Namespace CarStorageApi.Controllers

Classes

CarController

Interfaces

ICarController

Class CarController

Namespace: CarStorageApi.Controllers

Assembly: CarStorageApi.dll

```
public class CarController : ICarController
```

Inheritance

object ← CarController

Implements

ICarController

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Constructors

CarController(ICarService)

Initializes a new instance of the CarController class.

```
public CarController(ICarService carService)
```

Parameters

`carService` ICarService

The car service.

Methods

GetCars()

Gets the cars.

```
public List<Car> GetCars()
```

Returns

List<Car>

See Also

ICarController

Interface ICarController

Namespace: CarStorageApi.Controllers

Assembly: CarStorageApi.dll

```
public interface ICarController
```

Methods

GetCars()

Gets the cars.

```
List<Car> GetCars()
```

Returns

List<Car>

Namespace CarStorageApi.Data

Classes

CarContext

Class CarContext

Namespace: CarStorageApi.Data

Assembly: CarStorageApi.dll

```
public class CarContext : DbContext, IInfrastructure<IServiceProvider>,
IDbContextDependencies, IDbSetCache, IDbContextPoolable, IResettableService,
IDisposable, IAsyncDisposable
```

Inheritance

object ← DbContext ← CarContext

Implements

IInfrastructure<IServiceProvider>, IDbContextDependencies, IDbSetCache, IDbContextPoolable, IResettableService, IDisposable, IAsyncDisposable

Inherited Members

DbContext.Set<TEntity>() , DbContext.Set<TEntity>(string) ,
DbContext.OnConfiguring(DbContextOptionsBuilder) ,
DbContext.ConfigureConventions(ModelConfigurationBuilder) ,
DbContext.OnModelCreating(ModelBuilder) , DbContext.SaveChanges() ,
DbContext.SaveChanges(bool) , DbContext.SaveChangesAsync(CancellationToken) ,
DbContext.SaveChangesAsync(bool, CancellationToken) , DbContext.Dispose() ,
DbContext.DisposeAsync() , DbContext.Entry<TEntity>(TEntity) , DbContext.Entry(object) ,
DbContext.Add<TEntity>(TEntity) , DbContext.AddAsync<TEntity>(TEntity, CancellationToken) ,
DbContext.Attach<TEntity>(TEntity) , DbContext.Update<TEntity>(TEntity) ,
DbContext.Remove<TEntity>(TEntity) , DbContext.Add(object) ,
DbContext.AddAsync(object, CancellationToken) , DbContext.Attach(object) ,
DbContext.Update(object) , DbContext.Remove(object) , DbContext.AddRange(params object[]) ,
DbContext.AddRangeAsync(params object[]) , DbContext.AttachRange(params object[]) ,
DbContext.UpdateRange(params object[]) , DbContext.RemoveRange(params object[]) ,
DbContext.AddRange(IEnumerable<object>) ,
DbContext.AddRangeAsync(IEnumerable<object>, CancellationToken) ,
DbContext.AttachRange(IEnumerable<object>) , DbContext.UpdateRange(IEnumerable<object>) ,
DbContext.RemoveRange(IEnumerable<object>) , DbContext.Find(Type, params object[]) ,
DbContext.FindAsync(Type, params object[]) ,
DbContext.FindAsync(Type, object[], CancellationToken) , DbContext.Find<TEntity>(params object[]) ,
DbContext.FindAsync<TEntity>(params object[]) ,
DbContext.FindAsync<TEntity>(object[], CancellationToken) ,

DbContext.FromExpression<TResult>(Expression<Func<IQueryable<TResult>>>) , DbContext.Database , DbContext.ChangeTracker , DbContext.Model , DbContext.ContextId , DbContext.SavingChanges , DbContext.SavedChanges , DbContext.SaveChangesFailed , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Constructors

CarContext(DbContextOptions)

```
public CarContext(DbContextOptions options)
```

Parameters

`options` DbContextOptions

Properties

Cars

```
public DbSet<Car> Cars { get; set; }
```

Property Value

DbSet<Car>

Parts

```
public DbSet<Part> Parts { get; set; }
```

Property Value

DbSet<Part>

Namespace CarStorageApi.Migrations

Classes

CreateCarandPartTable

A base class inherited by each EF Core migration.

addedDatetoCarandPart

A base class inherited by each EF Core migration.

addedPartDescription

A base class inherited by each EF Core migration.

Class CreateCarandPartTable

Namespace: CarStorageApi.Migrations

Assembly: CarStorageApi.dll

A base class inherited by each EF Core migration.

```
[DbContext(typeof(CarContext))]
[Migration("20240430094021_CreateCarandPartTable")]
public class CreateCarandPartTable : Migration
```

Inheritance

object ← Migration ← CreateCarandPartTable

Inherited Members

Migration.InitialDatabase , Migration.TargetModel , Migration.UpOperations , Migration.DownOperations , Migration.ActiveProvider , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

See Database migrations for more information and examples.

