**Java Front-End Program Proposal**

**CPG107P**

***Performance-Based Task #2***

**Section: I1201**

**Members:**

* **Olayvar, Hannali (programmer)**
* **Gelena, Elizabeth (programmer)**
* **Natividad, Chloe**
* **Cruz, Euone**

**A.** **Proposed Program:**

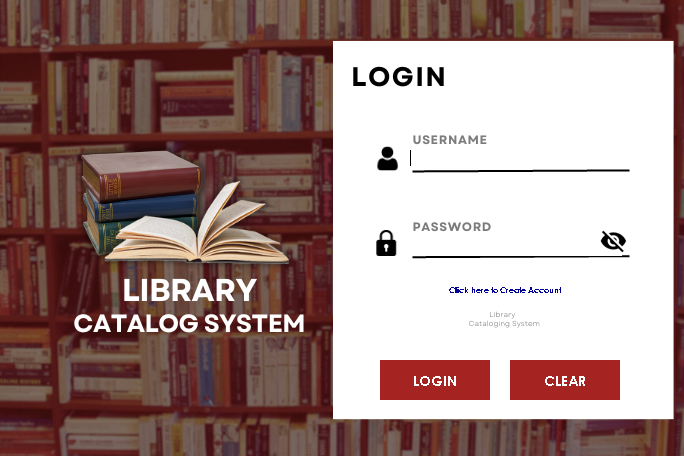
**Library Cataloging System with Checkout System**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_frmlogin\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Functions:*

1. *For logging in*
2. *Redirecting to frmsignup if there is no account through a button*
3. *If there is an account, will proceed to main gui*

The first thing the user sees upon running the program is this: frmlogin



The main Java components are:

* + textfield for the username
  + Passwordfield for password
  + Show password button
  + Create Account button (leads to frm signup)
  + Login button
  + Clear Button (Clears all textboxes)

**Login Button**  is clicked:

> Checks if the inputted username and password is null,

if yes it will show a notification toA **“Fill all Fields”,**

If no, proceeds to next step

> Check If the inputted username and password exists in the Database.

If yes, Store username string to GV.MAINUSERNAME

Store password string to GV.MAINPASSWORD (for later use)

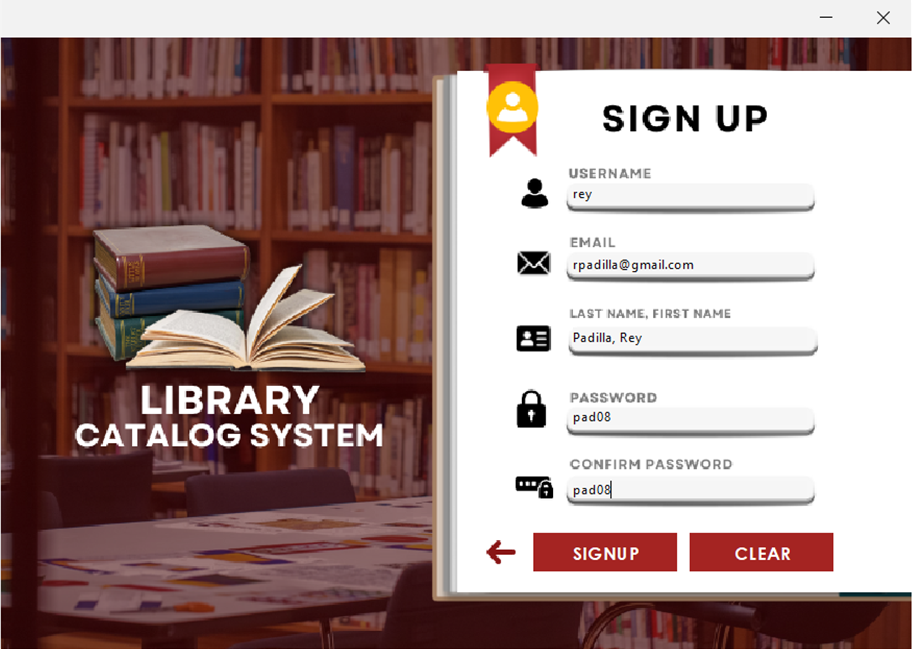
Proceeds to **frmmain.**

If no, Notification will show up “**Invalid Username or Password**”

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_frmSignUp\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Functions:*

1. *For creating an account*



Several textfields which must be filled

**Sign up button  Clicked:**

> Check if all textfields are filled

if no, A notification will pop up **“Answer all fields”**.

If yes, Proceeds to next step

>Create an Account

Stores all inputted data in **tbl\_accounts** of the lms database.

