Mobile Application Development

(Assignment 2)

**team members:**

17309003 Cho Manki

17309004 Kim Seonghyeon

17309005 Kum Seungwon

17309006 Min Siwan

**Technical features:**

for making "login-in for user authentication"(assignment2-1),"sign-up form to allow user to key in structured information "( assignment2-2),"posting the form infromation to back end with http request"( assignment2-4),"retrieving data from backend by calling a http request service"( assignment2-5), we used the php language, mysql, and the Apache server calling AMP

for making database using mysql , we used this sql statement,

// create database

CREATE DATABASE `member\_db` CHARACTER SET utf8 COLLATE utf8\_general\_ci;

// create table

CREATE TABLE `USER`(

userName VARCHAR(20) NOT NULL,

userPassword VARCHAR(20) NOT NULL,

userEmail VARCHAR(30) NOT NULL,

PRIMARY KEY(userName))

DEFAULT CHARSET=utf8;



we need to add this code to our Androidmanifest.xml document for communication with Android and the server.

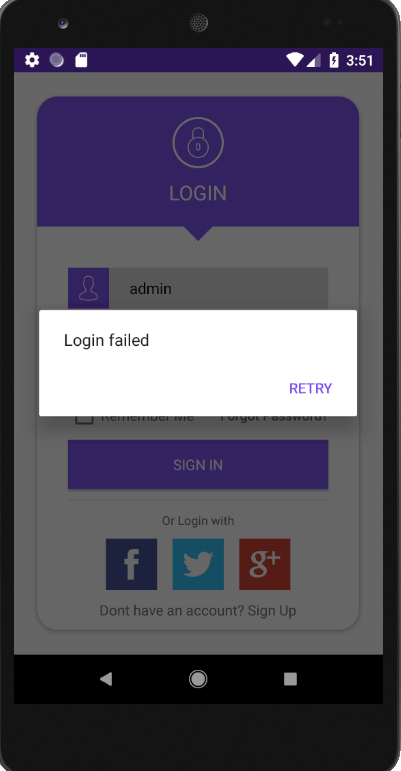
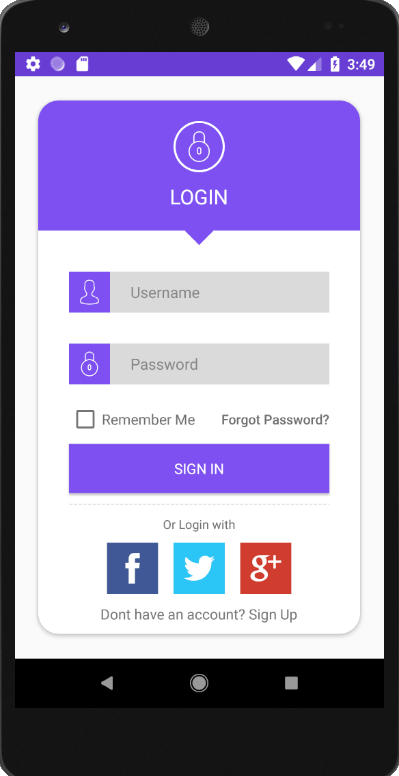


