

Namespace Logistics_Supplier1_API.DTOs.Authentication

Classes

[LoginRequest](#)

Represents a request to log in a user.

[RegisterRequest](#)

Represents a request to register a new user.

Class LoginRequest

Namespace: [Logistics_Supplier1_API.DTOs.Authentication](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to log in a user.

```
public class LoginRequest
```

Inheritance

[object](#) ← LoginRequest

Inherited Members

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

Properties

Password

Gets or sets the password of the user.

```
[Required]  
public string Password { get; set; }
```

Property Value

[string](#)

Username

Gets or sets the username of the user.

```
[Required]  
public string Username { get; set; }
```

Property Value

[string](#) ↗

Class RegisterRequest

Namespace: [Logistics_Supplier1_API.DTOs.Authentication](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to register a new user.

```
public class RegisterRequest
```

Inheritance

[object](#) ← RegisterRequest

Inherited Members

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

Properties

Address

Gets or sets the address of the user.

```
[Required]  
[StringLength(100, MinimumLength = 5)]  
public string Address { get; set; }
```

Property Value

[string](#)

Password

Gets or sets the password of the user.

```
[Required]  
[StringLength(50, MinimumLength = 8)]  
public string Password { get; set; }
```

Property Value

[string](#) ↗

Username

Gets or sets the username of the user.

```
[Required]  
[StringLength(50, MinimumLength = 5)]  
public string Username { get; set; }
```

Property Value

[string](#) ↗

inviteCode

Gets or sets the invite code for registration.

```
[Required]  
[StringLength(64, MinimumLength = 8)]  
public string inviteCode { get; set; }
```

Property Value

[string](#) ↗

Namespace Logistics_Supplier1_API.DTOs.Order Classes

[CreateOrderRequest](#)

Represents a request to create a new order.

[DeleteOrderRequest](#)

Represents a request to delete an order.

[GetOrderByIdRequest](#)

Represents a request to get an order by its ID.

[OrderItemDto](#)

Represents an item within an order in a create order request.

[UpdateOrderRequest](#)

Represents a request to update an order.

Class CreateOrderRequest

Namespace: [Logistics_Supplier1_API.DTOs.Order](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to create a new order.

```
public class CreateOrderRequest
```

Inheritance

[object](#) ← CreateOrderRequest

Inherited Members

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

Properties

OrderItems

Gets or sets the list of order items in the new order.

```
[Required]  
public List<OrderItemDto> OrderItems { get; set; }
```

Property Value

[List](#) <[OrderItemDto](#)>

Class DeleteOrderRequest

Namespace: [Logistics_Supplier1_API.DTOs.Order](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to delete an order.

```
public class DeleteOrderRequest
```

Inheritance

[object](#) ← DeleteOrderRequest

Inherited Members

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

Properties

OrderId

Gets or sets the ID of the order to delete.

```
[Required]  
public int OrderId { get; set; }
```

Property Value

[int](#)

Class GetOrderByIdRequest

Namespace: [Logistics_Supplier1_API.DTOs.Order](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to get an order by its ID.

```
public class GetOrderByIdRequest
```

Inheritance

[object](#) ← GetOrderByIdRequest

Inherited Members

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

Properties

OrderId

Gets or sets the ID of the order.

```
public int OrderId { get; set; }
```

Property Value

[int](#)

Class OrderItemDto

Namespace: [Logistics_Supplier1_API.DTOs.Order](#)

Assembly: Logistics-Supplier1-API.dll

Represents an item within an order in a create order request.

```
public class OrderItemDto
```

Inheritance

[object](#) ← OrderItemDto

Inherited Members

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

Properties

Quantity

Gets or sets the quantity of the product.

```
[Required]  
[Range(1, 2147483647)]  
public int Quantity { get; set; }
```

Property Value

[int](#)

Sku

Gets or sets the SKU (Stock Keeping Unit) of the product.

```
[Required]  
[MaxLength(100)]  
public string Sku { get; set; }
```

Property Value

[string](#) ↗

UnitPrice

Gets or sets the unit price of the product.

```
[Required]  
[Range(0.01, 1.7976931348623157E+308)]  
public decimal UnitPrice { get; set; }
```

Property Value

[decimal](#) ↗

Class UpdateOrderRequest

Namespace: [Logistics_Supplier1_API.DTOs.Order](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to update an order.

```
public class UpdateOrderRequest
```

Inheritance

[object](#) ← UpdateOrderRequest

Inherited Members

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

Properties

OrderId

Gets or sets the ID of the order to update.

```
[Required]  
public int OrderId { get; set; }
```

Property Value

[int](#)

OrderItems

Gets or sets the list of order items in the updated order.

```
[Required]  
public List<OrderItem> OrderItems { get; }
```

Property Value

[List ↗ <OrderItem>](#)

Namespace Logistics_Supplier1_API.DTOs. Product

Classes

[CreateProductRequest](#)

Represents a request to create a new product.

[DeleteProductRequest](#)

Represents a request to delete a product.

[GetProductBySkuRequest](#)

Represents a request to get a product by its SKU.

[UpdateProductRequest](#)

Represents a request to update a product.

Class CreateProductRequest

Namespace: [Logistics_Supplier1_API.DTOs.Product](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to create a new product.

```
public class CreateProductRequest
```

Inheritance

[object](#) ← CreateProductRequest

Inherited Members

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

Properties

Name

Gets or sets the name of the product.

```
[Required]  
[MaxLength(50)]  
public string Name { get; set; }
```

Property Value

[string](#)

Price

Gets or sets the price of the product.

```
[Required]  
[Range(0.01, 1.7976931348623157E+308)]  
public decimal Price { get; set; }
```

Property Value

[decimal ↗](#)

QuantityInStock

Gets or sets the initial stock quantity of the product.

```
[Required]  
[Range(0, 2147483647)]  
public int QuantityInStock { get; set; }
```

Property Value

[int ↗](#)

Sku

Gets or sets the SKU (Stock Keeping Unit) of the product.

```
[Required]  
[MaxLength(100)]  
public string Sku { get; set; }
```