After account creation, auto Redirect Back to **frmlogin** and proceed to login

**Back button Clicked:**

>Returns Back to frmlogin, in case you want to ot create an account (will clear all fields of both forms.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_frmmain\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Functions:*

1. *Different tabs to navigate with.*



< panel header

pnl >

bttns

< panel 3

**Panel Button - Button functionalities:**

>If Hovered;

Chanes bg color

> If Clicked will do the following:

(Clicked button will set its bgcolor to a different one and will stay as that color

unless another button is clicked, like home button’s state right now)

Home Button - proceeds to show pnlhome in panel 3

Update Button - show pnlupdate in panel 3

Recods Button - show pnlrecords in panel 3

Account Button - show pnlaccount in panel 3

**Panel header design explanation:**

> A image set in a jlabel icon

> Made using canva

> Pnl header stays on the top part of frm main

**Panel 3 explanation:**

> place holder for the 4 different panels (if those pnlaccout, pnlhome, pnlupdate, pnlrecords is called, it will show only inside panel 3)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_pnlHome\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Functions:*

1. *Display program overview*

****

**pnlHome Design explanation:**

>Made using canva

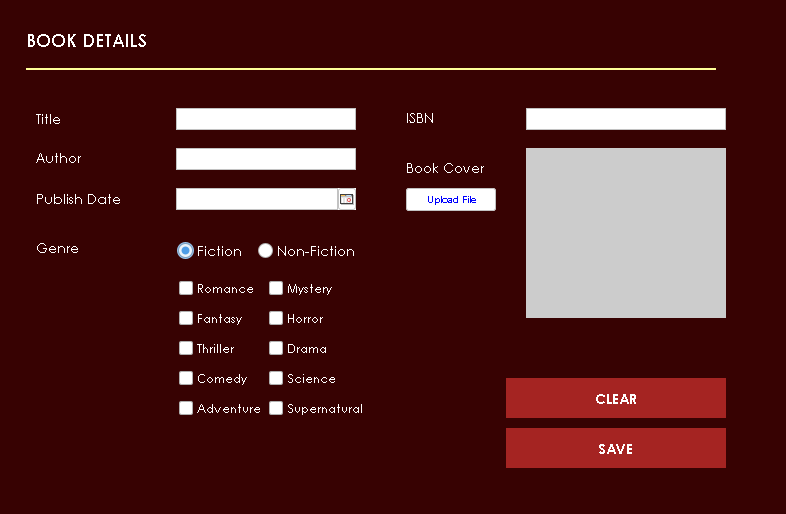
>Image in a jlabel’s Icon property

>this is the default pnl showed if frmmain is opened.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_pnlUpdate\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Functions:*

1. *Add a book in the database*
2. *All fields have null checks*

**

**Upload File Button Clicked;**

>Opens up a **file dialog** and filters to show only folders, image files.

>Select a file and it will show up in the panel beside it. —--------------->>

>Will force scale the image into a book cover size.

**Genre Radio Button is Selected:**

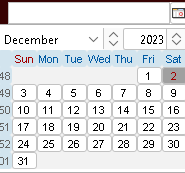
> Only one can be selected

> In a button Group

|  |  |
| --- | --- |
| >if **fiction** is clicked, these buttons will show up | >if **non fiction** is clicked, these buttons will show up |

**Date Chooser  in publish date is clicked; (Added a JDateChooser Library)**

> date chooser opens up, select year, month and date.



> Selected date wil be in text format in the text field of the publish date.

**Save Button  clicked:**

> Check if all fields are filled (except for image);

If yes, proceed to nextstep

If no, Notification shows up “**Fill All Fields**”

> Check if there is image uplopaded. Complicated, code too long.

If yes, Convert Image uploaded to Byte array. Proceed to next step

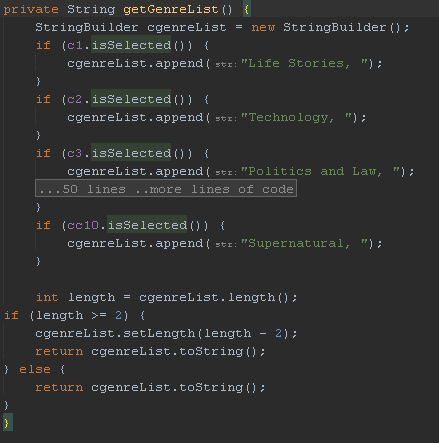
If no, upload the default file cover (Download from an url into the program) then convert it to byte array too. Proceed to next step.



