



Инициализация базы данных

Последнее обновление: 31.10.2015



Если нам необходимо, чтобы при первом обращении база данных уже была заполнена некоторыми начальными значениями, то мы можем произвести ее инициализацию.

Инициализация происходит при первом обращении к контексту данных.

Для инициализации мы можем использовать один из классов инициализаторов, которые имеются в библиотеке .NET:

- **CreateDatabaseIfNotExists:** инициализатор, используемый по умолчанию. Он не удаляет автоматически базу данных и данные, а в случае изменения структуры моделей и контекста данных выбрасывает исключение.
- **DropCreateDatabaseIfModelChanges:** данный инициализатор проверяет на соответствие моделям определения таблиц в базе данных. И если модели не соответствуют определению таблиц, то база данных пересоздается.
- **DropCreateDatabaseAlways:** этот инициализатор будет всегда пересоздавать базу данных.

Используем один из инициализаторов. Для этого нам надо переопределить метод **Seed**:

```
class MyContextInitializer : DropCreateDatabaseAlways<MobileContext>
{
    protected override void Seed(MobileContext db)
    {
        Phone p1 = new Phone {Name = "Samsung Galaxy S5", Price = 14000 };
        Phone p2 = new Phone {Name = "Nokia Lumia 630", Price = 8000 };

        db.Phones.Add(p1);
        db.Phones.Add(p2);
        db.SaveChanges();
    }
}

public class Phone
{
    public int Id { get; set; }
    public string Name { get; set; }
    public int Price { get; set; }
}

class MobileContext : DbContext
{
    static MobileContext()
    {
        Database.SetInitializer<MobileContext>(new MyContextInitializer());
    }

    public MobileContext() : base("DefaultConnection")
    { }
}
```

```
public DbSet<Phone> Phones { get; set; }  
}
```

Собственно инициализатор наследуется от одного из выше рассмотренных классов, который типизируется классом контекста: `DropCreateDatabaseAlways<MobileContext>`.

Все действия по инициализации происходят в методе `Seed`, а сама инициализация предполагает простое сохранение данных в бд с помощью контекста данных.

Чтобы инициализатор сработал, надо его вызвать. Один из способов вызова инициализатора предполагает вызов его в статическом конструкторе класса контекста:

```
static MobileContext()  
{  
    Database.SetInitializer<MobileContext>(new MyContextInitializer());  
}
```

[Назад](#) [Содержание](#) [Вперед](#)

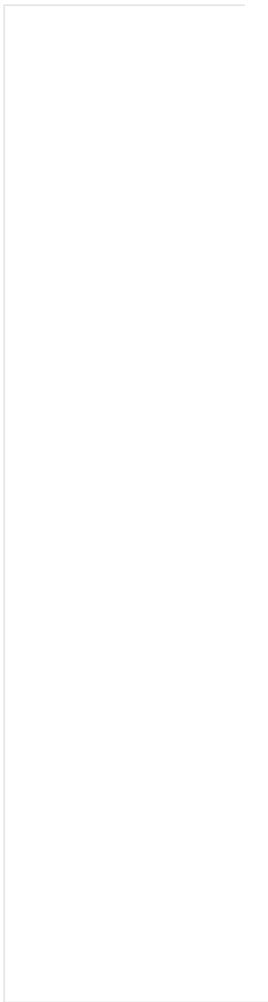


**Стилизатор изображений
Ашманова.** ✓

 ashmanov.net



Яндекс.Директ



[Вконтакте](#) | [Twitter](#) | [Google+](#) | [Канал сайта на youtube](#) | [Помощь сайту](#)

Контакты для связи: metanit22@mail.ru

Copyright © metanit.com, 2012-2017. Все права защищены.