INDEX

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr.no** | | **Title** | **Page no.** |
| 1. | | Abstract | 1 |
| 2. | | Introduction To Micro Project | 2 |
| 3. | | SYSTEM REQUIREMENTS | 3 |
| 4. | | Source code | 4-10 |
| 5. | | Output | 11-14 |
| 6. | Limitations & Future Enhancement | | 15 |
| 7. | Conclusion | | 16 |
| 8. | References | | 17 |

**Abstract**

In this fast growing world vehicles have become a part and parcel of our life. The management of vehicles is also a necessary thing as compared to the user.

All the details about vehicles cannot be stored in paper and this creates a need for computers to store the details. Our project is created in such way that helps the user to do the management of the vehicles

**Introduction To The Project**

Our project helps to calculate how many cars have been rented in the parking lot and how many vehicles and what kind of vehicles are there and what kind of vehicles are there is still in the parking lot and how many vehicles have arrived and the parking will be full. This created in such way that helps the user to do the management of the vehicles.

* **Used header files**

1. **iostream :** iostream stands for standard input-output stream. This header file contains definitions to objects like cin, cout etc.
2. **process.h :** is a header file in C++ that contains functional declarations and macros.
3. **stdlib.h :** is the header of the general purpose standard library of C programming language which includes functions involving memory allocation, process control, conversions and others. It is compatible with C++ and is known as stdlib in C++. The name "stdlib" stands for "standard library".

* **Features Used Of C++**

1. **Classes And Object**

A class in C++ is the building block , that leads to Object-Oriented programming. It is a user-defined data type, which holds its own data members and member functions, which can be accessed and used by creating an instance of that class.

An Object is an instance of a Class.

1. **Inheritance**

The capability of a class to derive properties and characteristics from another class is called Inheritance.

1. **Constructor**

A constructor is a member function of a class which initializes objects of a class.

**SYSTEM REQUIREMENTS**

* **HARDWARE**

|  |  |
| --- | --- |
| PROCESSOR | Intel i3 OR HIGHER |
| RAM | 2 GB OR HIGHER |
| MONITOR | 17 INCH COLOUR MONITOR |
| KEYBOARD | 104 KEYBOARD |
| MOUSE | STANDARD MOUSE |

* **SOFTWARE**

|  |  |
| --- | --- |
| OPERATING SYSTEM | WINDOWS 7 OR HIGHER |
| PROGRAM | TURBO C++ COMPILER /CODE::BLOCKS |

**Source code**

#include<iostream>

#include<process.h>

#include<stdlib.h>

using namespace std;

class Parking

{

protected:

int amount;

int Count;

int two,three,four;

public:

Parking()

{

amount=0;

Count=0;

two=0;

three=0;

four=0;

}

void Delet()

{

amount=0;

Count=0;

two=0;

three=0;

four=0;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"All Record deleted......."<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

};

class TwoWheel:virtual public Parking

{

public:

void t\_wheel()

{

if(Count<=50)

{

two++;

amount=amount+100;

Count=Count+1;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"A new TwoWheeler has parked"<<endl;

cout<<"Total numbers of vehicles "<<Count<<endl;

cout<<"now free space="<<50-(two+three+four)<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

else

{

cout<<"Sorry! parking is full"<<endl;

}

}

};

class ThreeWheel:virtual public Parking

{

public:

void th\_wheel()

{

if(Count<=50)

{

three++;

amount=amount+200;

Count=Count+1;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"A new Three Wheeler has parked"<<endl;

cout<<"Total numbers of vehicles "<<Count<<endl;

cout<<"now free space="<<50-(two+three+four)<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

else

{

cout<<"Sorry! parking is full";

}

}

};

class FourWheel:virtual public Parking

{

public:

void f\_wheel()

{

if(Count<=50)

{

four++;

amount=amount+300;