> Get All data imputed and store it in the **database, tbl\_books.**

**Process of storing genre list:**

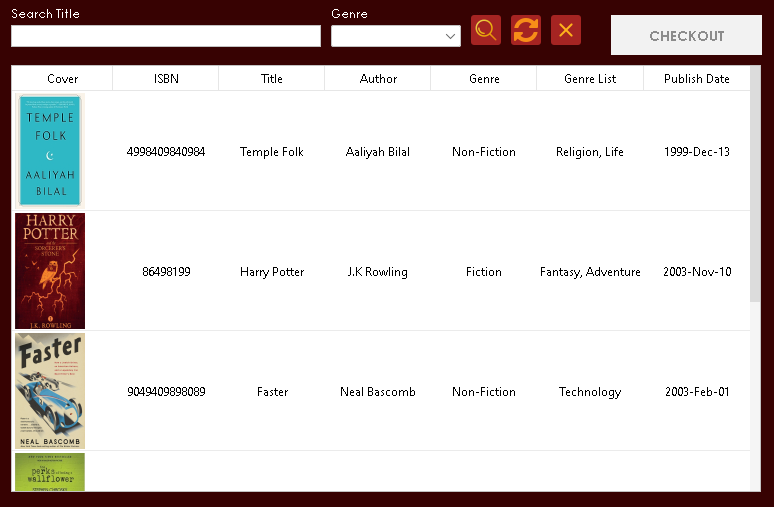
> Get string of selected checkbox, use string builder. Sample code:



**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_pnlRecords\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Functions:*

1. *Filter through book database*
2. *Display books in database in a table*
3. *Proceed to checkout*



**When panel records is shown it :**

>displays books in table from the database.

**Image Conversion Process: (code is very long)**

> Blob image is converted to byte array

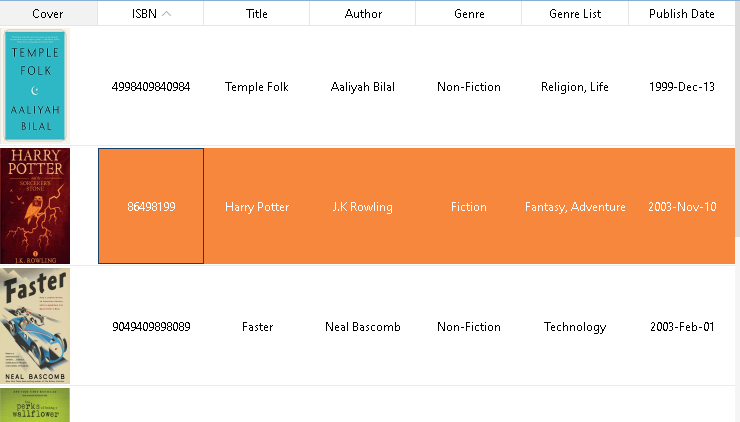
> Byte array is copnverted into an image

> Image will be shown in the table.

**Search filter button  is clicked:**

> Filter by title if there is text in the title txtfld

> Filter by Genre if something’s selected in the genre cmb.

> If both are null, no filter will be applied.

**Refresh Button  is clicked:**

> Refresh table. SQL query will be applied

**Clear Button  is clicked;**

> Clear all filter textboxes, and comboboxes

**If a table rows will be selected;**

> Background color will change to orange—--------------------------------------------^

> Click once again to deselect

> Multiple selections are allowed

> **Button Checkout** will be enabled. If clicked, proceed to open **frmcheckout**

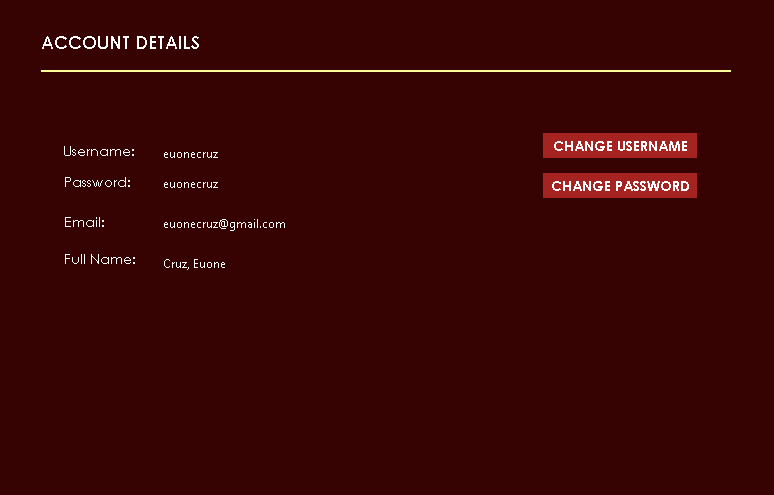
**If tbl header  is clicked:**

> row order will be changed, sorted ascending or descending according to what column header is clicked. Ex. Sort table by author’s by Alphabetical order.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_pnlAccounts\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Functions:*

1. *Change username*
2. *Change Password*

****

**If pnlAccounts is shown it does the ff:**

> get the username from GV. MAINUSERNAME and password from GV.MAINPASSWORD.

