# Class Library

java.lang.Object
    Library

---

```
public class Library
extends Object
```

This is the Library class to that is responsible for using other classes and running the project with a set of static methods.

## *Constructor Summary*

### Constructors

| Constructor | Description |
|---|---|
| `Library()` | |

## *Method Summary*

**All Methods** Instance Methods Concrete Methods

| Modifier and Type | Method | Description |
|---|---|---|
| void | `addIt(Scanner sc)` | This method add a new item object into the array list and update the test file. |
| void | `addItems(String fileName)` | Helper method enable loading things from a given text file. |
| Item | `addItemsFromFile(String type, Scanner input)` | Adds Items loaded from a given text file to the list of items in the store. |
| void | `addLibrarian(Scanner sc)` | This method add a new librarian into the array list and update the test file. |
| Librarian | `addLibrariansFromFile(Scanner input)` | Adds Librarians loaded from a given text file to the list of items in the store. |
| void | `borrowStuff(Scanner sc)` | This method check to see if the user can borrow this item. |
| void | `check(Scanner sc)` | This method check to see if this item is available or unavailable to borrow. |
| void | `createItem(String type, String type2, int id, String title,` | This method helps create an object based on the information provided. |

| | | |
|---|---|---|
| | boolean borrowed, String author, int number) | |
| void | printSummary() | Interface method to print a summary of the librarians in the library. |
| void | removeIt(Scanner sc) | This method remove a new item into the array list and update the test file. |
| void | removeLibrarian(Scanner sc) | This method remove a new librarian into the array list and update the test file. |
| void | returnStuff(Scanner sc) | This method check to see if the user can return this item. |
| Item | searchIt(int id) | This method use the HashMap to find the Item object. |
| boolean | searchItem(int id, boolean flag) | This method use the HashMap to find the item object. |
| Librarian | searchLibra(int id) | This method use the HashMap to find the librarian object. |
| boolean | searchLibrarian(int id, boolean flag) | This method use the HashMap to find the librarian object. |
| void | workWithFile() | This method works with PrintWriter and test file to update the information when it is called. |

### Methods inherited from class java.lang.Object

equals , getClass , hashCode , notify , notifyAll , toString , wait , wait , wait

## *Constructor Details*

### Library

```
public Library()
```

## *Method Details*

### addItems

```
public void addItems(String  fileName)
              throws FileNotFoundException
```

Helper method enable loading things from a given text file. This method is called from the main.

**Parameters:**

`fileName` - The name of the text file to the path to text file.

**Throws:**

`FileNotFoundException` - This one is thrown.

## addLibrariansFromFile

```
public Librarian addLibrariansFromFile(Scanner  input)
```

Adds Librarians loaded from a given text file to the list of items in the store. The items are stored line-by-line in the text file. Each line contains values separated by a tab character.

**Parameters:**

`input` - The scanner to scan the librarians.

**Returns:**

Librarian The librarian object.

## addItemsFromFile

```
public Item addItemsFromFile(String  type,
                             Scanner  input)
```

Adds Items loaded from a given text file to the list of items in the store. The items are stored line-by-line in the text file. Each line contains values separated by a tab character.

**Parameters:**

`type` - The type of the item.

`input` - The scanner to scan the librarians.

**Returns:**

Item The item object.

## printSummary

```
public void printSummary()
```

Interface method to print a summary of the librarians in the library.

## addLibrarian

```
public void addLibrarian(Scanner  sc)
                  throws FileNotFoundException
```

This method add a new librarian into the array list and update the test file.

**Parameters:**

`sc` - The scanner sc to scan the ID.

**Throws:**

`FileNotFoundException` - This one is thrown.

## searchLibrarian

```
public boolean searchLibrarian(int id,
                               boolean flag)
```

This method use the HashMap to find the librarian object. A HashMap is a data structure in computer programming that provides a way to store and retrieve data using a key-value pair mapping.

**Parameters:**

`id` - The id that the user inputs in.

`flag` - The boolean to check if this searching is for add librarian or remove librarian. If it is to add, the flag is true, and false for the remove librarian.

**Returns:**

boolean value This one to check if the object is in the array list.

## addIt

```
public void addIt(Scanner  sc)
           throws FileNotFoundException
```

This method add a new item object into the array list and update the test file.

**Parameters:**

`sc` - The scanner to scan the things user input.

**Throws:**

`FileNotFoundException` - This one is thrown.

## createItem

```
public void createItem(String  type,
                       String  type2,
                       int id,
                       String  title,
                       boolean borrowed,
                       String  author,
                       int number)
```

This method helps create an object based on the information provided.

**Parameters:**

`type` - The object of the item.

`type2` - The type of the item.

`id` - The id of the item.

`title` - The title of the item.

`borrowed` - The flag to see if this item is borrowed or not.

`author` - The author of the item.

`number` - The number of the item.

## searchItem

```
public boolean searchItem(int id,
                          boolean flag)
```

This method use the HashMap to find the item object. A HashMap is a data structure in computer programming that provides a way to store and retrieve data using a key-value pair mapping.

**Parameters:**

`id` - The id that the user inputs in.

`flag` - The boolean to check if this searching is for add librarian or remove librarian. If it is to add, the flag is true, and false for the remove librarian.

**Returns:**

boolean value This one to check if the object is in the array list.

## searchLibra

```
public Librarian searchLibra(int id)
```

This method use the HashMap to find the librarian object. A HashMap is a data structure in computer programming that provides a way to store and retrieve data using a key-value pair mapping.

**Parameters:**

`id` - The id that the user inputs in.

**Returns:**

Librarian The librarian object that the user needs to find

### removeLibrarian

```
public void removeLibrarian(Scanner  sc)
                    throws FileNotFoundException
```

This method remove a new librarian into the array list and update the test file.

**Parameters:**

`sc` - The scanner to scan the input.

**Throws:**

`FileNotFoundException` - This exception is thrown.

### searchIt

```
public Item searchIt(int id)
```

This method use the HashMap to find the Item object. A HashMap is a data structure in computer programming that provides a way to store and retrieve data using a key-value pair mapping.

**Parameters:**

`id` - The id that the user inputs in.

**Returns:**

Item The Item object that the user needs to find

### removeIt

```
public void removeIt(Scanner  sc)
             throws FileNotFoundException
```

This method remove a new item into the array list and update the test file.

**Parameters:**

`sc` - The scanner to scan the input.

**Throws:**

`FileNotFoundException` - This exception is thrown.

### borrowStuff

```
public void borrowStuff(Scanner  sc)
                throws FileNotFoundException
```

This method check to see if the user can borrow this item. If it is not, it will print out the isUnavailable() method in the interface. If they can borrow, it will update the information in the test file.

**Parameters:**

`sc` - The scanner sc to scan the input of the ID.

**Throws:**

`FileNotFoundException` - This one is thrown.

### returnStuff

```
public void returnStuff(Scanner  sc)
                 throws FileNotFoundException
```

This method check to see if the user can return this item. If it is not, it will print out the isAvailable() method in the interface. If they can return, it will update the information in the test file.

**Parameters:**

`sc` - The scanner to scan the input of the ID.

**Throws:**

`FileNotFoundException` - This one is thrown.

### check

```
public void check(Scanner  sc)
```

This method check to see if this item is available or unavailable to borrow.

**Parameters:**

`sc` - The scanner sc to scan the input of the ID.

### workWithFile

```
public void workWithFile()
                  throws FileNotFoundException
```

This method works with PrintWriter and test file to update the information when it is called.

**Throws:**

`FileNotFoundException` - This one throws.