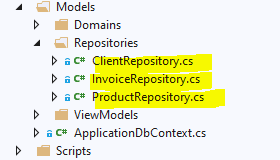
Operacje na Bazie danych



Wszelakich zmian związanych za operacjami na bazie danych musimy przeprowadzić tylko w plikach repozytorium. Dzięki temu, że mamy dobrze rozdzielone warstwy.

Zaczynamy od ClientRepository.

|  |
| --- |
| public class ClientRepository  {  public List<Client> GetClients(string userId)  {  using (var context = new ApplicationDbContext())  {  return context.Clients.Where(x => x.UserId == userId).ToList();  }  }  } |

|  |
| --- |
| public class ProductRepository  {  public List<Product> GetProducts()  {  using (var context = new ApplicationDbContext())  {  return context.Products.ToList();  }  }  public Product GetProduct(int productId)  {  using (var context = new ApplicationDbContext())  {  // może być First(), FirstOrDefault()  // Single jeżeli nie znajdzie lub będzie więcej produktów o danym Id  // To w tym miejscu zostanie rzucony wyjątek  return context.Products.Single(x => x.Id == productId);  }  }  } |

|  |
| --- |
| using System.Data.Entity;    public List<Invoice> GetInvoices(string userId)  {  using (var context = new ApplicationDbContext())  {  return context.Invoices.Include(x => x.Client )  .Where(x => x.UserId == userId)  .ToList();  }  } |

Include() ponieważ chcemy załączyć informacje o kliencie dla każdej faktury.  
Aby Include zadziałało musimy dodać namespace **System.Data.Entity**