Count=Count+1;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"A new FourWheeler has parked"<<endl;

cout<<"Total numbers of vehicles "<<Count<<endl;

cout<<"now free space="<<50-(two+three+four)<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

else

{

cout<<"Sorry! parking is full";

}

}

};

class ShowRecord:public TwoWheel,public ThreeWheel,public FourWheel

{

public:

void Show()

{

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"Total amount="<<amount<<endl;

cout<<"Total numbers of vehicles :"<<Count<<endl;

cout<<"Total numbers of Two Wheeler :"<<two<<endl;

cout<<"Total number of Three Wheeler :"<<three<<endl;

cout<<"Total number of Four Wheeler :"<<four<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

} ;

int main()

{

int u\_input;

ShowRecord s;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"VEHICLE PARKING SYSTEM"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"Parking number of vehicles limit=50"<<endl;

cout<<"TwoWheeler parking charges\t=100"<<endl;

cout<<"ThreeWheeler parking charges\t=200"<<endl;

cout<<"fourWheeler parking charges\t=300"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

while(1)

{

//Menu

cout<<"Choose our service........"<<endl;

cout<<"press 1 for Two Wheeler"<<endl;

cout<<"press 2 for Three Wheeler"<<endl;

cout<<"press 3 for Four Wheeler"<<endl;

cout<<"press 4 for show the record"<<endl;

cout<<"press 5 for delete the record"<<endl;

cout<<"press 6 for Exit"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cin>>u\_input;

switch(u\_input)

{

case 1:

system("cls");

s.t\_wheel();

break;

case 2:

system("cls");

s.th\_wheel();

break;

case 3:

system("cls");

s.f\_wheel();

break;

case 4:

system("cls");

s.Show();

break;

case 5:

system("cls");

s.Delet();

break;

case 6:

exit(0);

default:

{

system("cls");

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

cout<<"WRONG INPUT"<<endl;

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"<<endl;

