**Assignment 5**

**Name : Anjali Hingu**

**PRN : 2020033800098732**

**CODE:**

// See https://aka.ms/new-console-template for more information

using System.Runtime.InteropServices;

using System.Security.Cryptography.X509Certificates;

using static System.Runtime.InteropServices.JavaScript.JSType;

public class Verna

{

public string type { get; set; } = "4 wheeler"; public int seater { get; set; } = 4; public bool renttype { get; set; } = true; public int age { get; set; }

public int number { get; set; }

public static int rentperunit { get; set; } = 600; public int m\_day { get; set; }

public int m\_month { get; set; }

public int m\_year { get; set; }

public DateTime m\_date { get; set; }

public Verna()

{

Console.WriteLine("Verna"); Console.Write("Enter Number of vehicle :"); number = Convert.ToInt32(Console.ReadLine()); Console.Write("Enter Age of Vehicle :"); age = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter Maintance's Day Month Year :"); m\_day = Convert.ToInt32(Console.ReadLine()); m\_month = Convert.ToInt32(Console.ReadLine());

m\_year = Convert.ToInt32(Console.ReadLine());

m\_date = new DateTime(m\_year, m\_month, m\_day); if (DateTime.Compare(DateTime.Today, m\_date) < 0)

{

renttype = false;

}

}

public void Info()

{

Console.WriteLine("--Verna--");

Console.WriteLine("Type of Vehicle :" + type);

Console.WriteLine("Number of vehicle :" + number);

Console.WriteLine("Vehicle's seat :" + seater);

Console.Write("Is Vehicle Avaliable : ");

if (renttype == false)

{

Console.WriteLine("Verna " + number + " is Under Maintainance");

}

else

{

Console.WriteLine("Yes");

}

Console.WriteLine("\nMaintance Date " + m\_date);

}

}

class I10

{

public string type { get; set; } = "4 wheeler"; public int seater { get; set; } = 4; public bool renttype { get; set; } = true; public int number { get; set; }

public int age { get; set; }

public static int rentperunit { get; set; } = 700; public int m\_day { get; set; }

public int m\_month { get; set; }

public int m\_year { get; set; }

public DateTime m\_date { get; set; }

public I10()

{

Console.WriteLine("i10"); Console.Write("Enter number of vehicle :"); number = Convert.ToInt32(Console.ReadLine()); Console.Write("Enter Age of Vehicle :"); age = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter Maintance's Day Month Year :"); m\_day = Convert.ToInt32(Console.ReadLine()); m\_month = Convert.ToInt32(Console.ReadLine());

m\_year = Convert.ToInt32(Console.ReadLine());

m\_date = new DateTime(m\_year, m\_month, m\_day);

if (DateTime.Compare(DateTime.Today, m\_date) < 0)

{

renttype = false;

}

}

public void Info()

{

Console.WriteLine("---i10---"); Console.WriteLine("Type of Vehicle :" + type);

Console.WriteLine("Vehicle's seat :" + seater);

Console.Write("Is Vehicle Avaliable : ");

if (renttype == false)

{

Console.WriteLine("I10 " + number + " is Under Maintainance");

}

else

{

Console.WriteLine("Yes");

}

Console.WriteLine("\nMaintance Date " + m\_date);

}

}

class HondaCity

{

public string type { get; set; } = "4 wheeler"; public int seater { get; set; } = 4; public bool renttype { get; set; } = true; public int number { get; set; }

public int age { get; set; }

public static int rentperunit { get; set; } = 800; public int m\_day { get; set; }

public int m\_month { get; set; }

public int m\_year { get; set; }

public DateTime m\_date { get; set; }

public HondaCity()

{

Console.WriteLine("HondaCity"); Console.Write("Enter Number of vehicle :"); number = Convert.ToInt32(Console.ReadLine()); Console.Write("Enter Age of Vehicle :"); age = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter Maintance's Day Month Year :"); m\_day = Convert.ToInt32(Console.ReadLine()); m\_month = Convert.ToInt32(Console.ReadLine());

m\_year = Convert.ToInt32(Console.ReadLine());

m\_date = new DateTime(m\_year, m\_month, m\_day);

if (DateTime.Compare(DateTime.Today, m\_date) < 0)

{

renttype = false;

}

}

public void Info()

{

Console.WriteLine("---HondaCity---");

Console.WriteLine("Type of Vehicle :" + type);

Console.WriteLine("Vehicle's seat :" + seater); Console.Write("Is Vehicle Avaliable : ");

if (renttype == false)

{

Console.WriteLine("HondaCity " + number + " is Under Maintainance");

}

else

{

Console.WriteLine("Yes");

}

Console.WriteLine("\nMaintance Date " + m\_date);

}

}

class I20

{

public string type { get; set; } = "4 wheeler"; public int seater { get; set; } = 4; public bool renttype { get; set; }

public int number { get; set; }

public int age { get; set; }

public static int rentperunit { get; set; } = 500; public int m\_day { get; set; }

public int m\_month { get; set; }

public int m\_year { get; set; }

public DateTime m\_date { get; set; }

public I20()