and also need this api in build.gradle dependencies ,,

API : volley to can communication with android and server

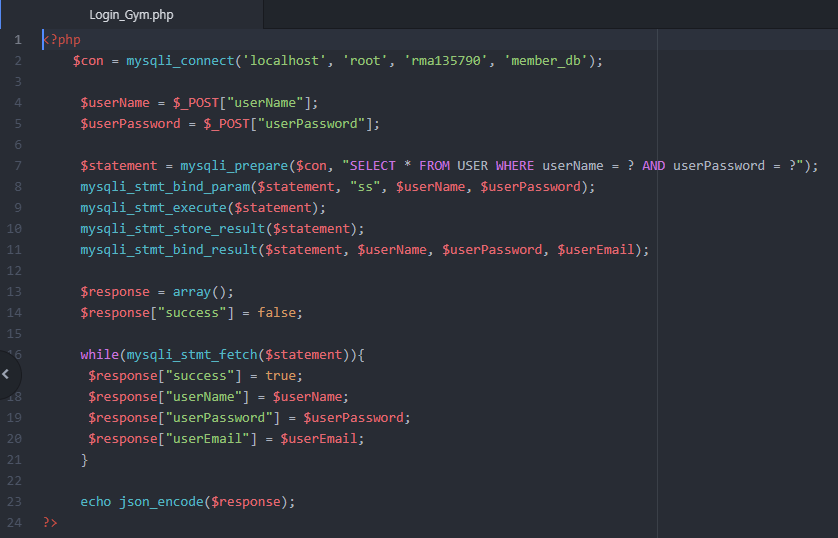


**1.Login-in for user authentication**

