CDI

Library Inventory Manager Project

Programming Fundamentals

Bruna Maciel (635-203263)
12-11-2019

Problem Recognition and Initial Definitions

Records must have

- Type (book, magazine, CD,DVD)
- Title
- Artist
- Value
- ISBN [13 digits]
- memberID
- availability

Members must have

- first name
- last name
- phone
- ID
- RecordID

There are a maximum of 30 records to lend at the same time, this includes records on loan and available for loan.

Records and Members can each be a structure.

ISBN can be used as a recordID (13 digits)

Member ID will be provided by the user. To have a pattern, it must have 5 digits between 0-9.

Use regex to validate user's inputs

To store records and member is possible to use arrays. Ask the user at the beginning how many of each will be added OR use lists to have a dynamic quantity of records and members.

Variable	Type	Initial Value
type	Enum or Int	User input
title	String	User input
artist	String	User input
value	Double or Float	User input
ISBN	String	User input
memberID	String	""
avaliability	Int	User input
fistName	String	User input
lastName	String	User input
Phone	String	User input
Id	String	User input
recordID	String	""

Variables with regex validation

- **ISBN**
  Must have exactly 13 digits [0-9]
  Valid input: 9781861972712
- **Phone**
  Must have 10 digits and may or may not have separators as: ( ) or –
  Valid input: 5141231234 or (514) 123 1234 or (514)123-1234
- **Id (members)**
  Must have 5 digits [0-9]
  Valid input: 12345

Variables with validation in loop

The user has 3 attempts to type a valid input, if not return to the main menu.

- **Type**
  Must be an integer between 1 and 4 (1-book, 2-magazine, 3- CD, 4-DVD)
  Valid input: 1,2,3 or 4
- **ISBN**
- Must have exactly 13 digits [0-9]
- Valid input: 9781861972712
- **Availability**
  Must be an integer between 1 and 2 (1-in stock, 2-avaliable). Can only be valued as 3 (3-on loan)
  by the program on the loan record method.
  Valid input: 1 or 2
- **Phone**
  Must have 10 digits and may or may not have separators as: ( ) or –
  Valid input: 5141231234 or (514) 123 1234 or (514)123-1234
- **memberID**
  Must have 5 digits [0-9]
  Valid input: 12345

Exceptions

Variables type, availability and value are not strings, any input different than digits will cause a format
exception and end the program.

Draft

The program in the folder "LibraryInventoryManager-vArrays" is the first version of the project using
arrays to storage records and members.

This program has limited number of records and members that can be add defined by the user at the
beginning. It has fewer validations that can cause errors in the execution (ex: does not validate the input
in availability while adding a new record). Those were adjusted in the delivery project.

Project Delivery

C# **final** project in the folder "LibraryInventoryManager-vList" using lists instead of arrays.

C# **draft** project in the folder "LibraryInventoryManager-vArrays".

Upgrades

- **Validate ISBN number**
  Calculate if the ISBN typed is good.
  https://isbn-information.com/check-digit-for-the-13-digit-isbn.html
- **Adjust exceptions**
  Instead of using the try catch on the main, use on the methods. This way the program can return to the main menu if there is an error instead of ending.
  Messages can be more user friendly
- **Update records or members info**
  Edit a record value or member`s phone number for example
- **Delete records or members**