

ASSIGNMENT II

Submitted by ,
Vaishnav Ratheesh
55

Q.Create a console app demonstrating CRUD operations using ADO.NET DataSet with a sample database

```
Imports System.Data.SqlClient
Module Program
' Connection string to the sample database
Private connectionString As String = "Data Source=YourServer;Initial
Catalog=YourDatabase;Integrated Security=True"
Sub Main(args As String())
' Create and display menu
While True
Console.WriteLine("1. Add Product")
Console.WriteLine("2. View Products")
Console.WriteLine("3. Update Product")
Console.WriteLine("4. Delete Product")
Console.WriteLine("5. Exit")
Console.WriteLine("Enter your choice:")
Dim choice As Integer = Convert.ToInt32(Console.ReadLine())
Select Case choice
Case 1
AddProduct()
Case 2
ViewProducts()

Case 3
UpdateProduct()
Case 4
DeleteProduct()
```

```

Case 5
Environment.Exit(0)
Case Else
Console.WriteLine("Invalid choice. Please try again.")
End Select
End While
End Sub
' Method to add a product
Private Sub AddProduct()
Console.WriteLine("Enter product name:")
Dim productName As String = Console.ReadLine()
Console.WriteLine("Enter product price:")
Dim productPrice As Decimal = Convert.ToDecimal(Console.ReadLine())
' Create a new DataSet and connect to the database
Using connection As New SqlConnection(connectionString)
Dim adapter As New SqlDataAdapter()
' Define the SQL command to insert a new product
Dim sql As String = "INSERT INTO Products (ProductName, Price)
VALUES (@ProductName, @Price)"
Dim command As New SqlCommand(sql, connection)
' Add parameters to the command
command.Parameters.AddWithValue("@ProductName", productName)
command.Parameters.AddWithValue("@Price", productPrice)
Try
' Open the connection and execute the command
connection.Open()
command.ExecuteNonQuery()

Console.WriteLine("Product added successfully.")
Catch ex As Exception
Console.WriteLine("Error adding product: " & ex.Message)
End Try
End Using
End Sub
' Method to view all products
Private Sub ViewProducts()
' Create a new DataSet and connect to the database
Using connection As New SqlConnection(connectionString)
Dim adapter As New SqlDataAdapter()
' Define the SQL command to select all products

```

```

Dim sql As String = "SELECT ProductID, ProductName, Price FROM
Products"
Dim command As New SqlCommand(sql, connection)
Try
' Create a new DataSet to hold the data
Dim dataSet As New DataSet()
' Fill the DataSet with the results of the command
adapter.SelectCommand = command
adapter.Fill(dataSet, "Products")
' Display the products
For Each row As DataRow In dataSet.Tables("Products").Rows
Console.WriteLine($"Product ID: {row("ProductID")}, Name:
{row("ProductName")}, Price: {row("Price")}")
Next
Catch ex As Exception
Console.WriteLine("Error viewing products: " & ex.Message)
End Try
End Using
End Sub

' Method to update a product
Private Sub UpdateProduct()
Console.WriteLine("Enter the ID of the product you want to update:")
Dim productId As Integer = Convert.ToInt32(Console.ReadLine())
Console.WriteLine("Enter the new name for the product:")
Dim productName As String = Console.ReadLine()
Console.WriteLine("Enter the new price for the product:")
Dim productPrice As Decimal = Convert.ToDecimal(Console.ReadLine())
' Create a new DataSet and connect to the database
Using connection As New SqlConnection(connectionString)
Dim adapter As New SqlDataAdapter()
' Define the SQL command to update the product
Dim sql As String = "UPDATE Products SET ProductName =
@ProductName, Price = @Price WHERE ProductID = @ProductID"
Dim command As New SqlCommand(sql, connection)
' Add parameters to the command
command.Parameters.AddWithValue("@ProductName", productName)
command.Parameters.AddWithValue("@Price", productPrice)
command.Parameters.AddWithValue("@ProductID", productId)
Try

```

```
' Open the connection and execute the command
connection.Open()
command.ExecuteNonQuery()
Console.WriteLine("Product updated successfully.")
Catch ex As Exception
Console.WriteLine("Error updating product: " & ex.Message)
End Try
End Using
End Sub
' Method to delete a product
```

```
Private Sub DeleteProduct()
Console.WriteLine("Enter the ID of the product you want to delete:")
Dim productId As Integer = Convert.ToInt32(Console.ReadLine())
' Create a new DataSet and connect to the database
Using connection As New SqlConnection(connectionString)
Dim adapter As New SqlDataAdapter()
' Define the SQL command to delete the product
Dim sql As String = "DELETE FROM Products WHERE ProductID =
@ProductID"
Dim command As New SqlCommand(sql, connection)
' Add parameters to the command
command.Parameters.AddWithValue("@ProductID", productId)
Try
' Open the connection and execute the command
connection.Open()
command.ExecuteNonQuery()
Console.WriteLine("Product deleted successfully.")
Catch ex As Exception
Console.WriteLine("Error deleting product: " & ex.Message)
End Try
End Using
End Sub
End Module
```