Property Value

[string ↗](#)

Class DeleteProductRequest

Namespace: [Logistics_Supplier1_API.DTOs.Product](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to delete a product.

```
public class DeleteProductRequest
```

Inheritance

[object](#) ← DeleteProductRequest

Inherited Members

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

Properties

Sku

Gets or sets the SKU of the product to delete.

```
[Required]  
[MaxLength(100)]  
public string Sku { get; set; }
```

Property Value

[string](#)

Class GetProductBySkuRequest

Namespace: [Logistics_Supplier1_API.DTOs.Product](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to get a product by its SKU.

```
public class GetProductBySkuRequest
```

Inheritance

[object](#) ← GetProductBySkuRequest

Inherited Members

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

Properties

Sku

Gets or sets the SKU of the product to retrieve.

```
[Required]  
[MaxLength(100)]  
public string Sku { get; set; }
```

Property Value

[string](#)

Class UpdateProductRequest

Namespace: [Logistics_Supplier1_API.DTOs.Product](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to update a product.

```
public class UpdateProductRequest
```

Inheritance

[object](#) ← UpdateProductRequest

Inherited Members

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

Properties

Name

Gets or sets the new name of the product.

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

Property Value

[string](#)

Price

Gets or sets the new price of the product.

```
[Range(0.01, 1.7976931348623157E+308)]  
public decimal Price { get; set; }
```

Property Value

[decimal](#) ↗

Sku

Gets or sets the SKU of the product to update.

```
[Required]  
[MaxLength(100)]  
public string Sku { get; set; }
```

Property Value

[string](#) ↗

StockQuantity

Gets or sets the new stock quantity of the product.

```
[Range(0, 2147483647)]  
public int StockQuantity { get; set; }
```

Property Value

[int](#) ↗

Namespace Logistics_Supplier1_API.DTOs.User Classes

[CreateUserRequest](#)

Represents a request to create a new user.

[GetAllUsersRequest](#)

Represents a request to get all users.

[GetUserRequest](#)

Represents a request to get a specific user.

[UpdatePasswordRequest](#)

Represents a request to update a user's password.

[UpdateUserRequest](#)

Represents a request to update a user's information.

Class CreateUserRequest

Namespace: [Logistics_Supplier1_API.DTOs.User](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to create a new user.

```
public class CreateUserRequest
```

Inheritance

[object](#) ← CreateUserRequest

Inherited Members

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

Properties

Address

Gets or sets the address for the new user.

```
[Required]  
public string Address { get; set; }
```

Property Value

[string](#)

Password

Gets or sets the password for the new user.

```
[MinLength(8)]  
[Required]  
public string Password { get; set; }
```

Property Value

[string](#) ↗

Username

Gets or sets the username for the new user.

```
[Required]  
[MinLength(3)]  
public string Username { get; set; }
```

Property Value

[string](#) ↗

inviteCode

Gets or sets the invite code for registration.

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

Property Value

[string](#) ↗

Class GetAllUsersRequest

Namespace: [Logistics_Supplier1_API.DTOs.User](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to get all users.

```
public class GetAllUsersRequest
```

Inheritance

[object](#) ← GetAllUsersRequest

Inherited Members

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

Properties

Address

Gets or sets the address of the user.

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

Property Value

[string](#)

Id

Gets or sets the ID of the user.

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

Property Value

[int ↗](#)

Role

Gets or sets the role of the user.

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

Property Value

[string ↗](#)

Username

Gets or sets the username of the user.

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

Property Value

[string ↗](#)

Class GetUserRequest

Namespace: [Logistics_Supplier1_API.DTOs.User](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to get a specific user.

```
public class GetUserRequest
```

Inheritance

[object](#) ← GetUserRequest

Inherited Members

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

Properties

Username

Gets or sets the username of the user to retrieve.

```
[Required]  
public string Username { get; set; }
```

Property Value

[string](#)

Class UpdatePasswordRequest

Namespace: [Logistics_Supplier1_API.DTOs.User](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to update a user's password.

```
public class UpdatePasswordRequest
```

Inheritance

[object](#) ← UpdatePasswordRequest

Inherited Members

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

Properties

CurrentPassword

Gets or sets the user's current password.

```
[Required]  
[MinLength(8)]  
public string CurrentPassword { get; set; }
```

Property Value

[string](#)

NewPassword

Gets or sets the user's new password.

```
[Required]  
[MinLength(8)]  
public string NewPassword { get; set; }
```

Property Value

[string](#) ↗

Class UpdateUserRequest

Namespace: [Logistics_Supplier1_API.DTOs.User](#)

Assembly: Logistics-Supplier1-API.dll

Represents a request to update a user's information.

```
public class UpdateUserRequest
```

Inheritance

[object](#) ← UpdateUserRequest

Inherited Members

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

Properties

Address

Gets or sets the new address for the user (optional).

```
public string? Address { get; set; }
```

Property Value

[string](#)

Role

Gets or sets the new role for the user (optional).

```
public string? Role { get; set; }
```

Property Value

[string](#) ↗

Namespace Logistics_Supplier1_API.Data

Classes

[MyDbContext](#)

Represents the database context for the application.

Class MyDbContext

Namespace: [Logistics_Supplier1_API.Data](#)

Assembly: Logistics-Supplier1-API.dll

Represents the database context for the application.

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

Inheritance

[object](#) ← [DbContext](#) ← MyDbContext

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.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

MyDbContext(DbContextOptions<MyDbContext>)

Initializes a new instance of the [MyDbContext](#) class.

```
public MyDbContext(DbContextOptions<MyDbContext> options)
```

Parameters

options [DbContextOptions](#)<[MyDbContext](#)>

The options for configuring the database context.

