**SALES APPLICATION - EXAM**

**ITEM.CS**

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Text;

namespace SalesOfItemsApplication

{

public class Item

{

public int ItemId { get; set; }

public string ProductName { get; set; }

public int ProductPrice { get; set; }

public int Quantity { get; set; }

}

}

**----------------------------------------------------------------------------------------------------------------------------**

**ITEMCONTEXT.CS**

using Microsoft.EntityFrameworkCore;

using System;

using System.Collections.Generic;

using System.Text;

namespace SalesOfItemsApplication

{

public class ItemContext : DbContext

{

protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)

{

optionsBuilder.UseSqlServer(@"Data Source=KANINI-LTP-476\SQLSERVER2021ACH;Integrated Security=true;Initial catalog=SalesItemDB");

}

public DbSet<Item> items { get; set; }

protected override void OnModelCreating(ModelBuilder modelBuilder)

{

modelBuilder.Entity<Item>().HasData(new Item()

{

ItemId = 101,

ProductName = "XXX",

ProductPrice = 12,

Quantity = 3

});

modelBuilder.Entity<Item>().HasData(new Item()

{

ItemId = 102,

ProductName = "YYY",

ProductPrice = 15,

Quantity = 0

});

modelBuilder.Entity<Item>().HasData(new Item()

{

ItemId = 103,

ProductName = "ZZZ",

ProductPrice = 20,

Quantity = 2

});

}

}

}

**----------------------------------------------------------------------------------------------------------------------------**

**PROGRAM.CS**

using System;

using System.Collections.Generic;

using System.Linq;

namespace SalesOfItemsApplication

{

class Program

{

ItemContext context;

public Program()

{

context = new ItemContext();

}

public void GetChoice()

{

int iChoice = 0;

try

{

do

{

Console.WriteLine("1. List All Items\n " +

"2. Exit\n");

Console.WriteLine("Please enter your choice");

iChoice = Convert.ToInt32(Console.ReadLine());

switch (iChoice)

{

case 1:

PrintAllItems();

break;

case 2:

Console.WriteLine("Thank you for shopping....Please Come again");

break;

default:

Console.WriteLine("Invalid choice. Please try again");

break;

}

} while (iChoice != 2);

}

catch (Exception e)

{

Console.WriteLine(e.Message);

GetChoice();

}

}

public void PrintAllItems()

{

List<Item> items = context.items.ToList();

foreach (var products in items)

{

Console.WriteLine("The Item details:");

Console.WriteLine(" Item id " + products.ItemId);

Console.WriteLine(" Product name " + products.ProductName);

Console.WriteLine(" Product Price " + products.ProductPrice);

Console.WriteLine(" Product Quantity " + products.Quantity);

Console.WriteLine("---------------------------------");

}

PrintSingleValue();

}

public void PrintSingleValue()

{

Item item1 = new Item();

Console.WriteLine("please enter the Item ID to select a product");

try

{

int itemid1 = Convert.ToInt32(Console.ReadLine());

List<Item> details = context.items.Where(e => e.ItemId == itemid1).ToList();

bool isEmpty = !details.Any();

if (isEmpty)

{

Console.WriteLine("Invalid input please enter valid item id");

}

else

{

foreach (var item in details)

{

Console.WriteLine("You have Selected :");

Console.WriteLine(" Item id " + item.ItemId);

Console.WriteLine(" Product name " + item.ProductName);

Console.WriteLine(" Product Price " + item.ProductPrice);

Console.WriteLine(" Product Quantity " + item.Quantity);

Console.WriteLine("---------------------------------");

Console.WriteLine("Please Enter the quantity");

int quant = Convert.ToInt32(Console.ReadLine());

int res;

if (item.Quantity > 0)

{

res = quant \* item.ProductPrice;

Console.WriteLine($"The total price is : {res}");

item.Quantity = item.Quantity - quant;

context.SaveChanges();

}

else

Console.WriteLine("Product is out of stock!!!sorry");

}

}

}

catch (Exception e)

{

Console.WriteLine(e.Message);

}

}

static void Main(string[] args)

{

Program program = new Program();

program.GetChoice();

Console.ReadKey();

}

}

}

**OUTPUTS:**