> Read in the database tbl\_accounts, which row has a username and password that match the strings from the first step.

> Get the row data from dbase and display in the respective textlabels in the gui.

|  |  |
| --- | --- |
| **If Change username bttn is clicked:**  > this panel shows | **If Change password bttn is clicked:**  > this panel shows |

**In the panel if the change username/password button is clicked:**

> Check if textfields are null

If yes, show notification **“Fill all Fields”**

If no, proceed to next step

> Check if the strings from textfield1 is the same as textfield2

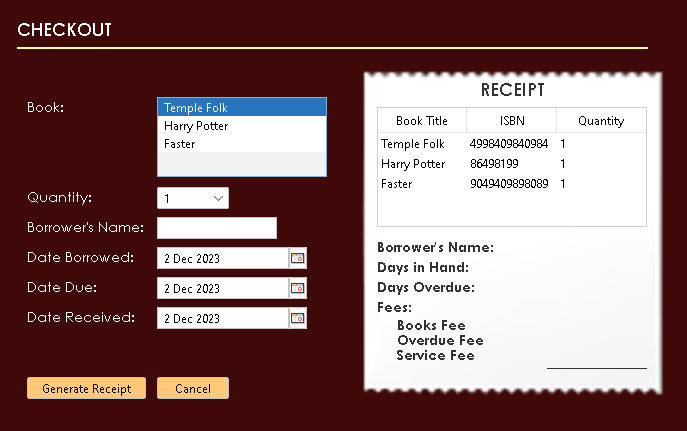
If yes, change username/password in the database itself.

If no, show notification “**Input does not match**”

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_frmcheckout\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Functions:*

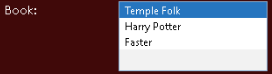
1. *Checkout books selected in the table*
2. *Image receipt generation*

**

**The selected books in pnlrecords will show in this listbox Automatically.**

> The selected rows, its title will be stored in an array

> Those values in the array will be displayed in this listbox:



**You can change the quantity per book.**

> Just select one book in the list above and edit its quantity using the quantity combobox:

**If generate Reciept btn is clicked:** :

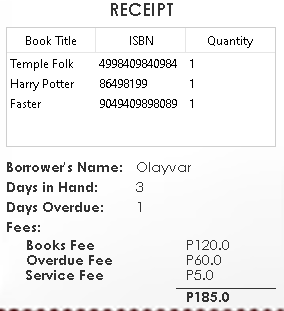
> Check if all fields are answered

If Yes, proceed to next step

If No, Show notif “**Fill all Fields**”

>Validate date

Will go to next step if: Date Borrow < Date Due && Date Recieve > Date Borrow

If not, show notif “**Invalid Date”**

>Calculations are made

Book fee is 40 per piece

Overdue fee is 20 per day overdue

Sevice fee is flat 5 pesos

>Show inputted data in the receipt —-------------------->

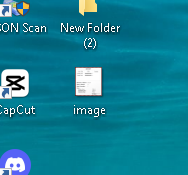
>Enable **Download Button**

**If download button is enabled then clicked:**

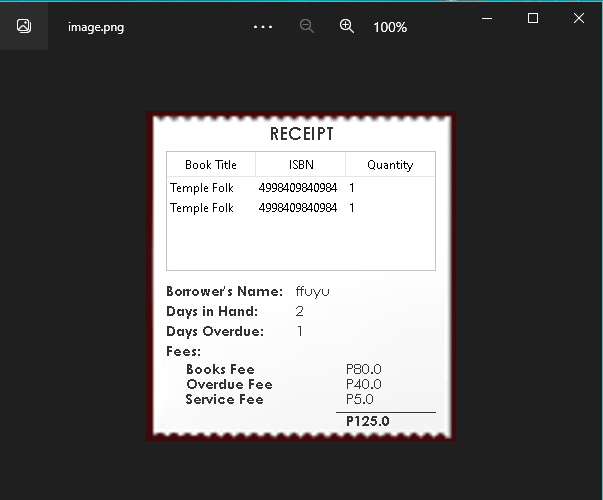
> An image of the receipt will be saved to your local files

> You can choose the directory and file name as the file dialog allows you to.