Properties

InviteCodes

Gets or sets the DbSet for InviteCode entities.

```
public DbSet<InviteCode> InviteCodes { get; set; }
```

Property Value

[DbSet](#)<[InviteCode](#)>

OrderItems

Gets or sets the DbSet for OrderItem entities.

```
public DbSet<OrderItem> OrderItems { get; set; }
```

Property Value

[DbSet](#) <OrderItem>

Orders

Gets or sets the DbSet for Order entities.

```
public DbSet<Order> Orders { get; set; }
```

Property Value

[DbSet](#) <Order>

Products

Gets or sets the DbSet for Product entities.

```
public DbSet<Product> Products { get; set; }
```

Property Value

[DbSet](#) <Product>

Users

Gets or sets the DbSet for User entities.

```
public DbSet<User> Users { get; set; }
```

Property Value

[DbSet](#) <User>

Methods

OnModelCreating(ModelBuilder)

Configures the model for the database context.

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

Parameters

`modelBuilder` [ModelBuilder](#)

The model builder.

Namespace Logistics_Supplier1_API.Handlers

Classes

[AuthenticationHandler](#)

Handles authentication-related API requests.

[InviteCodeHandler](#)

Handles API requests related to invite codes.

[OrderHandlers](#)

Handles API requests related to orders.

[ProductHandlers](#)

Handles API requests related to products.

[RegistrationHandler](#)

Handles user registration requests.

[UserHandlers](#)

Handles API requests related to users.

Class AuthenticationHandler

Namespace: [Logistics_Supplier1_API.Handlers](#)

Assembly: Logistics-Supplier1-API.dll

Handles authentication-related API requests.

```
public static class AuthenticationHandler
```

Inheritance

[object](#) ← AuthenticationHandler

Inherited Members

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

Methods

Authenticate(IUserRepository, LoginRequest)

Authenticates a user and returns a JWT token.

```
public static Task<IResult> Authenticate(IUserRepository userRepository, LoginRequest  
loginRequest = null)
```

Parameters

userRepository [IUserRepository](#)

The repository for accessing user data.

loginRequest [LoginRequest](#)

The login request containing username and password.

Returns

[Task](#)<[IResult](#)>

A JWT token if authentication is successful.

Class InviteCodeHandler

Namespace: [Logistics_Supplier1_API.Handlers](#)

Assembly: Logistics-Supplier1-API.dll

Handles API requests related to invite codes.

```
public class InviteCodeHandler
```

Inheritance

[object](#) ← InviteCodeHandler

Inherited Members

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

Methods

GenerateInviteCode(IInviteCodeRepository)

Generates a new invite code.

```
[Authorize(Roles = "admin")]
public static Task<IResult> GenerateInviteCode(IInviteCodeRepository inviteCodeRepository)
```

Parameters

inviteCodeRepository [IInviteCodeRepository](#)

The repository for accessing invite code data.

Returns

[Task](#)<[IResult](#)>

The generated invite code.

GetAllInviteCodes(IInviteCodeRepository)

Gets all invite codes.

```
[Authorize(Roles = "admin")]
public static Task<IResult> GetAllInviteCodes(IInviteCodeRepository inviteCodeRepository)
```

Parameters

`inviteCodeRepository` [IInviteCodeRepository](#)

The repository for accessing invite code data.

Returns

[Task](#) <[IResult](#)>

A list of all invite codes.

Class OrderHandlers

Namespace: [Logistics_Supplier1_API.Handlers](#)

Assembly: Logistics-Supplier1-API.dll

Handles API requests related to orders.

```
public class OrderHandlers
```

Inheritance

[object](#) ← OrderHandlers

Inherited Members

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

Methods

CreateOrder(IOrderRepository, IProductRepository, HttpContext,
CreateOrderRequest)

Creates a new order.

```
[Authorize(Roles = "user,admin", AuthenticationSchemes = "Bearer")]
public static Task<IResult> CreateOrder(IOrderRepository orderRepository, IProductRepository
productRepository, HttpContext httpContext, CreateOrderRequest req = null)
```

Parameters

orderRepository [IOrderRepository](#)

The repository for accessing order data.

productRepository [IProductRepository](#)

The repository for accessing product data.

httpContext [HttpContext](#)

The HTTP context of the request.

req [CreateOrderRequest](#)

The create order request.

Returns

[Task](#) <[IResult](#)>

The created order.

DeleteOrderAsync(int, IOrderRepository, HttpContext)

Deletes an order.

```
[Authorize(Roles = "user,admin", AuthenticationSchemes = "Bearer")]
public static Task<IResult> DeleteOrderAsync(int orderId, IOrderRepository orderRepository,
HttpContext httpContext)
```

Parameters

orderId [int](#)

The ID of the order to delete.

orderRepository [IOrderRepository](#)

The repository for accessing order data.

httpContext [HttpContext](#)

The HTTP context of the request.

Returns

[Task](#) <[IResult](#)>

A result indicating whether the order was deleted successfully.

GetAllOrders(IOrderRepository, HttpContext)

Gets all orders.

```
[Authorize]
public static Task<IResult> GetAllOrders(IOrderRepository orderRepository,
HttpContext httpContext)
```

Parameters

`orderRepository` [IOrderRepository](#)

The repository for accessing order data.

`httpContext` [HttpContext](#)

The HTTP context of the request.

Returns

[Task](#) <[IResult](#)>

A list of all orders.

GetOrderById(int, IOrderRepository, HttpContext)

Gets an order by its ID.

```
[Authorize(Roles = "user,admin", AuthenticationSchemes = "Bearer")]
public static Task<IResult> GetOrderById(int orderId, IOrderRepository orderRepository,
HttpContext httpContext)
```

Parameters

`orderId` [int](#)

The ID of the order.

`orderRepository` [IOrderRepository](#)

The repository for accessing order data.

`httpContext` [HttpContext](#)

The HTTP context of the request.

Returns

[Task](#) < [IResult](#) >

The order with the specified ID.

