

# Namespace Cars

## Classes

[Car](#)

[EasyCar](#)

Класс, хранящий информацию о легковом транспорте.

[Engine](#)

Класс, отвечающий за хранение информации о двигателях.

# Class Car

Namespace: [Cars](#)

Assembly: Cars.dll

```
public abstract class Car
```








## Inheritance

[object](#)  ← Car

## Derived

[EasyCar](#)

## Inherited Members

[object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  , [object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#)  , [object.ToString\(\)](#) 

# Constructors

## Car(string, Engine)

Конструктор для объекта класса Car

```
public Car(string carName, Engine carEngine)
```

## Parameters

carName [string](#) 

Переменная для обозначения названия автомобиля

carEngine [Engine](#)

Переменная для обозначения двигателя автомобиля

# Methods

## SetEngine(Engine)

Метод для сохранения информации о двигателе автомобиля.

```
public void SetEngine(Engine carEngine)
```

### Parameters

`carEngine` [Engine](#)

Переменная, обозначающая двигатель автомобиля.

## ShowCarInfo()

Метод для получения информации об автомобиле

```
public void ShowCarInfo()
```

# Class EasyCar

Namespace: [Cars](#)

Assembly: Cars.dll








Класс, хранящий информацию о легковом транспорте.

```
public class EasyCar : Car
```

## Inheritance

[object](#)  ← [Car](#) ← EasyCar

## Inherited Members

[Car.SetEngine\(Engine\)](#), [Car.ShowCarInfo\(\)](#), [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#) 

# Constructors

## EasyCar(string, Engine, string)

Конструктор для объекта класса HardCar

```
public EasyCar(string name, Engine engine, string type)
```

## Parameters

name [string](#) 

Переменная для обозначения названия автомобиля.

engine [Engine](#)

Переменная для обозначения двигателя автомобиля.

type [string](#) 

Переменная для обозначения типа автомобиля.

# Methods

## ShowType()

Метод для отображения информации о типе автомобиля.

```
public void ShowType()
```

# Class Engine

Namespace: [Cars](#)

Assembly: Cars.dll

Класс, отвечающий за хранение информации о двигателях.

```
public class Engine
```

## Inheritance

[object](#)  ← Engine

## Inherited Members

[object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  ,  
[object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#)  , [object.ToString\(\)](#) 


# Constructors

## Engine(string, int, string)


Конструктор для объекта класса Engine

```
public Engine(string nameEngine, int idEngine, string countryMakeEngine)
```

## Parameters

**nameEngine** [string](#) 

Переменная для названия двигателя

**idEngine** [int](#) 

Переменная для номера двигателя

**countryMakeEngine** [string](#) 

Переменная для страны-производителя двигателя

# Methods

## GetEngineInfo()

Метод для получения информации о двигателе

```
public string GetEngineInfo()
```

### Returns

[string](#) 

Информация о двигателе(его название, номер и страну-производителя)