Assignment 2 Milestone 1 – Database

Dean Zeller, Dr. Krishna Kambhampaty **100 points**
CS201
Fall 2022

Objective The student will implement a Java program to keep track of a collection of items.

Background – Databases

A common concern about advanced computer science students is "they don't have enough database experience!" As instructors, we can push the various implementations of database programming at lower levels, so it won't be so undaunting later in the computer science degree. OOP is well suited for the design of simple databases to keep records of several similar records. For this assignment, you will use a comic book database as a template to create your own database of items. The final submission will be used as a starting point for future milestones. This is not to be confused with the Java `Collection` datatype. Those familiar with the Collections library in Java are free to use it in this assignment, but it is not a requirement.

Requirements

1. Download the files `ComicBook.java`, `ComicBookTester.java`, `ComicCollection.java`, and `ComicCollectionTester.java` from the course website. This is the initial template for keeping track of a comic book collection. Read through these programs, to understand how they work.
2. This assignment will involve modifying the template to be a simple database for a collection of similar items. Before programming, decide on your database item and attributes.
3. The current template keeps track of basic information for a collection of comic books, storing as attributes the title, issue number, writer, artist, condition, and value. After deciding on a topic, choose at least seven attributes appropriate for your topic. While some of your fields may be Strings, have at least two numeric attributes (int or double) to use in calculations.
4. Modify the `ComicBook.java` and `ComicBookTester.java` files to represent a single item in the database. Implement all methods in ComicBook, modified for your selected collection type. The template files have several versions of the constructors, with different parameter schemes. Use at least three different constructors for your object, representing different ways to define the instances.
5. In `ComicBook.java`, there is a method called `createNextInSeries()`. Replace this method with a useful method that is appropriate for your chosen collection type. It should have some specific purpose for the object. Feel free to write additional methods.
6. Modify the `ComicCollection.java` and `ComicCollectionTester.java` files to continue the modifications implementing your selected collection type. Create some members through hardcoded programming in the tester, then display the results. Do not worry about user input at this point.
7. `ComicCollection.java` has the two method stubs `searchByTitle` and `searchByCreator`. At present, these stubs are empty. The intent of `searchByTitle` is to search the entire collection, only displaying comics that match the target title. `searchByCreator` is similar, but searches for the target string in either of the creator-related attributes (writer, artist). You are to modify and implement these stubs to be appropriate for your selected collection type. Use the modified tester file to test all aspects of your collection.
8. Submit your program files to the appropriate marmoset server.
9. Show your work to the instructor in class.

ComicBook.java

```
public class ComicBook
{
    /*************************************************************
    * ATTRIBUTES                                                 *
    *************************************************************/
    private String title;
    private int issueNum;
    private String writer;
    private String artist;
    private String condition;
    private double value;

    /*************************************************************
    * CONSTRUCTORS                                               *
    *************************************************************/

    //all necessary parameters for attributes
    public ComicBook(String title_,
                     int issueNum_,
                     String writer_,
                     String artist_,
                     String condition_,
                     double value_)
    {
        this.title = title_;
        this.issueNum = issueNum_;
        this.writer = writer_;
        this.artist = artist_;
        this.condition = condition_;
        this.value = value_;
    }

    //condition and value omitted
    public ComicBook(String title_,
                     int issueNum_,
                     String writer_,
                     String artist_)
    {
        this(title_,
             issueNum_,
             writer_,
             artist_,
             null,      // condition omitted
             0.00);     // value omitted
    }

    //only series name and issue number
    public ComicBook(String title_, int issueNum_)
    {
        this(title_,
             issueNum_,
             null,      // writer omitted
             null,      // artist omitted
             null,      // condition omitted
             0.00);     // value omitted
    }

    //writer and artist omitted
```

```
public ComicBook(String title_,
                 int issueNum_,
                 String condition_,
                 double value_)
{
    this(title_,
         issueNum_,
         null,       // writer omitted
         null,       // artist omitted
         condition_,
         value_);
}

/*************************************************************
* GET and SET METHODS                                        *
*************************************************************/

public void setTitle(String title_)
{
    this.title = title_;
}
public String getTitle()
{
    return this.title;
}

public void setIssueNume(int issueNum_)
{
    this.issueNum = issueNum_;
}
public int getIssueNum()
{
    return this.issueNum;
}

public void setWriter(String writer_)
{
    this.writer = writer_;
}
public String getWriter()
{
    return this.writer;
}

public void setArtist(String artist_)
{
    this.artist = artist_;
}
public String getArtist()
{
    return this.artist;
}

public void setCondition(String condition_)
{
    this.condition = condition_;
}
public String getCondition()
{
```

```
        return this.condition;
    }

    public void setValue(double value_)
    {
        this.value = value_;
    }
    public double getValue()
    {
        return this.value;
    }

    /*************************************************************
    * OUTPUT METHODS                                             *
    *************************************************************/

    //Return a string with only the title and issue number
    public String description()
    {
        return this.title + " Issue #" + this.issueNum;
    }

    //Return the entire object as a single String
    public String toString()
    {
        String result = this.title+" #" + this.issueNum;
        if (this.writer!=null)
        {
            result += "\n  Written by: "+this.writer;
            result += "\n  Art by: "+this.artist;
        }
        if (this.condition!=null)
        {
            result += "\n  Condition by: "+this.condition;
        }
        if (this.value!=0.00)
        {
            result += "\n  Value by: "+this.value;
        }
        result += "\n";
        return result;
    }

    /*************************************************************
    * OTHER METHODS                                              *
    *************************************************************/
    public ComicBook createNextInSeries()
    {
        return new ComicBook(this.title, this.issueNum+1);
    }

}
```