UpdateOrderAsync(int, IOrderRepository, HttpContext, IProductRepository, UpdateOrderRequest)

Updates an existing order.

```
[Authorize(Roles = "user,admin", AuthenticationSchemes = "Bearer")]
public static Task<IResult> UpdateOrderAsync(int orderId, IOrderRepository orderRepository,
HttpContext httpContext, IProductRepository productRepository, UpdateOrderRequest req
= null)
```

Parameters

orderId [int](#)

The ID of the order to update.

orderRepository [IOrderRepository](#)

The repository for accessing order data.

httpContext [HttpContext](#)

The HTTP context of the request.

productRepository [IProductRepository](#)

The repository for accessing product data.

req [UpdateOrderRequest](#)

The update order request.

Returns

Task <|Result|>

The updated order.

Class ProductHandlers

Namespace: [Logistics_Supplier1_API.Handlers](#)

Assembly: Logistics-Supplier1-API.dll

Handles API requests related to products.

```
public class ProductHandlers
```

Inheritance

[object](#) ← ProductHandlers

Inherited Members

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

Methods

CreateProductAsync(IProductRepository, CreateProductRequest)

Creates a new product.

```
[Authorize(Roles = "admin")]
public static Task<IResult> CreateProductAsync(IProductRepository productRepository,
CreateProductRequest req = null)
```

Parameters

productRepository [IProductRepository](#)

The repository for accessing product data.

req [CreateProductRequest](#)

The create product request.

Returns

[Task](#)<[IResult](#)>

The created product.

DeleteProductAsync(string, IProductRepository)

Deletes a product.

```
[Authorize(Roles = "admin")]
public static Task<IResult> DeleteProductAsync(string sku, IProductRepository
productRepository)
```

Parameters

sku [string](#)

The SKU of the product to delete.

productRepository [IProductRepository](#)

The repository for accessing product data.

Returns

[Task](#)<[IResult](#)>

A result indicating whether the product was deleted successfully.

GetAllProducts(IProductRepository)

Gets all products.

```
public static Task<IResult> GetAllProducts(IProductRepository productRepository)
```

Parameters

productRepository [IProductRepository](#)

The repository for accessing product data.

Returns

[Task](#) <IResult>>

A list of all products.

GetProductBySkuAsync(string, IProductRepository)

Gets a product by its SKU.

```
public static Task<IResult> GetProductBySkuAsync(string sku, IProductRepository  
productRepository)
```

Parameters

sku [string](#)

The SKU of the product.

productRepository [IProductRepository](#)

The repository for accessing product data.

Returns

[Task](#) <IResult>>

The product with the specified SKU.

UpdateProductAsync(string, IProductRepository, UpdateProductRequest)

Updates an existing product.

```
[Authorize(Roles = "admin")]  
public static Task<IResult> UpdateProductAsync(string sku, IProductRepository  
productRepository, UpdateProductRequest req = null)
```

Parameters

sku [string](#)

The SKU of the product to update.

productRepository [IProductRepository](#)

The repository for accessing product data.

req [UpdateProductRequest](#)

The update product request.

Returns

[Task](#) <[IResult](#)>

The updated product.

Class RegistrationHandler

Namespace: [Logistics_Supplier1_API.Handlers](#)

Assembly: Logistics-Supplier1-API.dll

Handles user registration requests.

```
public static class RegistrationHandler
```

Inheritance

[object](#) ← RegistrationHandler

Inherited Members

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

Methods

RegisterUser(IUserRepository, IInviteCodeRepository, CreateUserRequest)

Registers a new user.

```
public static Task<IResult> RegisterUser(IUserRepository userRepository,  
IInviteCodeRepository inviteCodeRepository, CreateUserRequest createUserRequest = null)
```

Parameters

userRepository [IUserRepository](#)

The repository for accessing user data.

inviteCodeRepository [IInviteCodeRepository](#)

The repository for accessing invite code data.

createUserRequest [CreateUserRequest](#)

The create user request.

Returns

[Task](#) <[IResult](#)>

An IResult indicating the success or failure of the registration.

Class UserHandlers

Namespace: [Logistics_Supplier1_API.Handlers](#)

Assembly: Logistics-Supplier1-API.dll

Handles API requests related to users.

```
public static class UserHandlers
```

Inheritance

[object](#) ← UserHandlers

Inherited Members

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

Methods

DeleteUserAsync(string, IUserRepository)

Deletes a user.

```
[Authorize(Roles = "admin")]
public static Task<IResult> DeleteUserAsync(string username, IUserRepository userRepository)
```

Parameters

username [string](#)

The username of the user to delete.

userRepository [IUserRepository](#)

The repository for accessing user data.

Returns

[Task](#)<[IResult](#)>

A result indicating whether the user was deleted successfully.

GetAllUsersAsync(IUserRepository)

Gets all users.

```
[Authorize(Roles = "admin")]
public static Task<IResult> GetAllUsersAsync(IUserRepository userRepository)
```

Parameters

userRepository [IUserRepository](#)

The repository for accessing user data.

Returns

[Task](#) <[IResult](#)>

A list of all users.

GetUserByUsernameAsync(string, IUserRepository)

Gets a user by their username.

```
[Authorize(Roles = "admin")]
public static Task<IResult> GetUserByUsernameAsync(string username,
IUserRepository userRepository)
```

Parameters

username [string](#)

The username of the user.

userRepository [IUserRepository](#)

The repository for accessing user data.

Returns

[Task](#) <[IResult](#)>

The user with the specified username.

UpdateUserAsync(string, IUserRepository, UpdateUserRequest)

Updates an existing user.

```
[Authorize(Roles = "admin")]
public static Task<IResult> UpdateUserAsync(string username, IUserRepository userRepository,
UpdateUserRequest req = null)
```

Parameters

username [string](#)

The username of the user to update.

userRepository [IUserRepository](#)

The repository for accessing user data.

req [UpdateUserRequest](#)

The update user request.

Returns

[Task](#) <[IResult](#)>

The updated user.

Namespace Logistics_Supplier1_API.Helpers

Classes

[JwtHelper](#)

Provides helper methods for working with JWT (JSON Web Token).

