

Object Oriented Programming Techniques for Assignment 4

Section 1:

- Interface
 - Database.java implements the Library.java interface
- Encapsulation
 - Database.java → getLibraryFile()
- Attribution
 - Database.java → private double libraryLines;
 - libraryLines is determined by getBookNewlineCount()
- Class/Objects (and Constructor)
 - Database.java → Database d = new Database(String database);
 - String database parameter is determined by Scanner method nextLine()
- Overriding
 - Database.java → getBookOccurrences(int b, String s) overrides Library.java → getBookOccurrences(int b, String s)
- Scope
 - Database.java → getBookNumLines(int b) → Variable Scanner sc
 - sc is visible throughout the entire getBookNumLines(int b) method.
 - Because sc is strictly in the getBookNumLines(int b) method, the variable Scanner sc can be used in other methods
- Recursion
 - Database.java → getBookNewlinePercentage(char[] n, int b) → indexOf(String s)
 - Uses recursion in regards to indexOf() method. Searches for an occurrence of the given character. After a character is found, it will search again from the next index until the end of the string has been reached (while loop)
- Local Variables
 - Database.java → getLibraryNewlinePercentage(char[] c)
 - int countAll
- Accessors and Mutators
 - Accessor: Database.java → getLibraryFile();
 - Mutator: Database.java → setLibraryFile(String l)

Section 2:

- setLibraryFile(String l)
 - Accessor method
 - Encapsulation
 - Overriding
 - Interface
 - Attribution
 - fileName

- `getLibraryFile()`
 - Getter method
 - Encapsulation
 - Interface
- `getLibraryBooks(boolean p)`
 - Encapsulation
 - Interface
 - Getter method
 - Attribution → `books = arrayOfBooksInLibrary`
 - Local variables → `counter, sc, setArray`
- `getNumberOfBooks()`
 - Encapsulation
 - Interface
 - Getter method
 - Override
 - Local variables → `sc, counter`
 - Uses database attribute
- `getLibraryOccurrences(String s)`
 - Encapsulation
 - Interface / Override
 - Getter method
 - Uses attribute `arrayOfBooksInLibrary`
 - Calls method `getBookOccurrences(int i, String s)`
- `getLibraryNewlinePercentage(char[] c)`
 - Encapsulation
 - Interface / Override
 - Getter method
 - Local variables → `countAll`
 - Attribute → `libraryLines`
 - Calls method `getBookNewlinePercentage()`
 - Recursion with `indexOf(String s)` method
- `getBookOccurrences(int b)`
 - Calls `getContentsInBook(int i)` → returns string of file contents
 - Calls `phrasesInString(String s, String p)`
 - Encapsulation
 - Interface / Override
 - Getter method
- `getBookNumLines(int b)`
 - Encapsulation
 - Interface / Override
 - Getter method
 - Local variables → `sc, countLine, line`

- getBookNewlineCharacters(int b)
 - Encapsulation
 - Interface / Override
 - Getter method
 - Local variables → fr, br, charArray

Section 3: Rest of Section 3 on Scan

<div><<Interface>></div> <div>Library</div>
<div>+setLibraryFile(libraryFileName: String) : Boolean</div> <div>+getLibraryFile(): File</div> <div>+ getLibraryBooks(withPath: Boolean) : String[]</div> <div>+getNumberOfBooks() : int</div> <div>+getLibraryOccurrences(phrase: String) : int</div> <div>+getLibraryNewlinePercentage(newlineChars: char[]) double</div> <div>+getBookOccurrences(bookNumber:int, phrase:String) : int</div> <div>+getBookNumLines(bookNumber : int) : int</div> <div>+getBookNewlineCharacters(bookNumber:int) : char[]</div>