{

Console.WriteLine("I20"); Console.Write("Enter Number of vehicle :"); number = Convert.ToInt32(Console.ReadLine()); Console.Write("Enter Age of Vehicle :"); age = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter Maintance's Day Month Year :"); m\_day = Convert.ToInt32(Console.ReadLine()); m\_month = Convert.ToInt32(Console.ReadLine()); m\_year = Convert.ToInt32(Console.ReadLine());

m\_date = new DateTime(m\_year, m\_month, m\_day); if (DateTime.Compare(DateTime.Today, m\_date) < 0)

{

renttype = false;

}

}

public void Info()

{

Console.WriteLine("---I20---");

Console.WriteLine("Type of Vehicle :" + type);

Console.WriteLine("Vehicle's seat :" + seater); Console.Write("Is Vehicle Avaliable : ");

if (renttype == false)

{

Console.WriteLine("I20 " + number + " is Under Maintainance");

}

else

{

Console.WriteLine("Yes");

}

Console.WriteLine("\nMaintance Date " + m\_date);

}

}

class RentVehicle<T>

{

public int s\_day { get; set; }

public int s\_month { get; set; }

public int s\_year { get; set; }

public DateTime s\_date { get; set; }

public int e\_day { get; set; }

public int e\_month { get; set; }

public int e\_year { get; set; }

public DateTime e\_date { get; set; }

public int noofkmstravelled { get; set; }

public int advancepayment { get; set; }

public object obj; public string vehicle;

public int flag, flag2 = 0;

public List<Verna> v\_list = new List<Verna>();

public List<I10> p\_list = new List<I10>();

public List<HondaCity> h\_list = new List<HondaCity>();

public List<I20> b\_list = new List<I20>();

public RentVehicle(object obj, string vehicle)

{ this.obj = obj; this.vehicle = vehicle; }

public void giveForRent()

{

Console.Write("Enter Start day month year for rent: ");

s\_day = Convert.ToInt32(Console.ReadLine());

s\_month = Convert.ToInt32(Console.ReadLine()); s\_year = Convert.ToInt32(Console.ReadLine());

s\_date = new DateTime(s\_year, s\_month, s\_day); flag = showAvailibilityforbookingforgivendate();

if (flag == 0)

{

Console.WriteLine("it's Available for your date");

Console.WriteLine("Enter End day month year for rent: ");

e\_day = Convert.ToInt32(Console.ReadLine());

e\_month = Convert.ToInt32(Console.ReadLine());

e\_year = Convert.ToInt32(Console.ReadLine());

e\_date = new DateTime(e\_year, e\_month, e\_day);

Console.WriteLine("Enter Kiometers : "); noofkmstravelled = Convert.ToInt32(Console.ReadLine());

calculateRent();

Console.WriteLine("Enter Advance Payment : "); advancepayment = Convert.ToInt32(Console.ReadLine());

check();

}

else

{

Console.WriteLine("sorry,this Vehicle is not available so please check it ");

getCheckListRentedandAvailableVehicle();

getCheckListvehiclesmaintanence(); check();

}

}

public void calculateRent()

{

if (vehicle == "v")

{

Console.WriteLine("rent of Vehicle will be : " + (Verna.rentperunit \* noofkmstravelled));

}

else if (vehicle == "p")

{

Console.WriteLine("rent of Vehicle will be : " + (I10.rentperunit \* noofkmstravelled));

}

if (vehicle == "h")

{

Console.WriteLine("rent of Vehicle will be : " + (HondaCity.rentperunit \* noofkmstravelled));

}

if (vehicle == "b")

{

Console.WriteLine("rent of Vehicle will be : " + (I20.rentperunit \* noofkmstravelled));

}

}

public int showAvailibilityforbookingforgivendate()

{

int avail = 1;

if (vehicle == "v")

{

foreach (var rent\_Verna in v\_list)

{

if (rent\_Verna.renttype == true)

{

avail = 0; rent\_Verna.Info(); rent\_Verna.renttype = false;

break;

}

}

}

else if (vehicle == "p")

{

foreach (var rent\_I10 in p\_list)

{

if (rent\_I10.renttype == true)

{

avail = 0; rent\_I10.Info();

rent\_I10.renttype = false;

break;

}

}

}

else if (vehicle == "h")

{

foreach (var rent\_HondaCity in h\_list)

{

if (rent\_HondaCity.renttype == true)

{

avail = 0;

rent\_HondaCity.Info(); rent\_HondaCity.renttype = false;

break;

}

}

}

else

{

foreach (var rent\_I20 in b\_list)

{

if (rent\_I20.renttype == true)

{

avail = 0; rent\_I20.Info();

rent\_I20.renttype = false; break;

}

}

}

return avail;

}

public void getCheckListRentedandAvailableVehicle()

{

Console.WriteLine("Rented And Available Vehicle List");

foreach (var rent\_verna in v\_list)

{

if (rent\_verna.renttype == true)

{

Console.WriteLine("Verna " + rent\_verna.number + " is Availble");

}

else if (DateTime.Compare(s\_date, rent\_verna.m\_date) > 0)

{

Console.WriteLine("Verna " + rent\_verna.number + " is Rented");