Class JwtHelper

Namespace: [Logistics_Supplier1_API.Helpers](#)

Assembly: Logistics-Supplier1-API.dll

Provides helper methods for working with JWT (JSON Web Token).

```
public class JwtHelper
```

Inheritance

[object](#) ← JwtHelper

Inherited Members

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

Methods

GetRoleFromJwt(HttpContext)

Gets the user's role from the JWT token in the HTTP context.

```
public static string GetRoleFromJwt(HttpContext httpContext)
```

Parameters

[httpContext](#) [HttpContext](#)

The HTTP context.

Returns

[string](#)

The user's role if found in the JWT token, otherwise null.

GetUserIdFromJwt(HttpContext)

Gets the user ID from the JWT token in the HTTP context.

```
public static int? GetUserIdFromJwt(HttpContext httpContext)
```

Parameters

httpContext [HttpContext](#)

The HTTP context.

Returns

[int](#)?

The user ID if found in the JWT token, otherwise null.

Namespace Logistics_Supplier1_API.Models

Classes

[InviteCode](#)

Represents an invite code in the system.

[Order](#)

Represents an order in the system.

[OrderItem](#)

Represents an item within an order.

[Product](#)

Represents a product in the system.

[User](#)

Represents a user in the system.

Class InviteCode

Namespace: [Logistics_Supplier1_API.Models](#)

Assembly: Logistics-Supplier1-API.dll

Represents an invite code in the system.

```
public class InviteCode
```

Inheritance

[object](#) ← InviteCode

Inherited Members

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

Properties

Code

Gets or sets the code itself.

```
[Required]  
[MaxLength(50)]  
public string Code { get; }
```

Property Value

[string](#)

ExpirationDate

Gets or sets the expiration date of the invite code.

```
public DateTime ExpirationDate { get; }
```

Property Value

[DateTime](#)

Id

Gets or sets the unique identifier of the invite code.

```
public int Id { get; }
```

Property Value

[int](#)

UsageCount

Gets or sets the number of times the code has been used.

```
public int UsageCount { get; }
```

Property Value

[int](#)

Methods

CreateNewInviteCode(string, int, DateTime)

Creates a new invite code with the specified details.

```
public InviteCode CreateNewInviteCode(string newCode, int usageCount,  
DateTime expirationDate)
```

Parameters

newCode [string](#)

The code itself.

usageCount [int ↗](#)

The initial usage count.

expirationDate [DateTime ↗](#)

The expiration date of the code.

Returns

[InviteCode](#)

The created invite code instance.

Class Order

Namespace: [Logistics_Supplier1_API.Models](#)

Assembly: Logistics-Supplier1-API.dll

Represents an order in the system.

```
public class Order
```

Inheritance

[object](#) ← Order

Inherited Members

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

Constructors

Order()

Initializes a new instance of the [Order](#) class (required for Entity Framework Core).

```
public Order()
```

Order(int, List<OrderItem>)

Initializes a new instance of the [Order](#) class.

```
public Order(int userId, List<OrderItem> orderItems)
```

Parameters

userId [int](#)

The ID of the user who placed the order.

orderItems [List](#)<[OrderItem](#)>

The list of order items.

Properties

Id

Gets or sets the unique identifier of the order.

```
public int Id { get; }
```

Property Value

[int](#)

LastUpdate

Gets or sets the date and time when the order was last updated.

```
public DateTime LastUpdate { get; }
```

Property Value

[DateTime](#)

OrderDate

Gets or sets the date and time when the order was placed.

```
public DateTime OrderDate { get; }
```

Property Value

[DateTime](#)

OrderItems

Gets or sets the list of order items associated with the order.

[Required]

```
public List<OrderItem> OrderItems { get; }
```

Property Value

[List](#)<[OrderItem](#)>

UserId

Gets or sets the ID of the user who placed the order.

```
public int UserId { get; set; }
```

Property Value

[int](#)

Methods

updateOrder(int, List<OrderItem>)

Updates the order with new details.

```
public Order updateOrder(int id, List<OrderItem> orderItems)
```

Parameters

id [int](#)

The ID of the order.

orderItems [List](#)<[OrderItem](#)>

The new list of order items.

Returns

Order

The updated order instance.

Class OrderItem

Namespace: [Logistics_Supplier1_API.Models](#)

Assembly: Logistics-Supplier1-API.dll

Represents an item within an order.

```
public class OrderItem
```

Inheritance

[object](#) ← OrderItem

Inherited Members

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

Constructors

OrderItem()

Initializes a new instance of the [OrderItem](#) class (required for Entity Framework Core).

```
public OrderItem()
```

OrderItem(string, int, decimal)

Initializes a new instance of the [OrderItem](#) class.

```
public OrderItem(string productSku, int quantity, decimal price)
```

Parameters

productSku [string](#)

The SKU of the product.

quantity [int](#)

The quantity of the product.

price [decimal](#)

The unit price of the product.

Properties

Id

Gets or sets the unique identifier of the order item.

```
public int Id { get; }
```

Property Value

[int](#)

ProductSku

Gets or sets the SKU (Stock Keeping Unit) of the product in the order item.

```
[Required]  
[MaxLength(50)]  
public string ProductSku { get; set; }
```

Property Value

[string](#)

Quantity

Gets or sets the quantity of the product in the order item.

```
[Required]  
public int Quantity { get; set; }
```

Property Value

[int ↗](#)

UnitPrice

Gets or sets the unit price of the product in the order item.

[Required]

```
public decimal UnitPrice { get; set; }
```

Property Value

[decimal ↗](#)

Class Product

Namespace: [Logistics_Supplier1_API.Models](#)

Assembly: Logistics-Supplier1-API.dll

Represents a product in the system.

```
public class Product
```

Inheritance

[object](#) ← Product

Inherited Members

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

Constructors

Product(string, string, decimal, int)

Initializes a new instance of the [Product](#) class.

```
public Product(string sku, string name, decimal price, int stockQuantity)
```

Parameters

sku [string](#)

The SKU of the product.

name [string](#)

The name of the product.

price [decimal](#)

The price of the product.

stockQuantity [int](#)

The stock quantity of the product.

Properties

Id

Gets or sets the unique identifier of the product.

