**Lab 1: Exception Handling**

**Q1. ABCL Corp wants to maintain list of Customers. While accepting the data, you need to validate CreditLimit property. If the value is invalid, you need to raise Exception. We need to implement custom exception class to implement the same.**

**Task 1:** Define a Customer class with following members

CustomerId, Customer Name, Address, City, Phone, CreditLimit

**Task 2:** Define the properties for all these members.

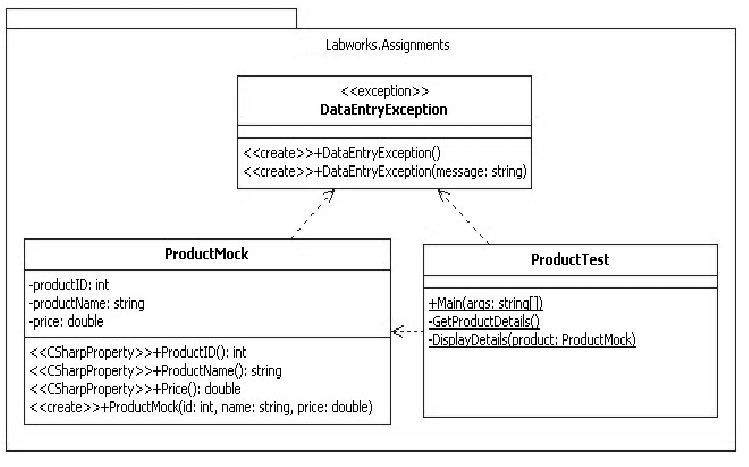
**Task 3:** Define two constructors (Default and Parameterised) to assign the values.

**Task 4:** You need to validate the CreditLimit. If the value is above 50000, then you need to raise Exception to handle this. Create InvalidCreditLimit custom Exception class to achieve the same.

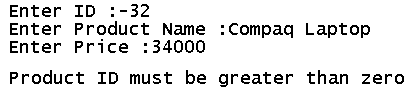
**Task 5:** Use the Exception class created to throw the exception. Ensure that the Client application catches the exception and handles the error properly.

**Q2. Create ProductMock Entity which throws DataEntryException when its properties initialize with following values. Get the Product details from the user and handle in-built and user defined exception. Refer the class diagram given below :**

|  |  |
| --- | --- |
| **Condition** | **Exception Message** |
| productID <=0 | Product ID must be greater than zero |
| productName = = “” | Product Name cannot be left blank |
| price <=0 | Price of product must be greater than zero. |
| productName | Product Name should have alphabets and numbers only |



**Output**



**Lab 2: Using List<> Generic Collection Class**

**Q1. You need to maintain a Contact List in a generic List Collection. You need to perform the following tasks:**

* + 1. AddContact() – To add contact detail to List
    2. DisplayContact() – To display particular contact detail from List
    3. EditContact() – To modiy particular contact detail from List
    4. ShowAllContacts() – To display all contact details from List

**Task 1:** Create the Contact class with the following properties:

public int **ContactNo{get;set;}**

publicstring **ContactName{get;set;}**

publicstring **CellNo{get;set;}**

**Task 2:** Create a Console application and write the Code for the required functionality mentioned above.

**Hint:**

* 1. There is a loop in Main() function which accepts the selection option
  2. There are additional 4 static functions to perform required tasks which are called based on selection
  3. Use the List<Contact> generic collection to maintain the list.

**Q2. Create a console application to accept Product Details like ProductNo, Name, Rate and Stock.**

[Use Array List Collection]

Display the Menu to perform the following:

================

* 1. Adding New Product
  2. Deleting Currently Searched Product
  3. Searching Product

Searching will work as shown below:

* User will enter ProductNo.
* If the product with that productno exists in Collection, then the details should be shown, otherwise show appropriate message.
  1. Save the New Product – The products should get saved in the sorted order of ProductNo.