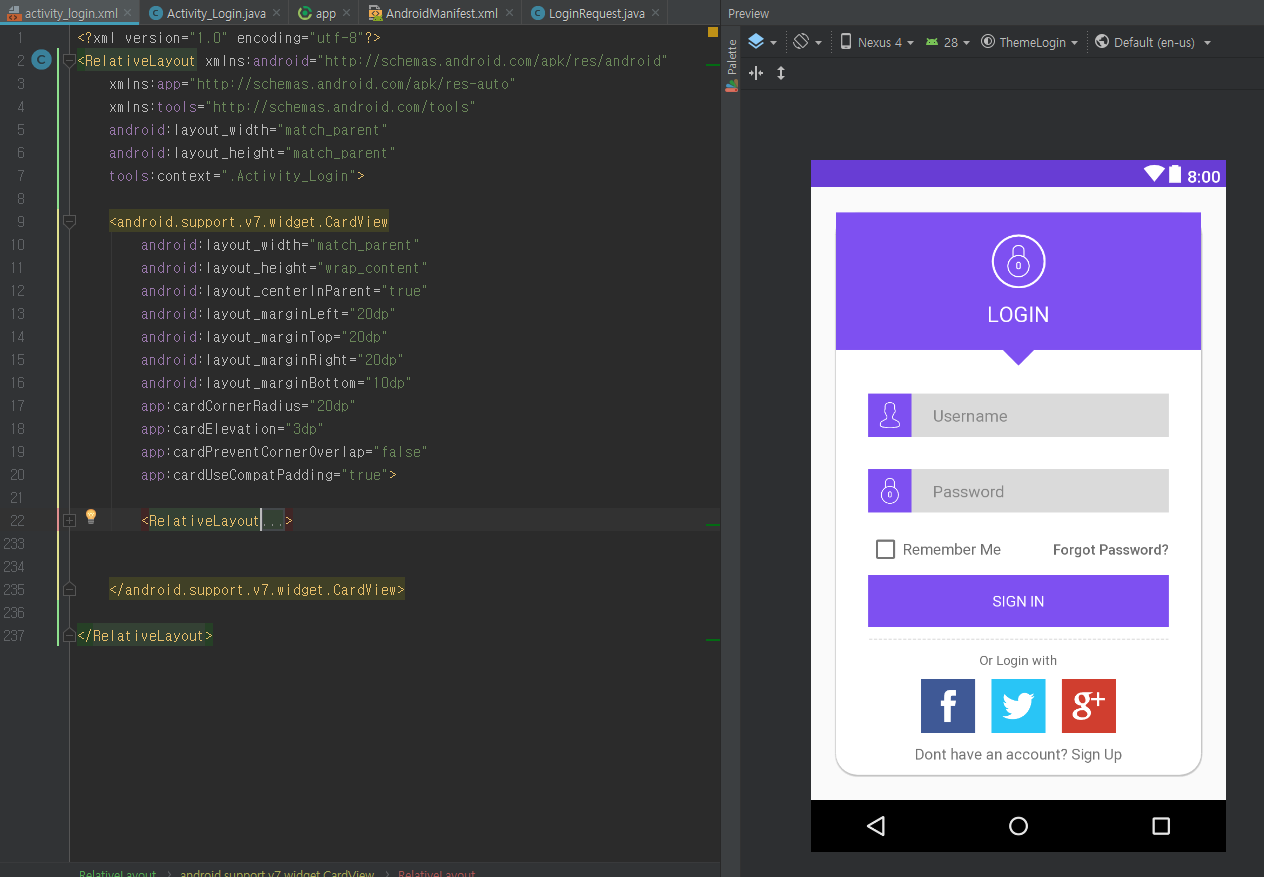


**Required API,php,mysql,XML and JAVA Codes :**

**php**: login\_Gym.php for seaching member database to login

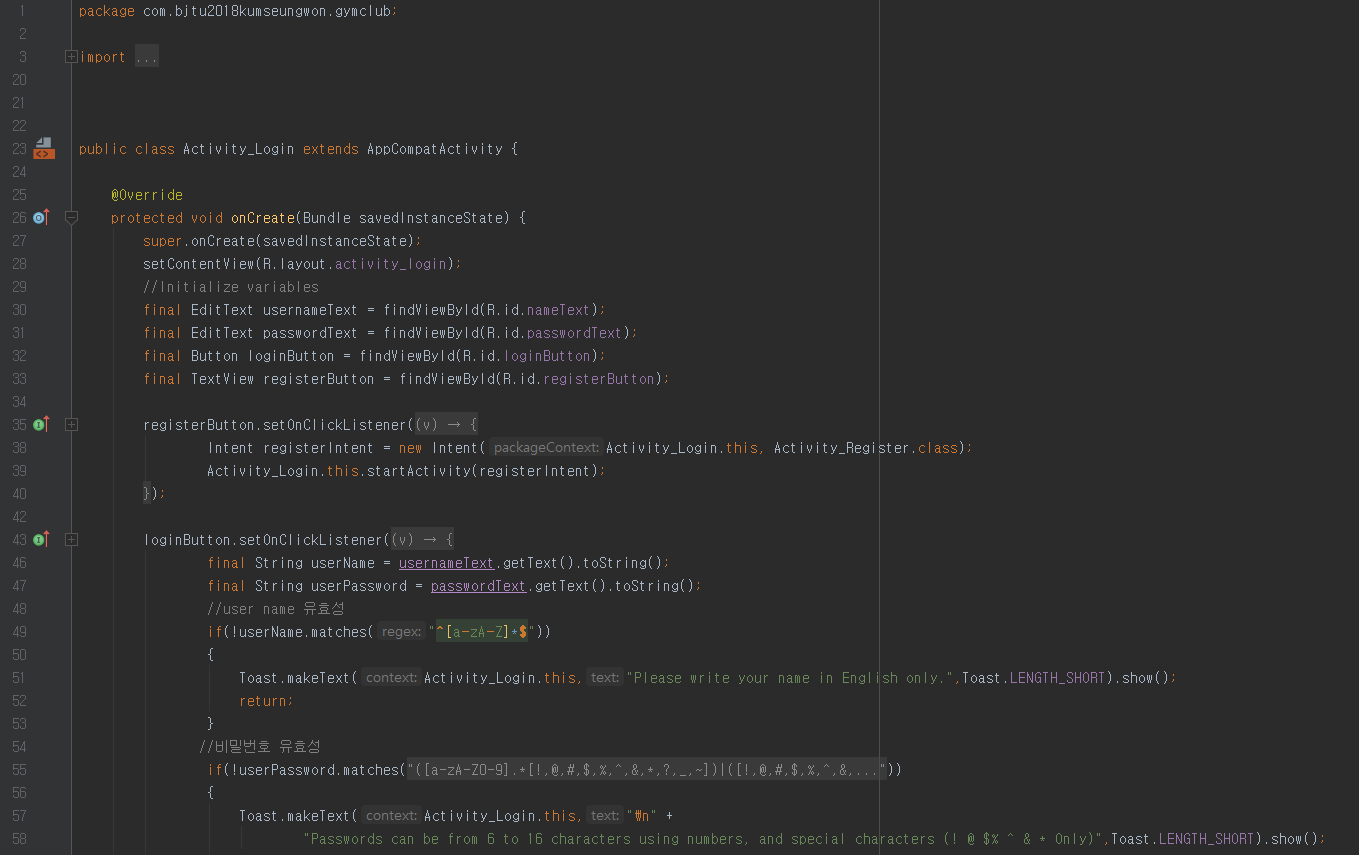