Methods

BuildTargetModel(ModelBuilder)

Implemented to build the TargetModel.

```
protected override void BuildTargetModel(ModelBuilder modelBuilder)
```

Parameters

`modelBuilder` ModelBuilder

The ModelBuilder to use to build the model.

Remarks

See Database migrations for more information and examples.

Down(MigrationBuilder)

Builds the operations that will migrate the database 'down'.

```
protected override void Down(MigrationBuilder migrationBuilder)
```

Parameters

`migrationBuilder` MigrationBuilder

The MigrationBuilder that will build the operations.

Remarks

That is, builds the operations that will take the database from the state left in by this migration so that it returns to the state that it was in before this migration was applied.

This method must be overridden in each class that inherits from Migration if both 'up' and 'down' migrations are to be supported. If it is not overridden, then calling it will throw and it will not be possible to migrate in the 'down' direction.

See Database migrations for more information and examples.

Up(MigrationBuilder)

Builds the operations that will migrate the database 'up'.

```
protected override void Up(MigrationBuilder migrationBuilder)
```

Parameters

`migrationBuilder` MigrationBuilder

The MigrationBuilder that will build the operations.

Remarks

That is, builds the operations that will take the database from the state left in by the previous migration so that it is up-to-date with regard to this migration.

This method must be overridden in each class that inherits from Migration.

See Database migrations for more information and examples.

Class addedDatetoCarandPart

Namespace: CarStorageApi.Migrations

Assembly: CarStorageApi.dll

A base class inherited by each EF Core migration.

```
[DbContext(typeof(CarContext))]
[Migration("20240501091847_addedDatetoCarandPart")]
public class addedDatetoCarandPart : Migration
```

Inheritance

object ← Migration ← addedDatetoCarandPart

Inherited Members

Migration.InitialDatabase , Migration.TargetModel , Migration.UpOperations , Migration.DownOperations , Migration.ActiveProvider , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

See Database migrations for more information and examples.

Methods

BuildTargetModel(ModelBuilder)

Implemented to build the TargetModel.

```
protected override void BuildTargetModel(ModelBuilder modelBuilder)
```

Parameters

`modelBuilder` ModelBuilder

The ModelBuilder to use to build the model.

Remarks

See Database migrations for more information and examples.

Down(MigrationBuilder)

Builds the operations that will migrate the database 'down'.

```
protected override void Down(MigrationBuilder migrationBuilder)
```

Parameters

`migrationBuilder` MigrationBuilder

The MigrationBuilder that will build the operations.

Remarks

That is, builds the operations that will take the database from the state left in by this migration so that it returns to the state that it was in before this migration was applied.

This method must be overridden in each class that inherits from Migration if both 'up' and 'down' migrations are to be supported. If it is not overridden, then calling it will throw and it will not be possible to migrate in the 'down' direction.

See Database migrations for more information and examples.

Up(MigrationBuilder)

Builds the operations that will migrate the database 'up'.

```
protected override void Up(MigrationBuilder migrationBuilder)
```

Parameters

`migrationBuilder` MigrationBuilder

The MigrationBuilder that will build the operations.

Remarks

That is, builds the operations that will take the database from the state left in by the previous migration so that it is up-to-date with regard to this migration.

This method must be overridden in each class that inherits from Migration.

See Database migrations for more information and examples.

Class addedPartDescription

Namespace: CarStorageApi.Migrations

Assembly: CarStorageApi.dll

A base class inherited by each EF Core migration.

```
[DbContext(typeof(CarContext))]
[Migration("20240430113507_addedPartDescription")]
public class addedPartDescription : Migration
```

Inheritance

object ← Migration ← addedPartDescription

Inherited Members

Migration.InitialDatabase , Migration.TargetModel , Migration.UpOperations , Migration.DownOperations , Migration.ActiveProvider , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

See Database migrations for more information and examples.

Methods

BuildTargetModel(ModelBuilder)

Implemented to build the TargetModel.

```
protected override void BuildTargetModel(ModelBuilder modelBuilder)
```

Parameters

`modelBuilder` ModelBuilder

The ModelBuilder to use to build the model.

Remarks

See Database migrations for more information and examples.

Down(MigrationBuilder)

Builds the operations that will migrate the database 'down'.

```
protected override void Down(MigrationBuilder migrationBuilder)
```

Parameters

`migrationBuilder` MigrationBuilder

The MigrationBuilder that will build the operations.

Remarks

That is, builds the operations that will take the database from the state left in by this migration so that it returns to the state that it was in before this migration was applied.