ComicBookTester.java

```
public class ComicBookTester
{
    public static void main(String[] args)
    {
        // Load initial collection (hardcoded for now)
        ComicBook book1 = new ComicBook("Captain Marvel",
                                        1,
                                        "Kelly Sue DeConnick",
                                        "Scott Hepburn");
        ComicBook book2 = book1.createNextInSeries();
        ComicBook book3 = new ComicBook("Darth Vader",
                                        1,
                                        "Kieron Gillen",
                                        "Salvador Larroca",
                                        "Mint",
                                        3.50);

        // Summary info
        System.out.println("My comic books:");
        System.out.println("  "+book1.description());
        System.out.println("  "+book2.description());
        System.out.println("  "+book3.description());
        System.out.println("--------------------------------------");

        // Full info
        System.out.println("\nDetailed information:\n");
        System.out.println(book1);
        System.out.println(book2);
        System.out.println(book3);

        // Change info
        book1.setWriter("Warren Ellis");
        book2.setArtist("Jim Lee");
        book2.setValue(40);
        book3.setCondition("Mint");
        book3.setValue(book2.getValue());

        // Full info, reflecting changes
        System.out.println("\nDetailed information, with changes:\n");
        System.out.println(book1);
        System.out.println(book2);
        System.out.println(book3);
    }
}
```

ComicCollection.java

```
import java.util.*;

public class ComicCollection
{
    /*************************************************************
    * ATTRIBUTES                                                 *
    *************************************************************/
    private String name;
    private ArrayList<ComicBook> comics;

    /*************************************************************
    * CONSTRUCTORS                                               *
    *************************************************************/
    // all necessary parameters for attributes
    public ComicCollection(String name_)
    {
        this.name = name_;
        this.comics = new ArrayList<ComicBook>();
    }

    // no initial parameters
    public ComicCollection()
    {
        this("My comic book collection");
    }

    /*************************************************************
    * GET and SET METHODS                                        *
    *************************************************************/

    // return a comic given the index
    public ComicBook getComic(int index)
    {
        return this.comics.get(index);
    }

    // return the number of comics in the array
    public int getSize()
    {
        return this.comics.size();
    }

    /*************************************************************
    * OUTPUT METHODS                                             *
    *************************************************************/
    // print the entire collection, with title and summary information
    public void printCollection()
    {
        System.out.println("\nCollection: "+this.name+"\n");
        System.out.println("-----------------------------------------");
        for (int i = 0; i < this.comics.size(); i++)
        {
            System.out.println((i+1)+": "+this.getComic(i).toString());
        }
        System.out.println(this.comics.size()+" in collection.");
        System.out.println("-----------------------------------------");
    }
```

```
    // print a summary of the collection
    public void printSummary()
    {
        for (int i = 0; i < this.comics.size(); i++)
        {
            System.out.println(this.getComic(i).description());
        }
    }

    /************************************************************
    * OTHER METHODS                                             *
    ************************************************************/

    public void addComic(ComicBook newComic)
    {
        this.comics.add(newComic);
    }

    public void SearchByTitle()
    {

    }

    public void SearchByCreator()
    {

    }
}
```

ComicCollectionTester.java

```
public class ComicCollectionTester
{

    public static void main(String[] args)
    {
        // Load initial collection, hardcoded for now
        ComicCollection collection1 = new ComicCollection("Dean's Comics");
        ComicBook tempComic;
        tempComic = new ComicBook("Captain Marvel",
                                  1,
                                  "Kelly Sue DeConnick",
                                  "Scott Hepburn");
        collection1.addComic(tempComic);

        tempComic = new ComicBook("Spider-Man",
                                  1,
                                  "Brian Michael Bendis",
                                  "Mark Bagley",
                                  "Mint",
                                  25.00);
        collection1.addComic(tempComic);
        collection1.printCollection();

        // Make changes through get and set methods
        collection1.getComic(0).setTitle("Superbwoman");
        collection1.getComic(1).setArtist("Gary Frank");
        collection1.printCollection();

    }
}
```