**Q3. One of the client has submitted the following request. Manish has developed the code for the same. The requirement is given below. You need to review the code, find out any issues / bugs with the code and correct the same. Also you need to develop a Console based Client application for the same requirement. The Client application should allow Adding new Employee’s of specified type, Searching Records, Delete Records and View all records operations.**

**Problem Statement:** XYZ computer Systems PVT Ltd. wants to develop an application to maintain employee details. You have to develop a .NET Application to accept new employee details and store the details in a Collection.

The steps involved in this are given below:

**Task 1:** Create a private DLL with a class Called Employee. Employee class will have Employee Number, Name and Basic Salary, and PF attributes. Define appropriate properties to access the attributes. Write 2 constructors, one default & one parameterized, to assign the values of the attributes when the object is created.

**Task 2:** Use List<Employee> collection.

**Lab 3: Using Dictionary<> Generic collection class**

**Q1. You need to maintain the file extensions along with file types in a dictionary class. Write the code to achieve the same.**

**Tasks to be performed:**

* Create a new dictionary of strings, with string keys.
* Add some elements to the dictionary. There should not be duplicate keys, but some of the values can be duplicates.
* The Add method throws an exception if the new key is already in the dictionary. Test this by adding a duplicate key.
* The indexer can be used to change the value associated with a key. Try changing the value of any record and display the updated value.
* If a key does not exist, setting the indexer for that key adds a new key/value pair. Try this by adding a new value.
* The indexer throws an exception if the requested key is not in the dictionary. Try printing any such key which is not present and handle the exception.
* When you use foreach to enumerate dictionary elements, the elements are retrieved as Key/Value Pair objects. Use a foreach loop to print the values and test this.
* Use the Remove method to remove a key/value pair.

**Lab 4: Using Delegate Concept**

**Q1. You need to perform Arithmetic operations on two numbers. The operations include Add Numbers, Multiply Numbers, Divide Numbers, Subtract Numbers and Find Max Number.**

**Task 1:** Define a class ArithmeticOperation having the above methods.

**Task 2:** Define a Delegate which can call these methods.

**Task 3:** Create a console application to accept two numbers and arithmetic operation to be performed from the user. Based on the choice, the Delegate instance will hold the address of the appropriate method.

**Task 4:** Execute the delegate to get the required result.

Lab 5: **StreamReader / StreamWriter Classes**

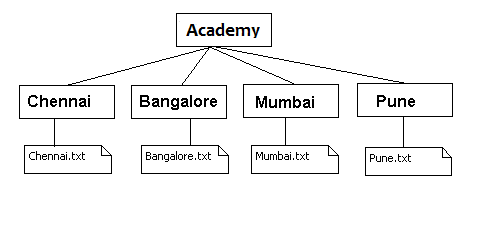
**Q1. Write a Code to Read and Display the contents of a text file. Accept the name of the file from the user. Handle all the exceptions that might occur during reading.**

**Q2. Write a Code to perform File Copy operation. You need to accept the source and destination file names. The data should be copied from source file to destination file.**

**Handle all the exceptions that might occur during the file copy operation.**

**Extended Assignment**

An institute have decided to automate their batch details operations. The application is to be developed in such a manner that a proper directory structure is to be maintained to store the files. The directory structure to be maintained is as shown below:



You need to perform the following operations in C# application

Create a menu based application to store batch details.

* The first option in the menu should allow the user to create a directory structure and the files (if not exists) in the c drive as shown in the above figure.
* The second option should accept the batch details from the user. Based on the location given by the user append the batch details in the respective files.
* The third menu option allows the user to create a backup copy of the Academy folder in D Drive
* The fourth option should allow the user to view the details of the text files "Bangalore.txt", "Chennai.txt", "Mumbai.txt" and "Pune.txt"

**Lab 6: Serialization to persist business data**

**Q1. You have already created the Contact class. You need to store List of Contacts in binary format on disk. Perform Binary Serialization to store the List.**

**To Do:**

Write the Program to Accept the data for multiple contacts, store them in a List. Serialize the List using Binary formatter.

**Q2. The Client has suggested an enhancement to the above code. You need to write the code to Deserialize the data from the Binary file, and print the details.**

**Q3. There is a Change request from the client. Instead of serializing in Binary format, client wants to serialize it using XML format.** Modify the above program to XML serialize the details.