Q 1 ) Implement an online shopping website using Asp.Net MVC with

Entity Framework. The appliaction must have 3 tables . Products. Orders and

Customer .There is a 1-to-Many relation from Product table to Orders table

Also there is a 1-to-Many relation from Customer table to Orders table.

Create Model classes, data context class, do migration and implement all the

CRUD operations.

Ans :-

Model :

public class Product

{

public int ProductId { get; set; }

public string Name { get; set; }

public decimal Price { get; set; }

public virtual ICollection<Order> Orders { get; set; }

}

public class Customer

{

public int CustomerId { get; set; }

public string Name { get; set; }

public virtual ICollection<Order> Orders { get; set; }

}

public class Order

{

public int OrderId { get; set; }

public int ProductId { get; set; }

public int CustomerId { get; set; }

public virtual Product Product { get; set; }

public virtual Customer Customer { get; set; }

}

Create data context class :

public class ShoppingDbContext : DbContext

{

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

public DbSet<Customer> Customers { get; set; }

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

protected override void OnModelCreating(ModelBuilder modelBuilder)

{

modelBuilder.Entity<Order>()

.HasOne(o => o.Product)

.WithMany(p => p.Orders)

.HasForeignKey(o => o.ProductId);

modelBuilder.Entity<Order>()

.HasOne(o => o.Customer)

.WithMany(c => c.Orders)

.HasForeignKey(o => o.CustomerId);

}

}

CRUD :

public IActionResult CreateOrder(Order order)

{

if (ModelState.IsValid)

{

\_context.Orders.Add(order);

\_context.SaveChanges();

return RedirectToAction("Index");

}

return View(order);

}

public IActionResult EditOrder(int id)

{

Order order = \_context.Orders.Find(id);

if (order == null)

{

return NotFound();

}

return View(order);

}

public IActionResult EditOrder(int id, Order order)

{

if (id != order.OrderId)

{

return NotFound();

}

if (ModelState.IsValid)

{

\_context.Orders.Update(order);

\_context.SaveChanges();

return RedirectToAction("Index");

}