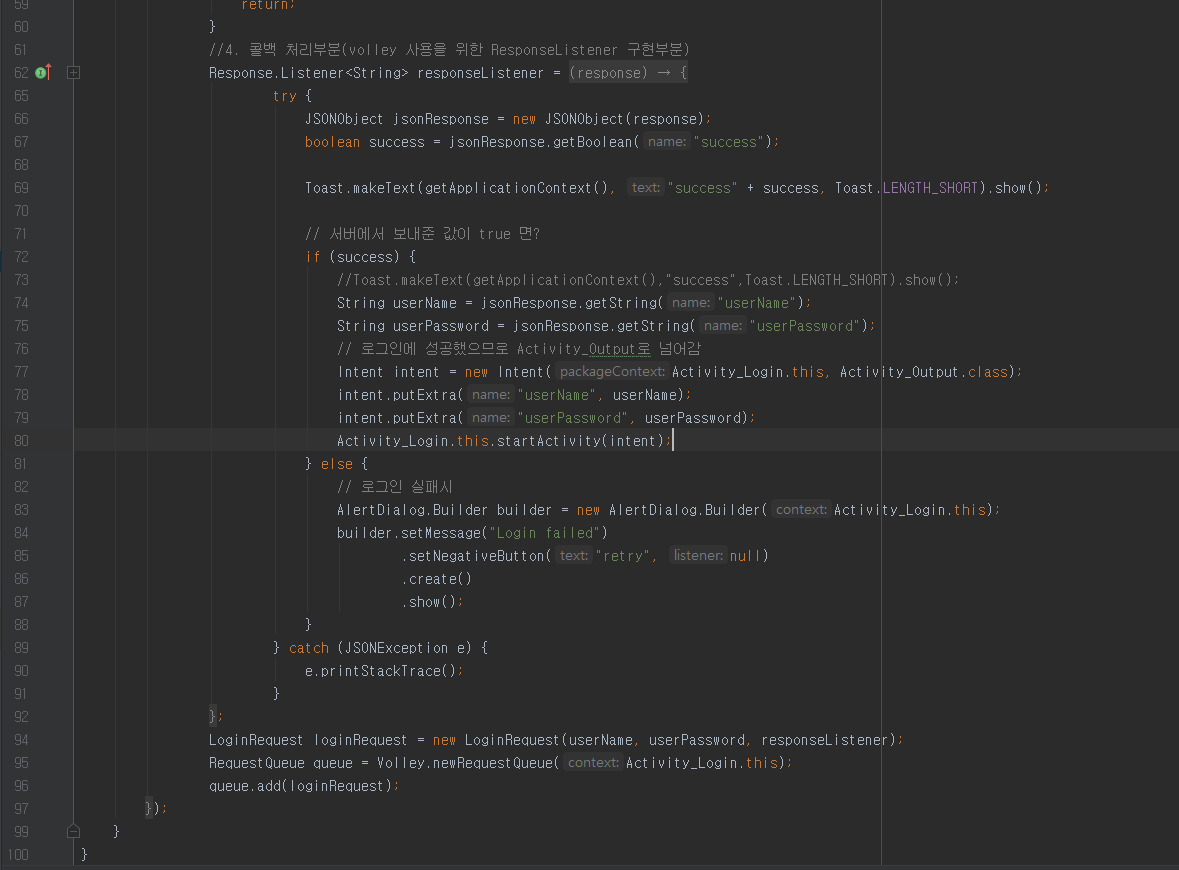
**xml**: activity\_login.xml( Same as the code used in assignment1)



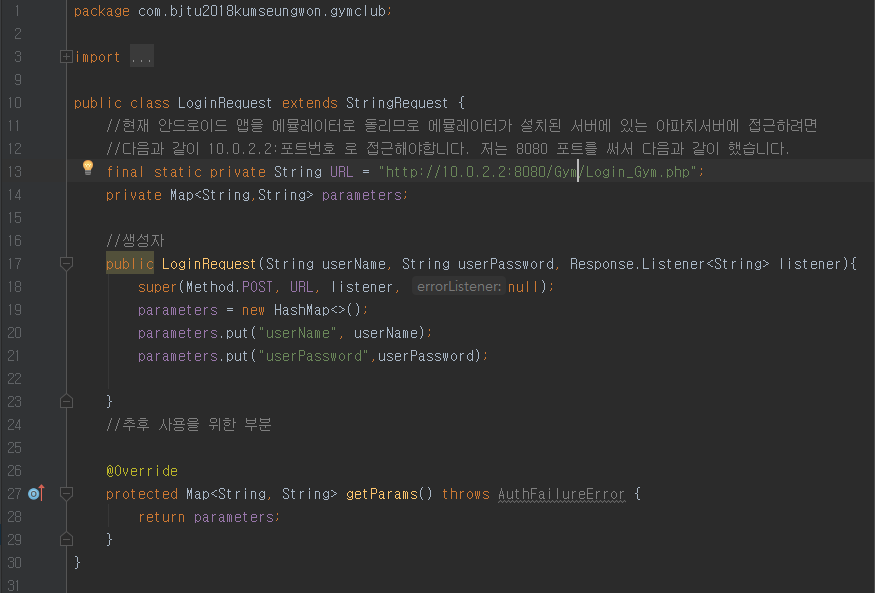
**Java code** : Activity\_Login, LoginRequest

