System Requirements Specification Index

For

CricketKit Application (Console)

Version 4.0

IIHT Pvt. Ltd.
IIHT Ltd, No: 15, 2nd Floor, Sri Lakshmi Complex, Off MG Road, Near SBI LHO,
Bangalore, Karnataka – 560001, India
fullstack@iiht.com

CRICKET KIT CONSOLE APPLICATION
System Requirements Specification

1.BUSINESS-REQUIREMENT:

Name of the Course   :   C#

Tool Stack   :   .net core console application with MySQL

Problem Statement   :

Provide a code solution to find out wholesale purchases for Cricket Kit Contribution

Description   :

1. Take console input of the number of Cricket kits to be purchased (n), from the user in the Main method of the program.
2. Take input of details of each cricket kit as string in following format
    purchase id,purchase date, user,item name 1, item cost 1, item quantity 1,item name 2, item cost 2, item quantity 2,.............
    Eg: 3,12-12-2012,Jack,Bat,1500,7,Ball,60,1,Stump,340,5,helmet,925,5,Pad,600,15

3. Create Purchase class with
    a. purchaseId as int
    b. purchaseDate as string
    c. totalAmount as double
    d. userName as string
    e. parameterized constructor.
    f. Override ToString()
    g. Create one static method as below:
        i. static Purchase obtainPurchaseWithAmount(string str) which accepts String Object as input, parse the String to extract information and create a Purchase object from it. Calculate total amount as follows: totalAmount = cost of item1 * quantity1 + cost of item2 * quantity2 + cost of item3 * quantity3 ...
        ii. if the number of items in the input is less than 5, Eg:
            2,12-12-2012,Peter,Bat,1500,3,Ball,60,8,Stump,340,1
            throw InvalidWholeSaleException.
        iii. Here InvalidWholeSaleException is a custom exception, you have to create it.
        iv. If string is not in proper expected format throw ParseException
        v. Here ParseException is custom exception, you have to create it
    h. Create another static method as below:
        i. Static bool StorePurchaseDetails(Purchase purchase) which accepts Purchase Object as input and extracts value from it and stores it into a database.
    i. Create another static method as below:

        i.   static List<Purchase> GetPurchaseDetails() which reads all the records from the database and convert them back into Purchase objects, save them in a List Collection and return the same.

4. Display List of Purchase Objects in main method, with the help of ToString() method of Purchase class.

2. RESOURCES AVAILABLE:

2.1 PACKAGE: CRICKETKIT

Names	Resource	Remarks
Package Structure/Project		
CricketKit	Purchase.cs	This file contains all the business logic and properties related to cricket kit.
	Program.cs	Program.cs is an important class that is the entry point of application.
	InvalidWholeSaleException	This class contains custom exceptions.

2.2 PACKAGE: CRICKETKIT.TESTS

Resources

Note: - Under the CricketKit.Tests contain all test cases for code evaluation, please don't try to alter and edit it.

3. EXECUTION STEPS TO FOLLOW

- **All actions like build, compile, running application, running test cases will be through Command Terminal.**

- **To open the command terminal the test takers need to go to the Application menu (Three horizontal lines at left top) Terminal → New Terminal.**

- **On command prompt, cd into your project folder (cd <Your-Project-folder>).**

- **To build your project use command:**

  (Project_directory/CricketKit / dotnet build)

- **To launch your application, Run the following command to run the application:**
  (Project_directory/CricketKit / dotnet run)

- **This editor Auto Saves the code.**

- **To test any Restful application, the last option on the left panel of IDE, you can find ThunderClient, which is the lightweight equivalent of POSTMAN.**

- **To test any UI based application the second last option on the left panel of IDE, you can find Browser Preview, where you can launch the application.**

- **To run the test cases in CMD, Run the following command to test the application: dotnet test --logger "console;verbosity=detailed" (You can run this command multiple times to identify the test case status,**

  and refactor code to make maximum test cases passed before final submission)

- **If you want to exit(logout) and continue the coding later anytime (using Save & Exit option on Assessment Landing Page) then you need to use CTRL+Shift+B - command compulsorily on code IDE. This will push or save the updated contents in the internal git/repository. Else the code will not be available in the next login.**

- These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.

- You need to use CTRL+Shift+B - command compulsorily on code IDE, before final submission as well. This will push or save the updated contents in the internal git/repository, and will be used to evaluate the code quality.