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

Property Value

[int ↗](#)

Name

Gets or sets the name of the product.

```
[Required]  
[MaxLength(50)]  
public string Name { get; }
```

Property Value

[string ↗](#)

Price

Gets or sets the price of the product.

```
[Required]  
[Range(0.01, 1.7976931348623157E+308)]  
public decimal Price { get; }
```

Property Value

[decimal ↗](#)

Sku

Gets or sets the SKU (Stock Keeping Unit) of the product.

```
[Required]  
[MaxLength(100)]  
public string Sku { get; }
```

Property Value

[string](#) ↗

StockQuantity

Gets or sets the stock quantity of the product.

```
[Required]  
[Range(0, 2147483647)]  
public int StockQuantity { get; }
```

Property Value

[int](#) ↗

Methods

DecreaseStockQuantity(int)

Decreases the stock quantity of the product.

```
public void DecreaseStockQuantity(int quantity)
```

Parameters

quantity [int](#) ↗

The quantity to decrease by.

Exceptions

[ArgumentException](#)

Thrown when the quantity is not positive or is greater than the current stock quantity.

UpdateProduct(string, decimal, int)

Updates the product's information.

```
public void UpdateProduct(string name, decimal price, int stockQuantity)
```

Parameters

[name](#) [string](#)

The new name of the product.

[price](#) [decimal](#)

The new price of the product.

[stockQuantity](#) [int](#)

The new stock quantity of the product.

Class User

Namespace: [Logistics_Supplier1_API.Models](#)

Assembly: Logistics-Supplier1-API.dll

Represents a user in the system.

```
public class User
```

Inheritance

[object](#) ← User

Inherited Members

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

Constructors

User(string, string, string, string)

Initializes a new instance of the [User](#) class.

```
public User(string username, string passwordHash, string role, string address)
```

Parameters

username [string](#)

The username of the user.

passwordHash [string](#)

The hashed password of the user.

role [string](#)

The role of the user.

address [string](#)

The address of the user.

Properties

Address

Gets or sets the address of the user.

```
[Required]  
[MaxLength(100)]  
public string Address { get; }
```

Property Value

[string](#)

Id

Gets or sets the unique identifier of the user.

```
[MaxLength(100)]  
public int Id { get; }
```

Property Value

[int](#)

PasswordHash

Gets or sets the hashed password of the user.

```
[Required]  
public string PasswordHash { get; }
```

Property Value

[string](#)

Role

Gets or sets the role of the user.

```
[Required]  
[MaxLength(15)]  
public string Role { get; }
```

Property Value

[string](#)

Username

Gets or sets the username of the user.

```
[Required]  
[MinLength(3)]  
[MaxLength(30)]  
public string Username { get; }
```

Property Value

[string](#)

Methods

UpdateUser(string, string)

Updates the user's information.

```
public void UpdateUser(string role = null, string address = null)
```

Parameters

role [string](#)

The new role of the user (optional).

address [string](#) ↗

The new address of the user (optional).

Namespace Logistics_Supplier1_API. Repositories

Classes

[InviteCodeRepository](#)

Provides methods for interacting with InviteCode data in the database.

[OrderRepository](#)

Provides methods for interacting with Order data in the database.

[ProductRepository](#)

Provides methods for interacting with Product data in the database.

[UserRepository](#)

Provides methods for interacting with User data in the database.

Interfaces

[IInviteCodeRepository](#)

Defines methods for interacting with InviteCode data.

[IOrderRepository](#)

Defines methods for interacting with Order data.

[IProductRepository](#)

Defines methods for interacting with Product data.

[IUserRepository](#)

Defines methods for interacting with User data.

Interface IInviteCodeRepository

Namespace: [Logistics_Supplier1_API.Repositories](#)

Assembly: Logistics-Supplier1-API.dll

Defines methods for interacting with InviteCode data.

```
public interface IInviteCodeRepository
```

Methods

GenerateInviteCodeAsync()

Generates a new invite code asynchronously.

```
Task<InviteCode> GenerateInviteCodeAsync()
```

Returns

[Task](#) <[InviteCode](#)>

A task representing the asynchronous operation. The task result contains the generated invite code.

GetAllInviteCodesAsync()

Gets all invite codes asynchronously.

```
Task<List<InviteCode>> GetAllInviteCodesAsync()
```

Returns

[Task](#) <[List](#) <[InviteCode](#)>>

A task representing the asynchronous operation. The task result contains a list of all invite codes.

IsValidInviteCodeAsync(string)

Checks if an invite code is valid asynchronously.

```
Task<bool> IsValidInviteCodeAsync(string inviteCode)
```

Parameters

`inviteCode` [string](#)

The invite code to check.

Returns

[Task](#) <[bool](#)>

A task representing the asynchronous operation. The task result contains true if the invite code is valid, otherwise false.

Interface IOrderRepository

Namespace: [Logistics_Supplier1_API.Repositories](#)

Assembly: Logistics-Supplier1-API.dll

Defines methods for interacting with Order data.

```
public interface IOrderRepository
```

Methods

CreateOrderAsync(Order)

Creates a new order asynchronously.

```
Task<int> CreateOrderAsync(Order order)
```

Parameters

order [Order](#)

The order to create.

Returns

[Task](#) <int>

A task representing the asynchronous operation. The task result contains the ID of the created order.

DeleteOrderAsync(Order)

Deletes an order asynchronously.

```
Task<Order> DeleteOrderAsync(Order order)
```

Parameters

`order` [Order](#)

The order to delete.

Returns

[Task](#) <[Order](#)>

A task representing the asynchronous operation. The task result contains the deleted order.

GetAllOrdersAsync()

Gets all orders asynchronously.

`Task<List<Order>> GetAllOrdersAsync()`

Returns

[Task](#) <[List](#) <[Order](#)>>

A task representing the asynchronous operation. The task result contains a list of all orders.

GetOrderByIdAsync(int)

Gets an order by its ID asynchronously.

`Task<Order> GetOrderByIdAsync(int orderId)`

Parameters

`orderId` [int](#)

The ID of the order to retrieve.

