTO Student (Roll_Number,Name,Batch,Mark1,Mark2,Mark3) VALUES
(1, 'Alice', 'Science', 85, 90, 92),
(2, 'Bob', 'Commerce', 78, 82, 75),
(3, 'Charlie', 'Humanities', 95, 88, 90),
(4, 'David', 'Commerce', 80, 75, 85),
(5, 'Eve', 'Humanities', 23, 34, 25);

➤ UPDATE Student SET Total = Mark1 + Mark2 + Mark3;

➤ SELECT * FROM Student WHERE Batch = 'Commerce';

➤ SELECT Name, Total FROM Student WHERE Total < 90;

➤ SELECT Name, Batch FROM Student ORDER BY Batch, Name;

OUTPUT

Query : 1

Student

Roll_Number	Name	Batch	Mark1	Mark2	Mark3	Total
empty						

Query : 2

Student

Roll_Number	Name	Batch	Mark1	Mark2	Mark3	Total
1	Alice	Science	85	90	92	
2	Bob	Commerce	78	82	75	
3	Charlie	Humanities	95	88	90	
4	David	Commerce	80	75	85	
5	Eve	Humanities	23	34	25	

Query : 3

Student

Roll_Number	Name	Batch	Mark1	Mark2	Mark3	Total
1	Alice	Science	85	90	92	267
2	Bob	Commerce	78	82	75	235
3	Charlie	Humanities	95	88	90	273
4	David	Commerce	80	75	85	240
5	Eve	Humanities	23	34	25	82

Query : 4

Output

Roll_Number	Name	Batch	Mark1	Mark2	Mark3	Total
2	Bob	Commerce	78	82	75	235
4	David	Commerce	80	75	85	240

Query : 5

Output

Name	Total
Eve	82

Query : 6

Output

Name	Batch
Bob	Commerce
David	Commerce
Charlie	Humanities
Eve	Humanities
Alice	Science

LAB- 18

2. Create a table Employee with the following fields and insert at least 5 records into the table except the column Gross_pay and DA.

Emp_code	Integer	Primary key
Emp_name	Varchar (20)	
Designation	Varchar (25)	
Department	Varchar (25)	
Basic	Decimal (10,2)	
DA	Decimal (10,2)	
Gross_pay	Decimal (10,2)	

a. Update DA with 75% of Basic.
b. Display the details of employees in Purchase, Sales and HR departments.
c. Update the Gross_pay with the sum of Basic and DA.
d. Display the details of employees with gross pay below 10000.

SQL CODE

➤ CREATE TABLE Employee (Emp_code INT PRIMARY KEY, Emp_name VARCHAR(20), Designation VARCHAR(25), Department VARCHAR(25), Basic DECIMAL(10,2), DA DECIMAL(10,2), Gross_pay DECIMAL(10,2));

```
INSERT INTO Employee (Emp_code, Emp_name, Designation, Department, Basic)
VALUES
    (1, 'John Doe', 'Manager', 'Market', 10000.00),
    (2, 'Jane Smith', 'Salesperson', 'Sales', 4000.00),
    (3, 'Alice Johnson', 'HR Executive', 'HR', 6000.00),
    (4, 'Bob Williams', 'Supervisor', 'Purchase', 9000.00),
    (5, 'Charlie Brown', 'Salesperson', 'Sales', 3500.00);
```

➤ UPDATE Employee SET DA = Basic * 0.75;

➤ SELECT * FROM Employee WHERE Department IN ('Purchase', 'Sales', 'HR');

➤ UPDATE Employee SET Gross_pay = Basic + DA;

➤ SELECT * FROM Employee WHERE Gross_pay < 10000;

OUTPUT

Query : 1

Employee

Emp_code	Emp_name	Designation	Department	Basic	DA	Gross_pay
empty						

Query : 2

Employee

Emp_code	Emp_name	Designation	Department	Basic	DA	Gross_pay
1	John Doe	Manager	Market	10000		
2	Jane Smith	Salesperson	Sales	4000		
3	Alice Johnson	HR Executive	HR	6000		
4	Bob Williams	Supervisor	Purchase	9000		
5	Charlie Brown	Salesperson	Sales	3500		

Query : 3

Employee

Emp_code	Emp_name	Designation	Department	Basic	DA	Gross_pay
1	John Doe	Manager	Market	10000	7500	
2	Jane Smith	Salesperson	Sales	4000	3000	
3	Alice Johnson	HR Executive	HR	6000	4500	
4	Bob Williams	Supervisor	Purchase	9000	6750	
5	Charlie Brown	Salesperson	Sales	3500	2625	

Query : 4

Output

Emp_code	Emp_name	Designation	Department	Basic	DA	Gross_pay
2	Jane Smith	Salesperson	Sales	4000	3000	
3	Alice Johnson	HR Executive	HR	6000	4500	
4	Bob Williams	Supervisor	Purchase	9000	6750	
5	Charlie Brown	Salesperson	Sales	3500	2625	

Query : 5

Employee

Emp_code	Emp_name	Designation	Department	Basic	DA	Gross_pay
1	John Doe	Manager	Market	10000	7500	17500
2	Jane Smith	Salesperson	Sales	4000	3000	7000
3	Alice Johnson	HR Executive	HR	6000	4500	10500
4	Bob Williams	Supervisor	Purchase	9000	6750	15750
5	Charlie Brown	Salesperson	Sales	3500	2625	6125

Query : 6

Output

Emp_code	Emp_name	Designation	Department	Basic	DA	Gross_pay
2	Jane Smith	Salesperson	Sales	4000	3000	7000
5	Charlie Brown	Salesperson	Sales	3500	2625	6125

LAB- 19

3. Create a table Stock, which stores daily sales of items in a shop, with the following fields and insert at least 5 records into the table.

