[Classes]

Imagine a tollbooth at a bridge. Cars passing by the booth are expected to pay a 50 Rs. Toll. Mostly they do, but sometimes a car goes by without paying. The tollbooth keeps track of the number of cars that have gone by, and of the total amount of money collected. Model this tollbooth with a class called tollbooth. The two data items are a type unsigned int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both of these to 0. A member function called payingCar () increments the total car and adds 5 to the cash total. Another function called nopayCar (), increments the car total but adds nothing to the cash total. Finally a member function called display () displays the two totals. Make appropriate member function’s constant. Include a program to test the class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car. Pushing the Esc key should cause the program to print out the total cars and total cash and then exit.

**//Tollbooth.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Main {

class Tollbooth {

private int totalCars;

private double totalCash;

public Tollbooth() {

this.totalCars = 0;

this.totalCash = 0.0;

}

public void payingCar() {

this.totalCars++;

this.totalCash += 5;

}

public void nonpayingCars() {

this.totalCars++;

}

public void display() {

Console.WriteLine("Total Cars: " + this.totalCars);

Console.WriteLine("Total Cash to be collected: " + this.totalCash);

}

}

}

**//Program.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Main {

class Program {

static void Main(string[] args) {

Tollbooth t1 = new Tollbooth();

Console.WriteLine("Press P if paying car found, N for non-paying and D for status: ");

while (true) {

char status = Convert.ToChar(Console.ReadLine());

switch (status) {

case 'P':

case 'p':

t1.payingCar();

break;

case 'N':

case 'n':

t1.nonpayingCars();

break;

case 'D':

case 'd':

t1.display();

break;

default:

Console.WriteLine("Invalid Input!");

break;

}

}

}

}

}