}

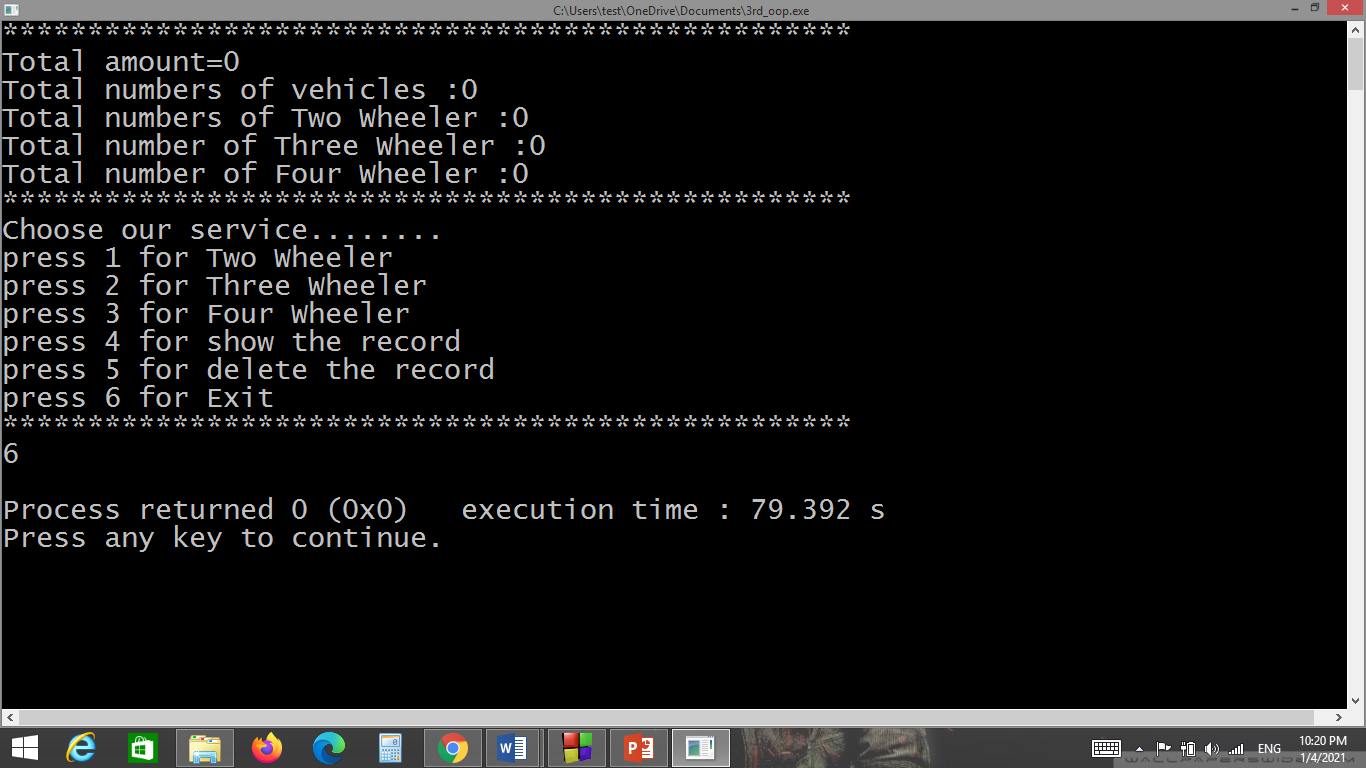
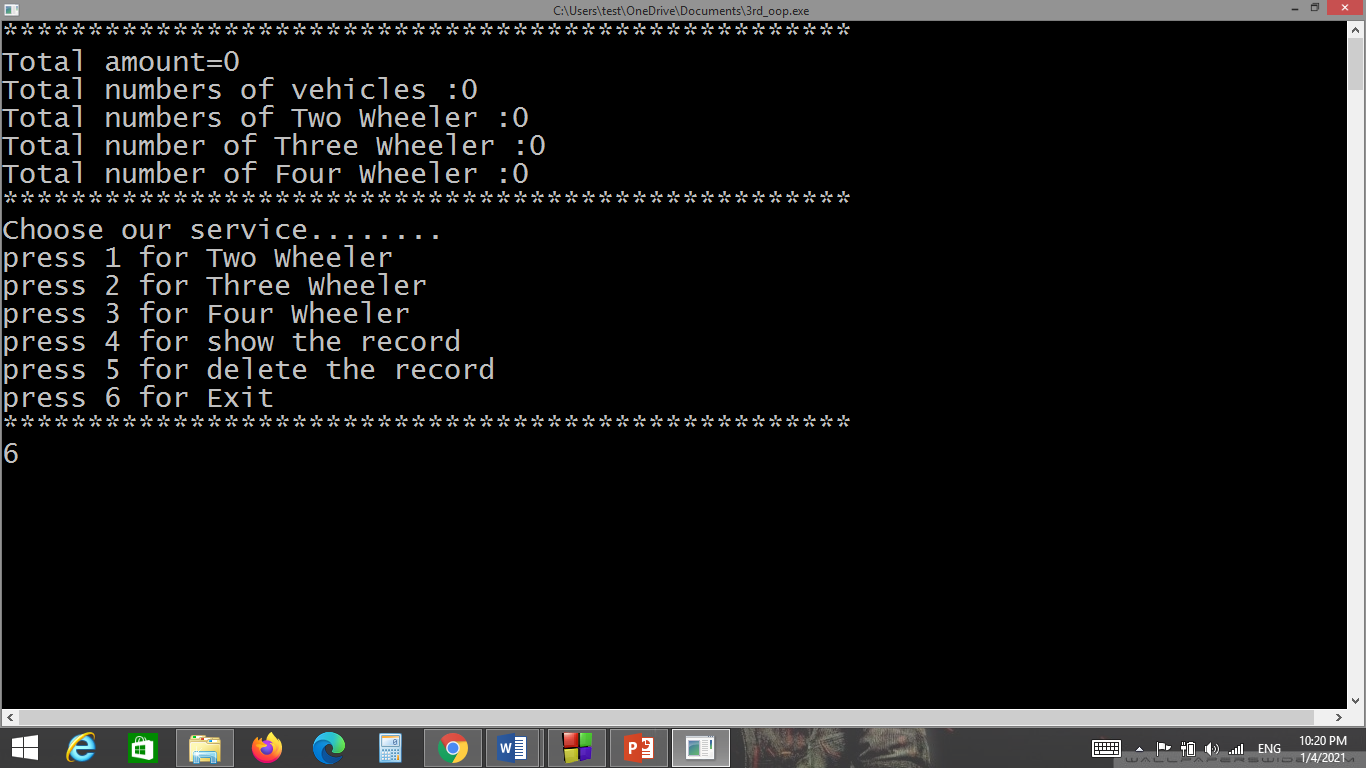
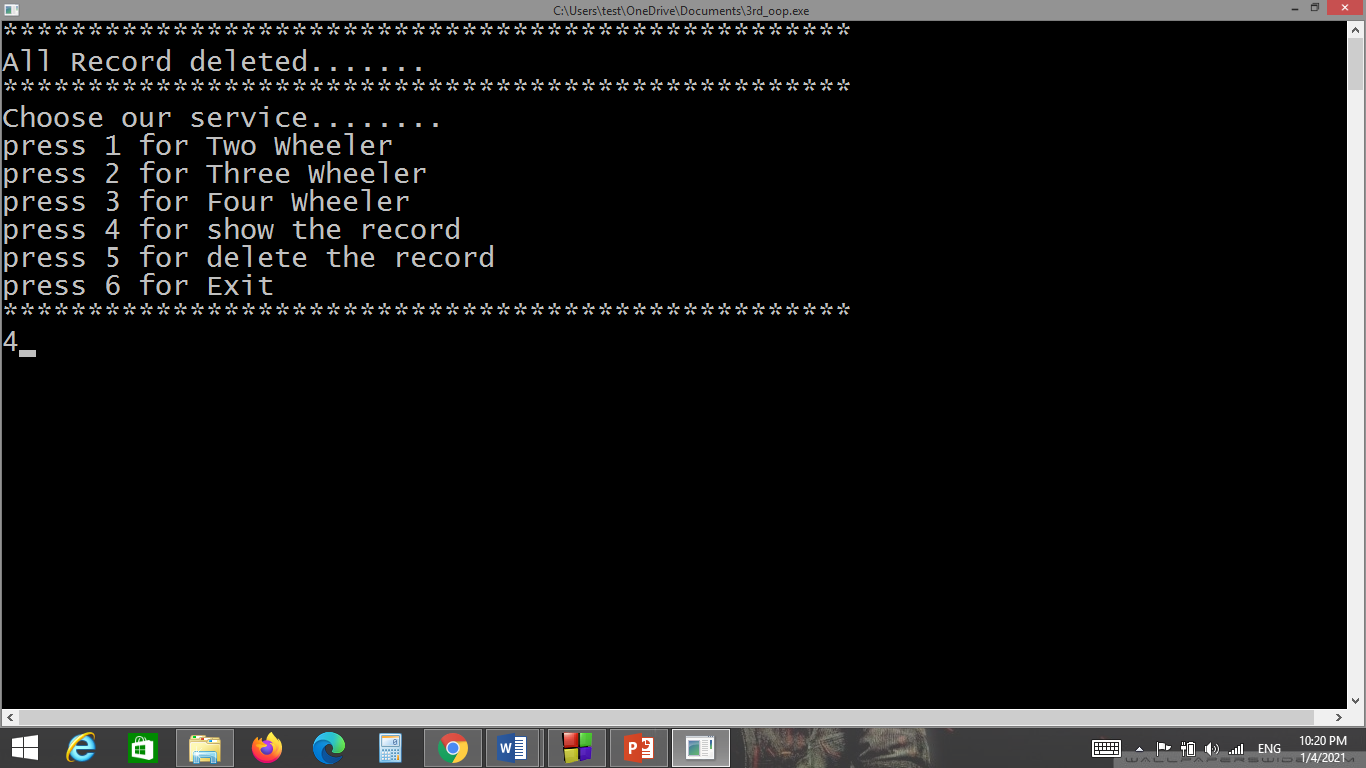
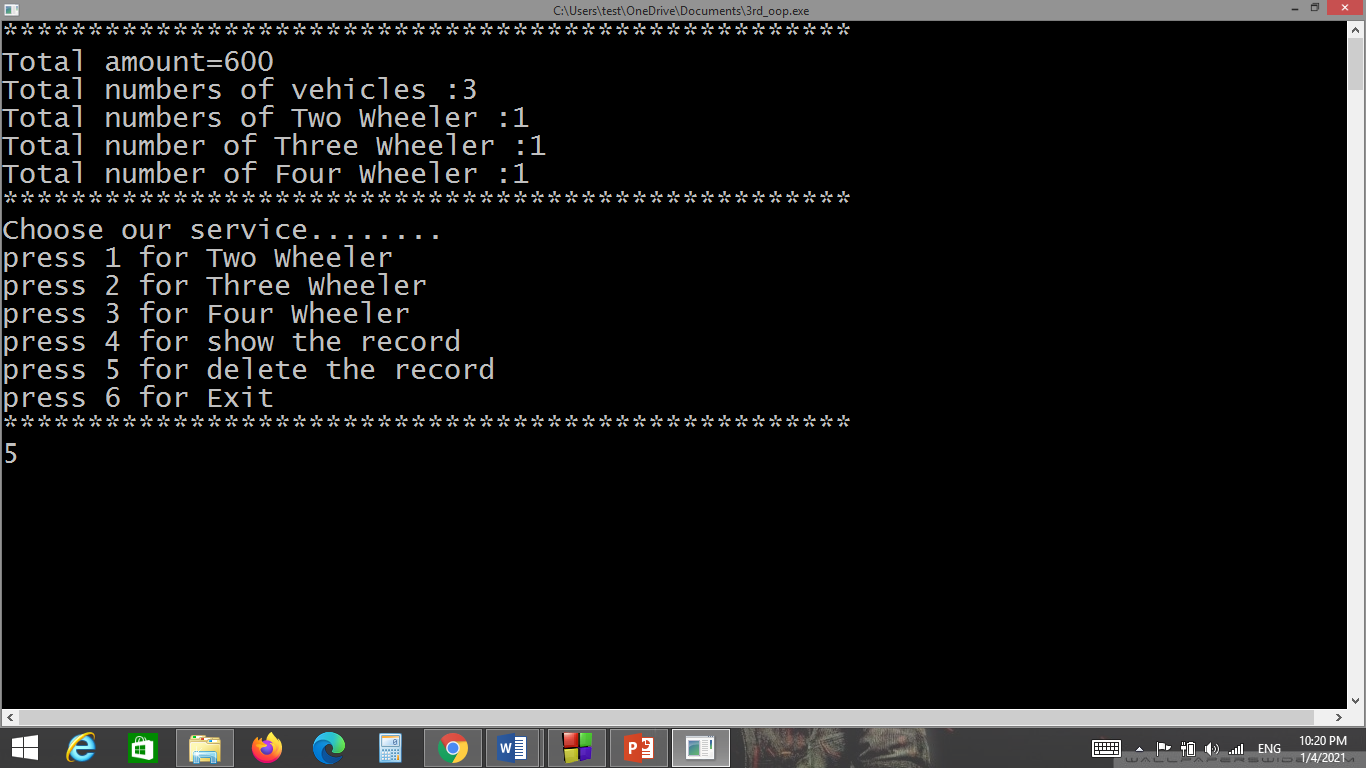
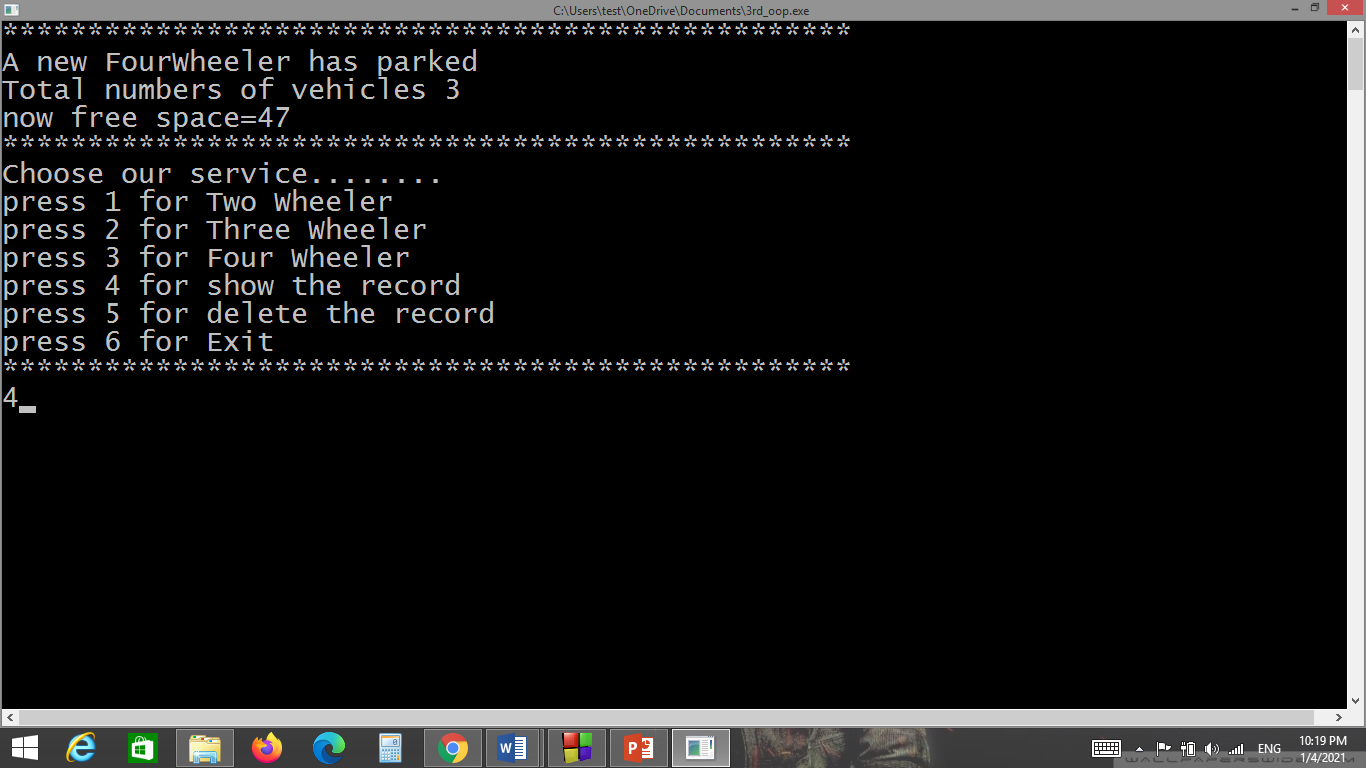
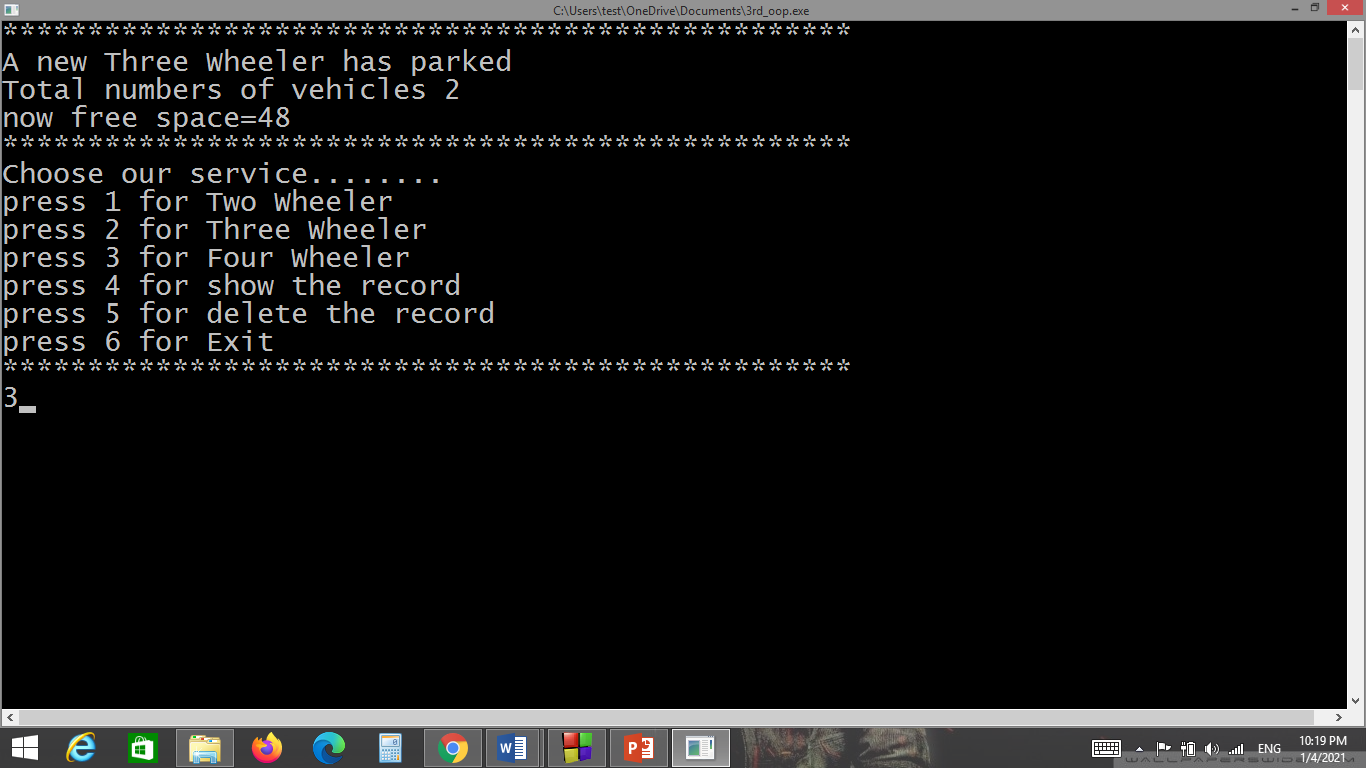