}

}

foreach (var rent\_i10 in p\_list)

{

if (rent\_i10.renttype == true)

{

Console.WriteLine("I10 " + rent\_i10.number + " is Availble");

}

else if (DateTime.Compare(s\_date, rent\_i10.m\_date) > 0)

{

Console.WriteLine("I10 " + rent\_i10.number + " is Rented");

}

}

foreach (var rent\_hondacity in h\_list)

{

if (rent\_hondacity.renttype == true)

{

Console.WriteLine("HondaCity " + rent\_hondacity.number + " is Availble");

}

else if (DateTime.Compare(s\_date, rent\_hondacity.m\_date) > 0)

{

Console.WriteLine("HondaCity " + rent\_hondacity.number + " is Rented");

}

}

foreach (var rent\_i20 in b\_list)

{

if (rent\_i20.renttype == true)

{

Console.WriteLine("I20" + rent\_i20.number + " is Availble");

}

else if (DateTime.Compare(s\_date, rent\_i20.m\_date) > 0)

{

Console.WriteLine("I20" + rent\_i20.number + " is Rented");

}

}

}

public void getCheckListvehiclesmaintanence()

{

Console.WriteLine("Under Maintance Vehicle List"); foreach (var main\_verna in v\_list)

{

if (DateTime.Compare(s\_date, main\_verna.m\_date) < 0)

{

Console.WriteLine("Verna " + main\_verna.number + " is Under Maintainance");

}

}

foreach (var main\_i10 in p\_list)

{

if (DateTime.Compare(s\_date, main\_i10.m\_date) < 0)

{

Console.WriteLine("I10 " + main\_i10.number + " is Under Maintainance");

}

}

foreach (var main\_hondacity in h\_list)

{

if (DateTime.Compare(s\_date, main\_hondacity.m\_date) < 0)

{

Console.WriteLine("HondaCity " + main\_hondacity.number + " is Under Maintainance");

}

}

foreach (var main\_i20 in b\_list)

{

if (DateTime.Compare(s\_date, main\_i20.m\_date) < 0)

{

Console.WriteLine("I20 " + main\_i20.number + " is Under Maintainance");

}

}

}

public void check()

{

foreach (var main\_verna in v\_list)

{

if ((DateTime.Compare(DateTime.Today, main\_verna.m\_date) > 0 && main\_verna.renttype == false) && (DateTime.Compare(DateTime.Today, e\_date) > 0 && main\_verna.renttype == false))

{

main\_verna.renttype = true;

}

}

foreach (var main\_i10 in p\_list)

{

if ((DateTime.Compare(DateTime.Today, main\_i10.m\_date) > 0 && main\_i10.renttype == false) && (DateTime.Compare(DateTime.Today, e\_date) > 0 && main\_i10.renttype == false))

{

main\_i10.renttype = true;

}

}

foreach (var main\_hondacity in h\_list)

{

if ((DateTime.Compare(DateTime.Today, main\_hondacity.m\_date) > 0 && main\_hondacity.renttype == false) && (DateTime.Compare(DateTime.Today, e\_date) > 0 && main\_hondacity.renttype == false))

{

main\_hondacity.renttype = true;

}

}

foreach (var main\_i20 in b\_list)

{

if ((DateTime.Compare(DateTime.Today, main\_i20.m\_date) > 0 && main\_i20.renttype == false) && (DateTime.Compare(DateTime.Today, e\_date) > 0 && main\_i20.renttype == false))

{

main\_i20.renttype = true;

}

}

}

}

public class Assignment5

{

public static void Main(string[] args)

{

Verna[] verna = new Verna[2];

I10[] i10 = new I10[2];

HondaCity[] hondacity = new HondaCity[2];

I20[] i20 = new I20[2];

string vehicle; int flag = 1;

RentVehicle<Verna> re = new RentVehicle<Verna>(verna[0], "v"); for (int i = 0; i< 2; i++)

{

verna[i] = new Verna();

re.v\_list.Add(verna[i]);

}

/\* for (int i = 0; i < 1; i++)

{

qualis[i] = new Qualis();

re.q\_list.Add(qualis[i]);

}

for (int i = 0; i < 1; i++)

{

davidharley[i] = new Davidharley();

re.d\_list.Add(davidharley[i]);

}

for (int i = 0; i < 1; i++)

{

mbenz[i] = new MBenzE(); re.m\_list.Add(mbenz[i]);

}\*/

while (flag == 1)

{

Console.WriteLine(" Welcome to rent vehicle");

Console.WriteLine("Which Car you want to take on rent ???" +

"\n v).Verna" +

"\n p).I10" +

"\n h).HondaCity" +

"\n s).I20" +

"\n e).Enter e for exit " + "\n Enter Your Choice : ");

vehicle = Console.ReadLine();

if (vehicle == "v" | vehicle == "p" | vehicle == "h" | vehicle == "s")

{

re.giveForRent();

}

else if (vehicle == "e")

{

Console.WriteLine("Exited Successfully"); flag = 0;

}

else

{

Console.WriteLine("Enter Correct Option");

}

}

}

}

**Output:**