Item_code	Integer	Primary key
Item_name	Varchar (20)	
Manufacturer_Code	Varchar (5)	
Qty	Integer	
Unit_Price	Decimal (10,2)	

a. Display the item names with stock zero.
b. Display the number of items manufactured by the same manufacturer.
c. Display the highest price and lowest quantity in the stock.
d. Increase the unit price of all items by 10%.

SQL CODE

➤ CREATE TABLE Stock (Item_code INT PRIMARY KEY, Item_name VARCHAR(20), Manufacturer_Code VARCHAR(5), Qty INT, Unit_Price DECIMAL(10,2));

➤ INSERT INTO Stock (Item_code, Item_name, Manufacturer_Code, Qty, Unit_Price) VALUES
(1, 'Product A', 'M001', 100, 150.00),
(2, 'Product B', 'M002', 50, 200.00),
(3, 'Product C', 'M001', 80, 120.00),
(4, 'Product D', 'M003', 0, 180.00),
(5, 'Product E', 'M002', 120, 160.00);

➤ SELECT Item_name FROM Stock WHERE Qty = 0;

➤ SELECT Manufacturer_Code, COUNT(*) AS Item_Count FROM Stock GROUP BY Manufacturer_Code;

➤ SELECT MAX(Unit_Price) AS Highest_Price, MIN(Qty) AS Lowest_Quantity FROM Stock;

➤ UPDATE Stock SET Unit_Price = Unit_Price * 1.10;

OUTPUT

Query : 1

Stock

Item_code	Item_name	Manufacturer_Code	Qty	Unit_Price
empty				

Query : 2

Stock

Item_code	Item_name	Manufacturer_Code	Qty	Unit_Price
1	Product A	M001	100	150
2	Product B	M002	50	200
3	Product C	M001	80	120
4	Product D	M003	0	180
5	Product E	M002	120	160

Query : 3

Output

Item_name
Product D

Query : 4

Output

Manufacturer_Code	Item_Count
M001	2
M002	2
M003	1

Query : 5

Output

Highest_Price	Lowest_Quantity
200	0

Query : 6

Stock

Item_code	Item_name	Manufacturer_Code	Qty	Unit_Price
1	Product A	M001	100	165
2	Product B	M002	50	220.00000000000003
3	Product C	M001	80	132
4	Product D	M003	0	198.00000000000003
5	Product E	M002	120	176

LAB- 20

4. Create a table Bank with the following fields and insert at least 5 records into the table. Acc_No Integer Primary key Acc_Name Varchar (20) Branch_Name Varchar (25) Acc_ Type Varchar (10) Amount Decimal (10,2)

a. Display the account details of "Savings Account" in Kodungallur branch.
b. Change the branch name "Trivandrum" to "Thiruvananthapuram".
c. Display the details of customers in Thiruvananthapuram, Ernakulam and Kozhikode.
d. List the details of customers in Thrissur branch having a minimum balance of Rs. 5000.

SQL CODE

➤ CREATE TABLE Bank (Acc_No INTEGER PRIMARY KEY, Acc_Name VARCHAR(20), Branch_Name VARCHAR(25), Acc_Type VARCHAR(10), Amount DECIMAL(10,2));

➤ INSERT INTO Bank (Acc_No, Acc_Name, Branch_Name, Acc_Type, Amount) VALUES (1001, 'John Doe', 'Kochi', 'Savings', 5000.00), (1002, 'Jane Smith', 'Trivandrum', 'Current', 10000.00), (1003, 'Alice Johnson', 'Kozhikode', 'Savings', 3000.00), (1004, 'Bob Williams', 'Kochi', 'Current', 8000.00), (1005, 'Eve Brown', 'Thrissur', 'Savings', 7000.00);

➤ SELECT * FROM Bank WHERE Branch_Name = 'Kodungallur' AND Acc_Type = 'Savings';

➤ UPDATE Bank SET Branch_Name = 'Thiruvananthapuram' WHERE Branch_Name = 'Trivandrum';

➤ SELECT * FROM Bank WHERE Branch_Name IN ('Thiruvananthapuram', 'Ernakulam', 'Kozhikode');

➤ SELECT * FROM Bank WHERE Branch_Name = 'Thrissur' AND Amount >= 5000;

OUTPUT

Query : 1

Bank

Acc_No	Acc_Name	Branch_Name	Acc_Type	Amount
empty				

Query : 2

Bank

Acc_No	Acc_Name	Branch_Name	Acc_Type	Amount
1001	John Doe	Kochi	Savings	5000
1002	Jane Smith	Trivandrum	Current	10000
1003	Alice Johnson	Kozhikode	Savings	3000
1004	Bob Williams	Kodungallur	Savings	8000
1005	Eve Brown	Thrissur	Savings	7000

Query : 3

Output

Acc_No	Acc_Name	Branch_Name	Acc_Type	Amount
1004	Bob Williams	Kodungallur	Savings	8000

Query : 4

Bank

Acc_No	Acc_Name	Branch_Name	Acc_Type	Amount
1001	John Doe	Kochi	Savings	5000
1002	Jane Smith	Thiruvananthapuram	Current	10000
1003	Alice Johnson	Kozhikode	Savings	3000
1004	Bob Williams	Kodungallur	Savings	8000
1005	Eve Brown	Thrissur	Savings	7000

Query : 5

Output

Acc_No	Acc_Name	Branch_Name	Acc_Type	Amount
1002	Jane Smith	Thiruvananthapuram	Current	10000
1003	Alice Johnson	Kozhikode	Savings	3000

Query : 6

Output

Acc_No	Acc_Name	Branch_Name	Acc_Type	Amount
1005	Eve Brown	Thrissur	Savings	7000