using System;

using System.Collections.Generic;

using System.Linq;

using System.Runtime.InteropServices;

using System.Security.Cryptography.X509Certificates;

using System.Text;

using System.Threading.Tasks;

namespace project\_22\_03\_2023\_cli

{

internal class Test\_06\_04\_2023

{

public void Pattern(int n)

{

for(int i = 0; i < n; i++)

{

for(int j = 0; j<=i; j++)

{

Console.Write("\*");

}Console.WriteLine();

}

}

}

public class EB

{

const int pricePerUnit = 7;

public int ebId;

public string name;

public int unit;

HouseDet ebHouse = new HouseDet();

public void GetEBHouse()

{

ebHouse = Program.GetHouseDetails(this.name);

Console.WriteLine(ebHouse.hasElectricityAccess);

}

public EB(int ebId, string name)

{

this.ebId = ebId;

this.name = name;

this.unit = EnterReading();

}

public int EnterReading()

{

int u = 0;

if (ebHouse.hasElectricityAccess)

{

if (ebHouse.hasFan)

{

u += 5;

}

if (ebHouse.hasWashingMachine)

{

u += 7;

}

if (ebHouse.hasAC)

{

u += 20;

}

if (ebHouse.hasTV)

{

u += 15;

}

}

else

{

unit = 0;

}

return u;

}

public int CalculateBill()

{

return EnterReading() \* pricePerUnit;

}

}

public class HouseDet

{

public HouseDet()

{

}

public string name;

public bool hasElectricityAccess;

public bool hasTV;

public bool hasFan;

public bool hasAC;

public bool hasWashingMachine;

public HouseDet(string name, bool hasElectricityAccess, bool hasTV, bool hasFan, bool hasAC, bool hasWashingMachine)

{

this.name = name;

this.hasElectricityAccess = hasElectricityAccess;

this.hasTV = hasTV;

this.hasFan = hasFan;

this.hasAC = hasAC;

this.hasWashingMachine= hasWashingMachine;

}

}

class Players

{

public int Player\_Id;

public string Player\_Name;

public int Team\_Id;

public string Player\_City;

public Players(int Player\_Id, string Player\_name, int Team\_Id, string Player\_City)

{

this.Player\_Id = Player\_Id;

this.Player\_Name = Player\_name;

this.Team\_Id = Team\_Id;

this.Player\_City = Player\_City;

}

}

partial class Calc

{

public void Display()

{

Console.WriteLine("Calculator");

}

}

partial class Calc

{

public int Add(int a, int b)

{

return a + b;

}

}

partial class Calc

{

public int Sub(int a, int b)

{

return a - b;

}

}

partial class Calc

{

public int Mul(int a, int b)

{

return a \* b;

}

}

partial class Calc

{

public int Div(int a, int b)

{

return a / b;

}

}

public class IdGen

{

public static int id;

static IdGen()

{

id++;

Console.WriteLine($"id={id}");

}

public IdGen()

{

if(id > 0)

{

id++;

Console.WriteLine($"id={id}");

}

}

}

}

/\*DRIVERS CODE\*/

Test\_06\_04\_2023 test = new Test\_06\_04\_2023();

Console.WriteLine("Enter a Number");

int n = Int32.Parse(Console.ReadLine());

test.Pattern(n);

ebData.Add(new EB(101, "Person1"));

ebData.Add(new EB(102, "Person2"));

ebData.Add(new EB(103, "Person3"));

ebData.Add(new EB(104, "Person4"));

persons.Add(new HouseDet("Person1", true,true, true, true,true));

persons.Add(new HouseDet("Person2", true, false, true, true, true));

persons.Add(new HouseDet("Person3", false, false, false, false, false));

persons.Add(new HouseDet("Person4", true, true, false, true, false));

bool e\_b = true;

do

{

Console.WriteLine("Enter EB id");

int id = Int32.Parse(Console.ReadLine());

bool da = false;

EB ebPerson = null;

foreach(EB eb in ebData)

{

if(id == eb.ebId)

{

ebPerson = eb;

da = true;

break;

}

else

{

da = false;

}

}

if (da)

{

ebPerson.GetEBHouse();

Console.WriteLine("Hi {0}", ebPerson.name);

Console.WriteLine("Welcome to Electricity Board");

Console.WriteLine("Press 0 to generate Reading");

Console.WriteLine("Press 1 to Calculate Bill");

bool e\_y\_n = true; ;

do

{

int act = Int32.Parse(Console.ReadLine());

if (act == 0)

{

Console.WriteLine("{0} Number of units consumed is {1}", ebPerson.name, ebPerson.EnterReading());

}

else if (act == 1)

{

Console.WriteLine("{0} your Electricity Bill is {1}", ebPerson.name, ebPerson.CalculateBill());

}

else

{

e\_y\_n = false;

}

}

while (e\_y\_n);

}

else

{

Console.WriteLine("No User Data Found!");

}

bool y\_n = true;

while (y\_n)

{

Console.WriteLine("");

Console.WriteLine("Enter Y/N");

string action = Console.ReadLine().ToLower();

Console.WriteLine();

if (action == "y")

{

e\_b = true;

y\_n = false;

}

else if (action == "n")

{

e\_b = false;

y\_n = false;

break;

}

else

{

Console.WriteLine("Enter between Y/N!!!");

Console.WriteLine();

y\_n = true;

}

}

}

while (e\_b);

List<Players> pList = new List<Players>();

pList.Add(new Players(101, "Playe1", 1001, "chennai"));

pList.Add(new Players(102, "Playe2", 1002, "chennai"));

pList.Add(new Players(103, "Playe3", 1003, "chennai"));

pList.Add(new Players(104, "Playe4", 1001, "delhi"));

pList.Add(new Players(105, "Playe5", 1002, "mumbai"));

pList.Add(new Players(106, "Playe6", 1002, "mumbai"));

var count = from p in pList

group p by p.Player\_City into p\_group

select new { city = p\_group.Key, count = p\_group.Count() };

foreach(var v in count)

{

Console.WriteLine(v);

}

Calc c = new Calc();

Console.WriteLine("Enter A");

int a = Int32.Parse(Console.ReadLine());

Console.WriteLine("Enter B");

int b = Int32.Parse(Console.ReadLine());

c.Display();

Console.WriteLine("Addition of a & b "+ c.Add(a, b));

Console.WriteLine("Subtraction of a & b " + c.Sub(a, b));

Console.WriteLine("Multiplication of a & b " + c.Mul(a, b));

Console.WriteLine("Division of a & b " + c.Div(a, b));

IdGen ig1 = new IdGen();

IdGen ig2 = new IdGen();

IdGen ig3 = new IdGen();