> Sample saved file in my desktop:



>once opened in photos this is the result:



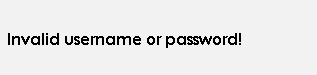
**If cancel button  is clicked:**

> Closes frm checkout and gets back to frmmain, pnlrecords.

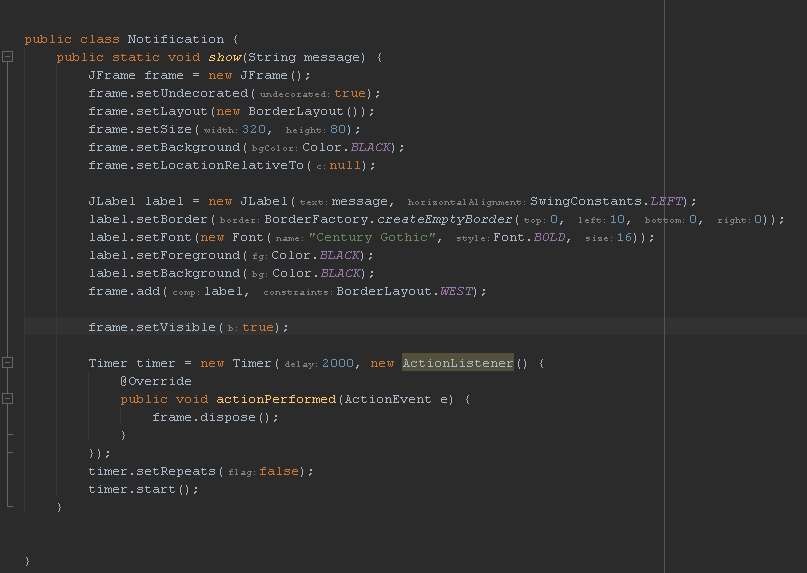
**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Notification Class\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Functions:*

1. *Show a jframe with timer*
2. *It closes automatically after 2 seconds*

**Sample Interface:**

**Code used:**



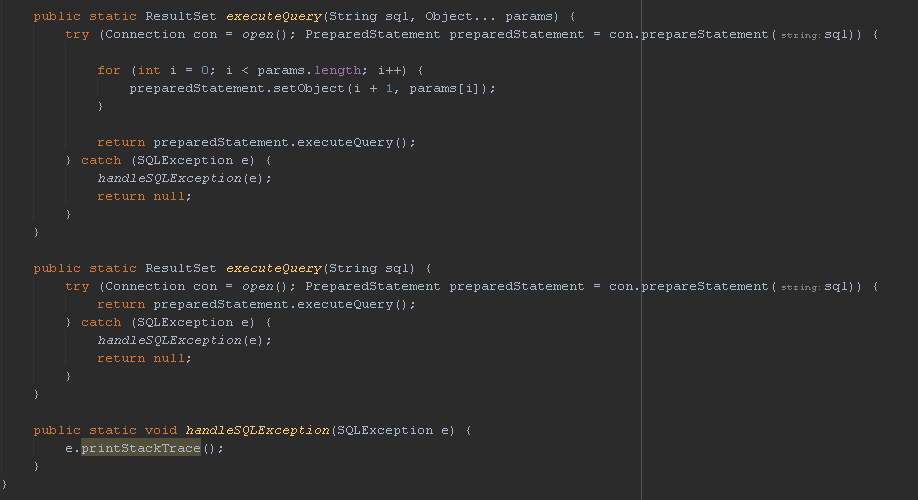
**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_DB Class\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Functions:*

1. *Utility Class*
2. *For easier Database connection*

**Code:**



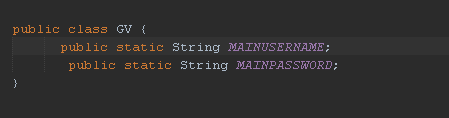


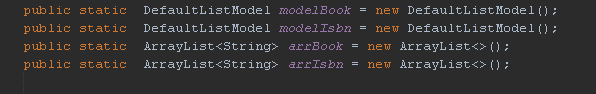
**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_GV CLASS\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Functions:*

1. *Variables that can be accessed anywhere*
2. *Stored in frmlogin and used for frmaccount*

**Code:**

****

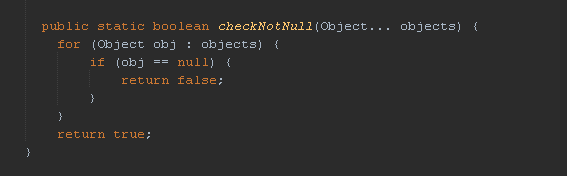
****

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_REUSABLES CLASS\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*Functions:*

1. *Methods that are used in mostly all code*
2. *checkNotNull (for null checks)*

**Code:**

**