

Completing the CRUD Operations Using the Data Layer



Alex Wolf

.NET Developer

www.thecodewolf.com



Understanding CRUD Operations

Create

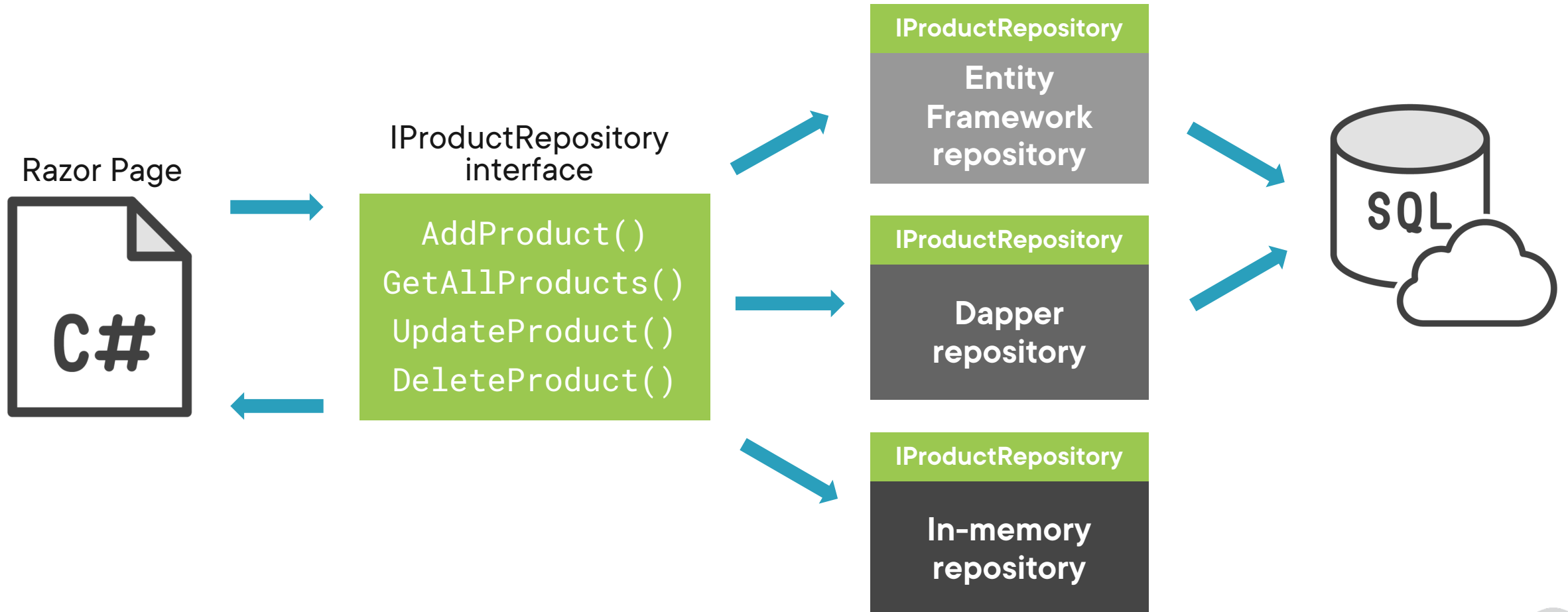
Read

Update

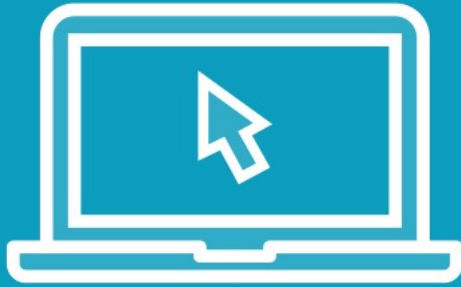
Delete



Understanding the Repository Pattern



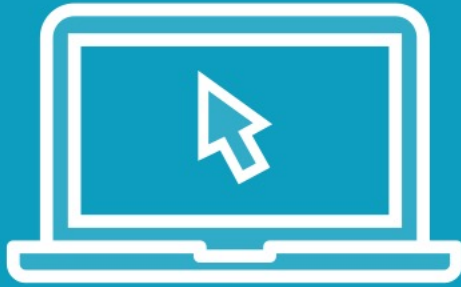
Demo



Creating a repository class



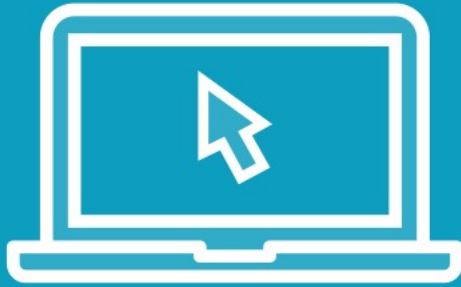
Demo



Retrieving the list of products



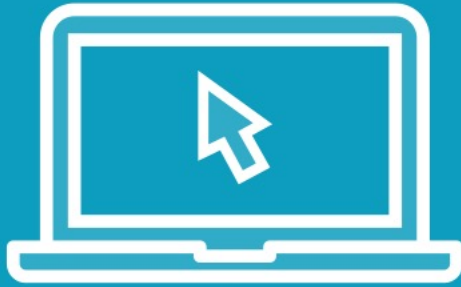
Demo



Seeding additional product data



Demo



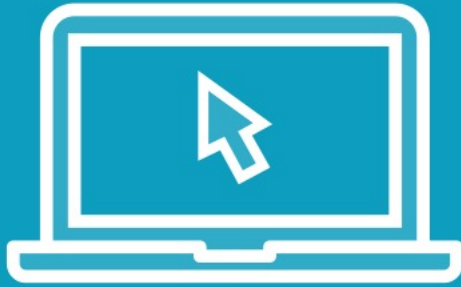
Refactoring the add product page



A note about add and edit
functionality.



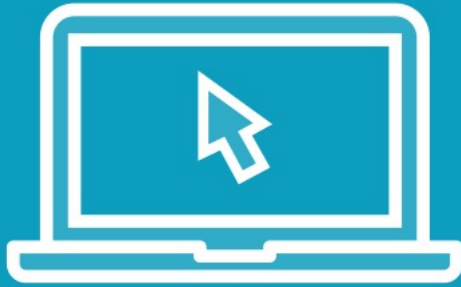
Demo



Implementing edit functionality



Demo



Deleting a product



Summary



Most data driven apps required a complete set of create, read, update and delete operations (CRUD)

A repository layer creates an abstraction over data access classes to provide loose coupling

Loosely coupled services can be created using an interface and a matching implementation class

Services and repositories can be registered with the app DI container for use throughout the app

Entity Framework can seed data by overriding the OnModelCreating method of DbContext

Razor Pages support multiple handler methods for the same form using asp-page-handler