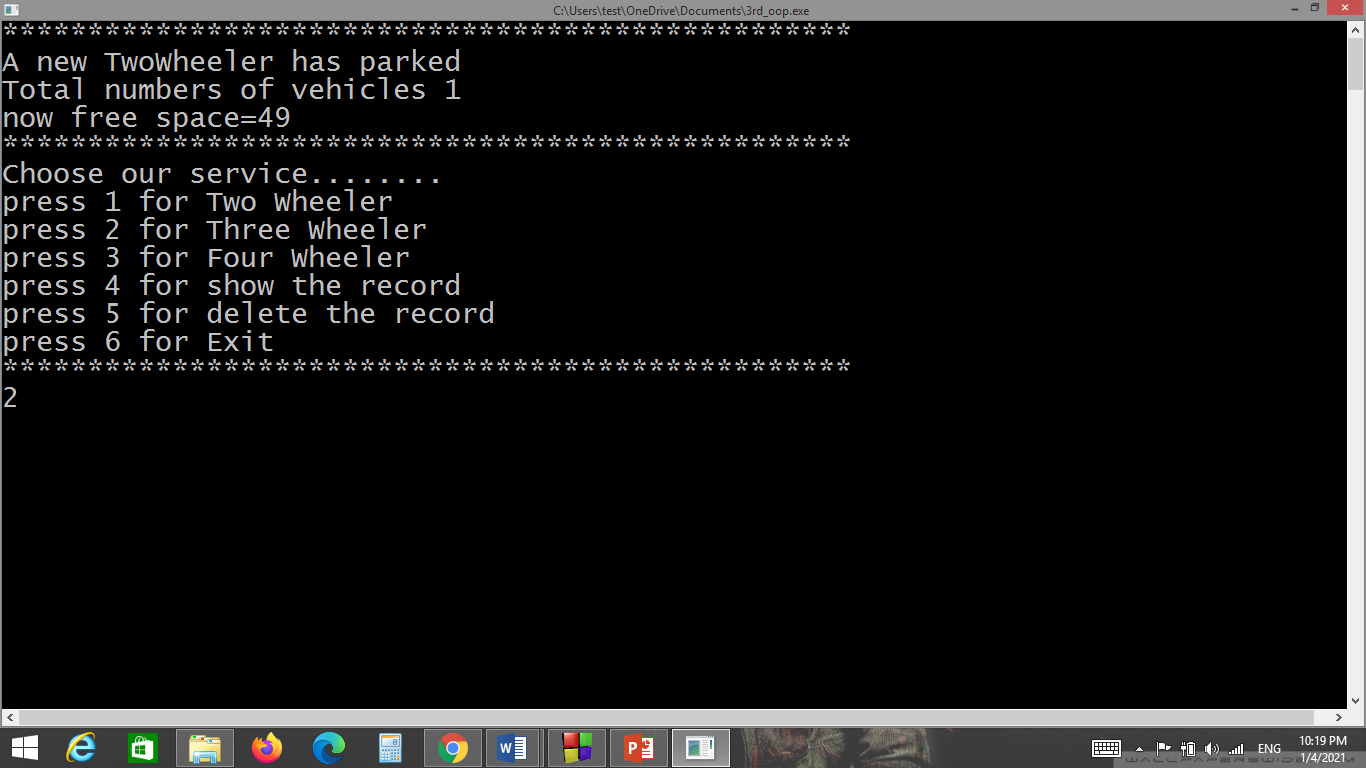
}

}

return(0);

}

**Output**



**Limitations And Future Enhancement**

|  |  |
| --- | --- |
| * **Limitations** | * **Future Enhancement** |
| We cannot delete individual vehicle records in the project | In the future, we will be able to delete individual vehicle records in the project |
| You cannot calculate rent for more then one day | In future, we can calculate more than one day’s rent in a project |
| You cannot add a name to our project | In the future, you can add your name to the project |

**Conclusion**

This project is created in such a way that helps the user to do the management of rented vehicles and parking of vehicles.This project helps in parking managements.We learn from this project how to create class and objects and also a inheritance, Constructor etc.

**References**

* **Geeksforgeeks :**

https://www.geeksforgeeks.org/inheritance-in-c/

<https://www.geeksforgeeks.org/c-classes-and-objects/>

<https://www.geeksforgeeks.org/constructors-c/>

* **Code with harry :**

<https://youtu.be/j8nAHeVKL08>

* **Mysirg :**

<https://youtu.be/Iuo9PpGE04Y>