23DCS037

Internal Exam .NET

SET 2

OnlineBookStore (Database)

```
create database OnlineBookStore
go

use OnlineBookStore
go

create table books(
bookid int primary key,
title nvarchar(100) not null,
author nvarchar(100),
price decimal(10,2) not null,
stock int not null,
)

create table customers(
customerid int primary key,
name nvarchar(25) not null,
email nvarchar(50) unique,
phone nvarchar(15),
)

create table orders(
orderid int primary key,
```

```
customerid int not null,
orderdate datetime default getdate(),
foreign key (customerid) references customers(customerid)
)

create table orderdetails(
orderdetailid int primary key,
orderid int not null,
bookid int not null,
quantity int not null,
foreign key (orderid) references orders(orderid),
foreign key (bookid) references books(bookid)
)

insert into books(bookid,title,author,price,stock) values
(1,'C# Programming','Kathan Patel',700,5),
(2,'C Programming','Manthan Shah',800,5),
(3,'C++ Programming','Anshul Jangid',900,5),
(4,'Java Programming','Evan Gregor',600,5),
(5,'Python Programming','Tirth Ganatra',500,5);

insert into customers(customerid,name,email,phone) values
(101,'Stany Gregor','stanygregor@gmail.com',1234567890),
(102,'Divyansh Thakur','divyanshthakur@gmail.com',9876543210),
(103,'James Dhandhukiya','jamesdhandhukiya@gmail.com', 6543210987);
```

23DCS037

#Program.cs

```
using System;
using System.Data;
using Microsoft.Data.SqlClient;

namespace OnlineBookStoreApp
{
   class Program
   {
      static void Main(string[] args)
      {

         string connectionString = @"Data Source=EVAN\SQLEXPRESS;Initial
Catalog=OnlineBookStore;Integrated Security=True;Trust Server Certificate=True";

         using (SqlConnection conn = new SqlConnection(connectionString))
         {
            try
            {

               SqlDataAdapter da = new SqlDataAdapter("SELECT * FROM Books", conn);
               SqlCommandBuilder builder = new SqlCommandBuilder(da);

               DataSet ds = new DataSet();

               da.Fill(ds, "Books");
               ds.Tables["Books"].PrimaryKey = new DataColumn[] {
ds.Tables["Books"].Columns["bookid"] };
```

```
            Console.WriteLine("===== Available Books =====");
            foreach (DataRow row in ds.Tables["Books"].Rows)
            {
                Console.WriteLine($"{row["bookid"]} - {row["title"]} by {row["author"]} | Price:
{row["price"]} | Stock: {row["stock"]}");
            }

            Console.WriteLine("\nEnter Book ID to update stock:");
            int bookId = Convert.ToInt32(Console.ReadLine());

            DataRow bookRow = ds.Tables["Books"].Rows.Find(bookId);

            if (bookRow != null)
            {
                Console.WriteLine($"Current Stock for '{bookRow["title"]}':
{bookRow["stock"]}");

                Console.WriteLine("Enter quantity to add:");
                int qty = Convert.ToInt32(Console.ReadLine());

                bookRow["stock"] = Convert.ToInt32(bookRow["stock"]) + qty;

                da.Update(ds, "Books");

                Console.WriteLine($"Stock updated successfully! New Stock:
{bookRow["stock"]}");
            }
            else
```

```
            {
                Console.WriteLine("Book not found!");
            }
        }
        catch (Exception ex)
        {
            Console.WriteLine("Error: " + ex.Message);
        }
    }

    Console.WriteLine("\nPress any key to exit...");
    Console.ReadKey();
  }
 }
}
```

Output: