**Assessment Parameters**

* Solution Completeness (all requirements addressed) – 40 %
* Working solution for each requirement – 25 %
* Comments – 10%
* Coding standards – 25 %
* Code with compilation errors will not be considered for assessment.

**Q 1) Doctor Management System**

Create a Doctor Management system which will be used for maintaining the doctor’s details. This system can be distributed to the customers/consumers. This information helps patients to find appropriate doctor based on their ailment(s).

Create a menu driven console application to

* Accept Doctor Information from user and insert the data into a collection.
  + Registration No
  + Doctor Name
  + City
  + Area of Specialization (E.g. - Heart Specialist, Ophthalmologist etc.)
  + Clinic Address
  + Clinic Timings
  + Contact No

|  |
| --- |
| **Validations** |
| All fields are mandatory. |
| Registration No should accept only numbers and should be of 7 digits only. |
| Doctor Name and Area of Specialization should accept only alphabets. |
| Clinic Timings should accept number and alphabets and special character |
| Contact No should accept only numbers and should be of 10 digits only |
| **Exception Handling** |
| Write a custom exception class to handle the exceptions |

**[45 Marks]**

**Note:** The code must use OOP concepts and modularity.

**Marks Distribution:**

|  |  |
| --- | --- |
| Validations | 8 Marks |
| Exception Handling | 5 Marks |
| Data Properly storing in collection | 8 Marks |
| Searching Doctor information based on his Registration No | 8 Marks |
| Coding Standards, Naming Conventions and Commenting | 6 Marks |

Save the data in file using file handling in json format and once we open the application we should be able to see the old data.