Returns

[Task](#) <[Order](#)>

A task representing the asynchronous operation. The task result contains the order if found, otherwise null.

GetOrdersByUserIdAsync(int)

Gets all orders for a specific user asynchronously.

```
Task<List<Order>> GetOrdersByUserIdAsync(int userId)
```

Parameters

[userId](#) [int](#)

The ID of the user.

Returns

[Task](#) <[List](#)<[Order](#)>>

A task representing the asynchronous operation. The task result contains a list of orders for the specified user.

UpdateOrderAsync(Order)

Updates an existing order asynchronously.

```
Task<Order> UpdateOrderAsync(Order order)
```

Parameters

[order](#) [Order](#)

The order to update.

Returns

[Task](#) <[Order](#)>

A task representing the asynchronous operation. The task result contains the updated order.

Interface IProductRepository

Namespace: [Logistics_Supplier1_API.Repositories](#)

Assembly: Logistics-Supplier1-API.dll

Defines methods for interacting with Product data.

```
public interface IProductRepository
```

Methods

CreateProductAsync(Product)

Creates a new product asynchronously.

```
Task<Product> CreateProductAsync(Product product)
```

Parameters

product [Product](#)

The product to create.

Returns

[Task](#) <[Product](#)>

A task representing the asynchronous operation. The task result contains the created product.

DeleteProductAsync(Product)

Deletes a product asynchronously.

```
Task<Product> DeleteProductAsync(Product product)
```

Parameters

product [Product](#)

The product to delete.

Returns

[Task](#) <[Product](#)>

A task representing the asynchronous operation. The task result contains the deleted product.

GetAllProductsAsync()

Gets all products asynchronously.

`Task<List<Product>> GetAllProductsAsync()`

Returns

[Task](#) <[List](#) <[Product](#)>>

A task representing the asynchronous operation. The task result contains a list of all products.

GetProductBySkuAsync(string)

Gets a product by its SKU asynchronously.

`Task<Product> GetProductBySkuAsync(string sku)`

Parameters

sku [string](#)

The SKU of the product to retrieve.

Returns

[Task](#) <[Product](#)>

A task representing the asynchronous operation. The task result contains the product if found, otherwise null.

UpdateProductAsync(Product)

Updates an existing product asynchronously.

Task<Product> **UpdateProductAsync**(Product product)

Parameters

product [Product](#)

The product to update.

Returns

[Task](#) <Product>

A task representing the asynchronous operation. The task result contains the updated product.

Interface IUserRepository

Namespace: [Logistics_Supplier1_API.Repositories](#)

Assembly: Logistics-Supplier1-API.dll

Defines methods for interacting with User data.

```
public interface IUserRepository
```

Methods

CreateUserAsync(User)

Creates a new user asynchronously.

```
Task<User> CreateUserAsync(User user)
```

Parameters

user [User](#)

The user to create.

Returns

[Task](#) <[User](#)>

A task representing the asynchronous operation. The task result contains the created user.

DeleteUserAsync(User)

Deletes a user asynchronously.

```
Task<User> DeleteUserAsync(User user)
```

Parameters

`user` [User](#)

The user to delete.

Returns

[Task](#) <[User](#)>

A task representing the asynchronous operation. The task result contains the deleted user.

GetAllUsersAsync()

Gets all users asynchronously.

`Task<List< GetAllUsersRequest>> GetAllUsersAsync()`

Returns

[Task](#) <[List](#) <[GetAllUsersRequest](#)>>

A task representing the asynchronous operation. The task result contains a list of all users.

GetUserByUsernameAsync(string)

Gets a user by their username asynchronously.

`Task<User> GetUserByUsernameAsync(string username)`

Parameters

`username` [string](#)

The username of the user to retrieve.

Returns

[Task](#) <[User](#)>

A task representing the asynchronous operation. The task result contains the user if found, otherwise null.

UpdateUserAsync(User)

Updates an existing user asynchronously.

Task<User> **UpdateUserAsync**(User user)

Parameters

user [User](#)

The user to update.

Returns

[Task](#) <User>

A task representing the asynchronous operation. The task result contains the updated user.

Class InviteCodeRepository

Namespace: [Logistics_Supplier1_API.Repositories](#)

Assembly: Logistics-Supplier1-API.dll

Provides methods for interacting with InviteCode data in the database.

```
public class InviteCodeRepository : IInviteCodeRepository
```

Inheritance

[object](#) ← InviteCodeRepository

Implements

[IInviteCodeRepository](#).

Inherited Members

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

Constructors

InviteCodeRepository(MyDbContext)

Initializes a new instance of the InviteCodeRepository class.

```
public InviteCodeRepository(MyDbContext context)
```

Parameters

context [MyDbContext](#)

The database context.

Methods

GenerateInviteCodeAsync()

Generates a new invite code asynchronously.

```
public Task<InviteCode> GenerateInviteCodeAsync()
```

Returns

[Task](#) <[InviteCode](#)>

A task representing the asynchronous operation. The task result contains the generated invite code.

GetAllInviteCodesAsync()

Gets all invite codes asynchronously.

```
public Task<List<InviteCode>> GetAllInviteCodesAsync()
```

Returns

[Task](#) <[List](#) <[InviteCode](#)>>

A task representing the asynchronous operation. The task result contains a list of all invite codes.

IsValidInviteCodeAsync(string)

Checks if an invite code is valid asynchronously.

```
public Task<bool> IsValidInviteCodeAsync(string inviteCode)
```

Parameters

inviteCode [string](#)

The invite code to check.

Returns

[Task](#) <[bool](#)>

A task representing the asynchronous operation. The task result contains true if the invite code is valid, otherwise false.

Class OrderRepository

Namespace: [Logistics_Supplier1_API.Repositories](#)

Assembly: Logistics-Supplier1-API.dll

Provides methods for interacting with Order data in the database.

```
public class OrderRepository : IOrderRepository
```

Inheritance

[object](#) ← OrderRepository

Implements

[IOrderRepository](#)

Inherited Members

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

Constructors

OrderRepository(MyDbContext)

Initializes a new instance of the OrderRepository class.

```
public OrderRepository(MyDbContext context)
```

Parameters

context [MyDbContext](#)

The database context.