This method must be overridden in each class that inherits from Migration if both 'up' and 'down' migrations are to be supported. If it is not overridden, then calling it will throw and it will not be possible to migrate in the 'down' direction.

See Database migrations for more information and examples.

Up(MigrationBuilder)

Builds the operations that will migrate the database 'up'.

```
protected override void Up(MigrationBuilder migrationBuilder)
```

Parameters

`migrationBuilder` MigrationBuilder

The MigrationBuilder that will build the operations.

Remarks

That is, builds the operations that will take the database from the state left in by the previous migration so that it is up-to-date with regard to this migration.

This method must be overridden in each class that inherits from Migration.

See Database migrations for more information and examples.

Namespace CarStorageApi.Models

Classes

Car

Part

Class Car

Namespace: CarStorageApi.Models

Assembly: CarStorageApi.dll

```
public class Car
```

Inheritance

object ← Car

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Properties

Color

Gets or sets the color.

```
public string Color { get; set; }
```

Property Value

string

The color.

CreatedDate

Gets or sets the created date.

```
public DateTime? CreatedDate { get; set; }
```

Property Value

DateTime?

The created date.

Id

Gets or sets the identifier.

```
public Guid Id { get; set; }
```

Property Value

Guid

The identifier.

Name

Gets or sets the name.

```
public string Name { get; set; }
```

Property Value

string

The name.

Part

Gets or sets the part.

```
public virtual List<Part> Part { get; set; }
```

Property Value

List<Part>

The part.

SoldDate

Gets or sets the sold date.

```
public DateTime? SoldDate { get; set; }
```

Property Value

DateTime?

The sold date.

Type

Gets or sets the type.

```
public string Type { get; set; }
```

Property Value

string

The type.

Class Part

Namespace: CarStorageApi.Models

Assembly: CarStorageApi.dll

```
public class Part
```

Inheritance

object ← Part

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Properties

CarId

Gets or sets the car identifier.

```
public Guid CarId { get; set; }
```

Property Value

Guid

The car identifier.

CreatedDate

Gets or sets the created date.

```
public DateTime? CreatedDate { get; set; }
```

Property Value

DateTime?

The created date.

Description

Gets or sets the description.

```
public string Description { get; set; }
```

Property Value

string

The description.

Id

Gets or sets the identifier.

```
public Guid Id { get; set; }
```

Property Value

Guid

The identifier.

Name

Gets or sets the name.

```
public string Name { get; set; }
```

Property Value

string

The name.

Namespace CarStorageApi.Repositories

Classes

CarRepository

Interfaces

ICarRepository

Class CarRepository

Namespace: CarStorageApi.Repositories

Assembly: CarStorageApi.dll

```
public class CarRepository : ICarRepository
```

Inheritance

object ← CarRepository

Implements

ICarRepository

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Constructors

CarRepository(CarContext)

Initializes a new instance of the CarRepository class.

```
public CarRepository(CarContext context)
```

Parameters

`context` CarContext

The context.

Methods

GetCars()

Gets the cars.

```
public List<Car> GetCars()
```

Returns

List<Car>

See Also

ICarRepository

Interface ICarRepository

Namespace: CarStorageApi.Repositories

Assembly: CarStorageApi.dll

```
public interface ICarRepository
```

Methods

GetCars()

Gets the cars.

```
List<Car> GetCars()
```

Returns

List<Car>

Namespace CarStorageApi.Services

Classes

CarService

RabbitMQService

Interfaces

ICarService

Class CarService

Namespace: CarStorageApi.Services

Assembly: CarStorageApi.dll

```
public class CarService : ICarService
```

Inheritance

object ← CarService

Implements

ICarService

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Constructors

CarService(ICarRepository)

Initializes a new instance of the CarService class.

```
public CarService(ICarRepository carInventoryRepo)
```

Parameters

`carInventoryRepo` ICarRepository

The car inventory repo.

Methods

GetCars()

Gets the cars.

```
public List<Car> GetCars()
```

Returns

List<Car>

See Also

ICarService

Interface ICarService

Namespace: CarStorageApi.Services

Assembly: CarStorageApi.dll

```
public interface ICarService
```

Methods

GetCars()

Gets the cars.

```
List<Car> GetCars()
```

Returns

List<Car>

Class RabbitMQService

Namespace: CarStorageApi.Services

Assembly: CarStorageApi.dll

```
public class RabbitMQService
```

Inheritance

object ← RabbitMQService

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Constructors

RabbitMQService(IConfiguration)

Initializes a new instance of the RabbitMQService class.

```
public RabbitMQService(IConfiguration configuration)
```

Parameters

`configuration` IConfiguration

The configuration.

Methods

Publish(List<Car>)

Publishes the specified cars.

```
public void Publish(List<Car> cars)
```

Parameters

`cars` List<Car>

The cars.