Methods

CreateOrderAsync(Order)

Creates a new order asynchronously.

```
public Task<int> CreateOrderAsync(Order order)
```

Parameters

`order` [Order](#)

The order to create.

Returns

[Task](#)<int>

A task representing the asynchronous operation. The task result contains the ID of the created order.

DeleteOrderAsync(Order)

Deletes an order asynchronously.

```
public Task<Order> DeleteOrderAsync(Order order)
```

Parameters

`order` [Order](#)

The order to delete.

Returns

[Task](#)<Order>

A task representing the asynchronous operation. The task result contains the deleted order.

GetAllOrdersAsync()

Gets all orders asynchronously.

```
public Task<List<Order>> GetAllOrdersAsync()
```

Returns

[Task](#) <[List](#) <[Order](#)>>

A task representing the asynchronous operation. The task result contains a list of all orders.

GetOrderByIdAsync(int)

Gets an order by its ID asynchronously.

```
public Task<Order> GetOrderByIdAsync(int orderId)
```

Parameters

orderId [int](#)

The ID of the order to retrieve.

Returns

[Task](#) <[Order](#)>

A task representing the asynchronous operation. The task result contains the order if found, otherwise null.

GetOrdersByUserIdAsync(int)

Gets all orders for a specific user asynchronously.

```
public Task<List<Order>> GetOrdersByUserIdAsync(int userId)
```

Parameters

userId [int](#)

The ID of the user.

Returns

[Task](#) <[List](#)<[Order](#)>>

A task representing the asynchronous operation. The task result contains a list of orders for the specified user.

UpdateOrderAsync(Order)

Updates an existing order asynchronously.

```
public Task<Order> UpdateOrderAsync(Order order)
```

Parameters

order [Order](#)

The order to update.

Returns

[Task](#) <[Order](#)>

A task representing the asynchronous operation. The task result contains the updated order.

Class ProductRepository

Namespace: [Logistics_Supplier1_API.Repositories](#)

Assembly: Logistics-Supplier1-API.dll

Provides methods for interacting with Product data in the database.

```
public class ProductRepository : IProductRepository
```

Inheritance

[object](#) ← ProductRepository

Implements

[IProductRepository](#).

Inherited Members

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

Constructors

ProductRepository(MyDbContext)

Initializes a new instance of the ProductRepository class.

```
public ProductRepository(MyDbContext context)
```

Parameters

context [MyDbContext](#)

The database context.

Methods

CreateProductAsync(Product)

Creates a new product asynchronously.

```
public Task<Product> CreateProductAsync(Product product)
```

Parameters

product [Product](#)

The product to create.

Returns

[Task](#) <Product>

A task representing the asynchronous operation. The task result contains the created product.

DeleteProductAsync(Product)

Deletes a product asynchronously.

```
public Task<Product> DeleteProductAsync(Product product)
```

Parameters

product [Product](#)

The product to delete.

Returns

[Task](#) <Product>

A task representing the asynchronous operation. The task result contains the deleted product.

GetAllProductsAsync()

Gets all products asynchronously.

```
public Task<List<Product>> GetAllProductsAsync()
```

Returns

[Task](#) <[List](#) <[Product](#)>>

A task representing the asynchronous operation. The task result contains a list of all products.

GetProductBySkuAsync(string)

Gets a product by its SKU asynchronously.

```
public Task<Product> GetProductBySkuAsync(string sku)
```

Parameters

sku [string](#)

The SKU of the product to retrieve.

Returns

[Task](#) <[Product](#)>

A task representing the asynchronous operation. The task result contains the product if found, otherwise null.

UpdateProductAsync(Product)

Updates an existing product asynchronously.

```
public Task<Product> UpdateProductAsync(Product product)
```

Parameters

product [Product](#)

The product to update.

Returns

[Task](#) <Product>

A task representing the asynchronous operation. The task result contains the updated product.

Class UserRepository

Namespace: [Logistics_Supplier1_API.Repositories](#)

Assembly: Logistics-Supplier1-API.dll

Provides methods for interacting with User data in the database.

```
public class UserRepository : IUserRepository
```

Inheritance

[object](#) ← UserRepository

Implements

[IUserRepository](#)

Inherited Members

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

Constructors

UserRepository(MyDbContext)

Initializes a new instance of the UserRepository class.

```
public UserRepository(MyDbContext dbcontext)
```

Parameters

dbcontext [MyDbContext](#)

The database context.

Methods

CreateUserAsync(User)

Creates a new user asynchronously.

```
public Task<User> CreateUserAsync(User user)
```

Parameters

user [User](#)

The user to create.

Returns

[Task](#) <User>

A task representing the asynchronous operation. The task result contains the created user.

DeleteUserAsync(User)

Deletes a user asynchronously.

```
public Task<User> DeleteUserAsync(User user)
```

Parameters

user [User](#)

The user to delete.

Returns

[Task](#) <User>

A task representing the asynchronous operation. The task result contains the deleted user.

GetAllUsersAsync()

Gets all users asynchronously.

```
public Task<List< GetAllUsersRequest >> GetAllUsersAsync()
```

Returns

[Task](#) < [List](#) < [GetAllUsersRequest](#) >

A task representing the asynchronous operation. The task result contains a list of all users.

GetUserByUsernameAsync(string)

Gets a user by their username asynchronously.

```
public Task<User> GetUserByUsernameAsync(string username)
```

Parameters

username [string](#)

The username of the user to retrieve.

Returns

[Task](#) < [User](#) >

A task representing the asynchronous operation. The task result contains the user if found, otherwise null.

UpdateUserAsync(User)

Updates an existing user asynchronously.

```
public Task<User> UpdateUserAsync(User user)
```

Parameters

user [User](#)

The user to update.

Returns

[Task](#) <User>

A task representing the asynchronous operation. The task result contains the updated user.