In "Activity\_Login.java", we receive the result from the server in "LoginRequest.java", and if we succeed in login, we start "intent" with "Activity\_Output.java". If it did not succeed, I popped up a dialog to let me know that it did not succeed.

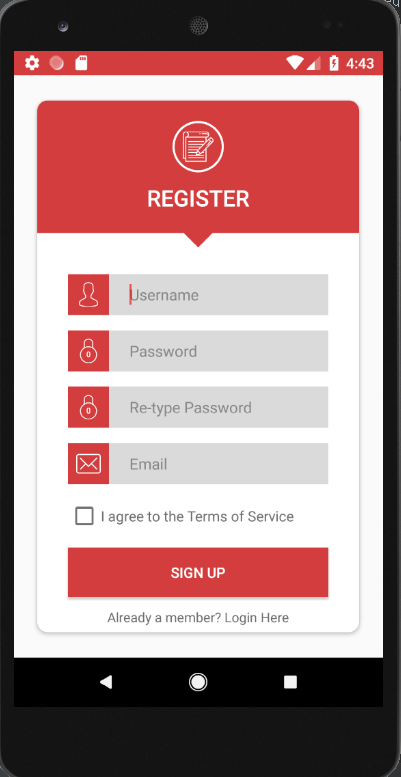
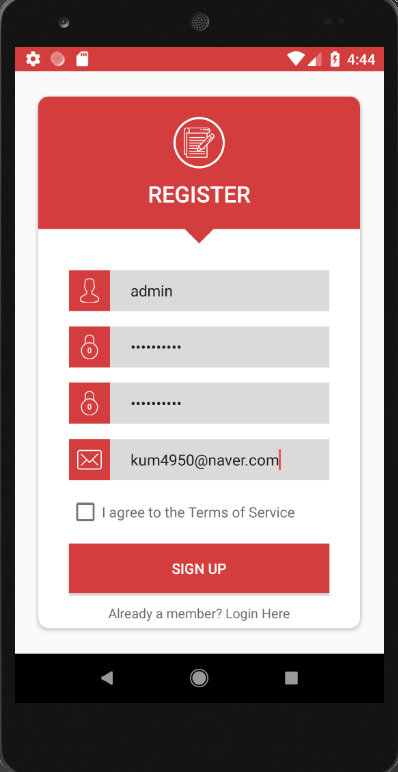




and using LoginRequest class, we can receive the result from the server

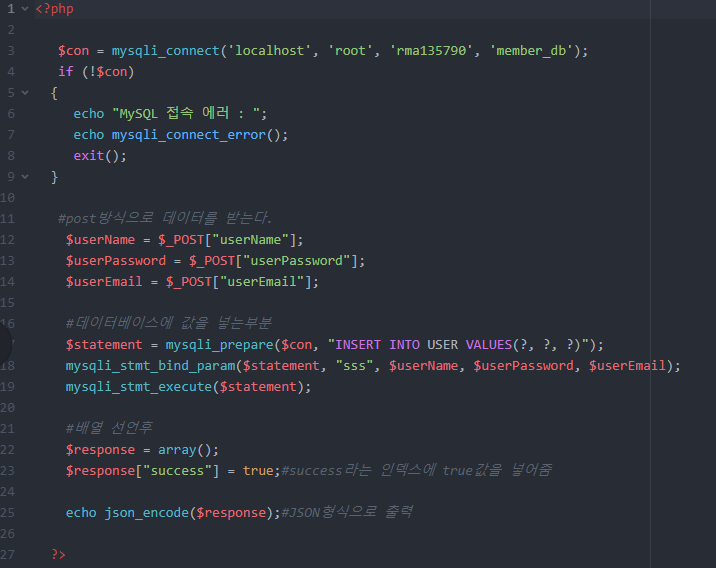


**2. A form allow user to key in structured information such as User Sign-up information.**

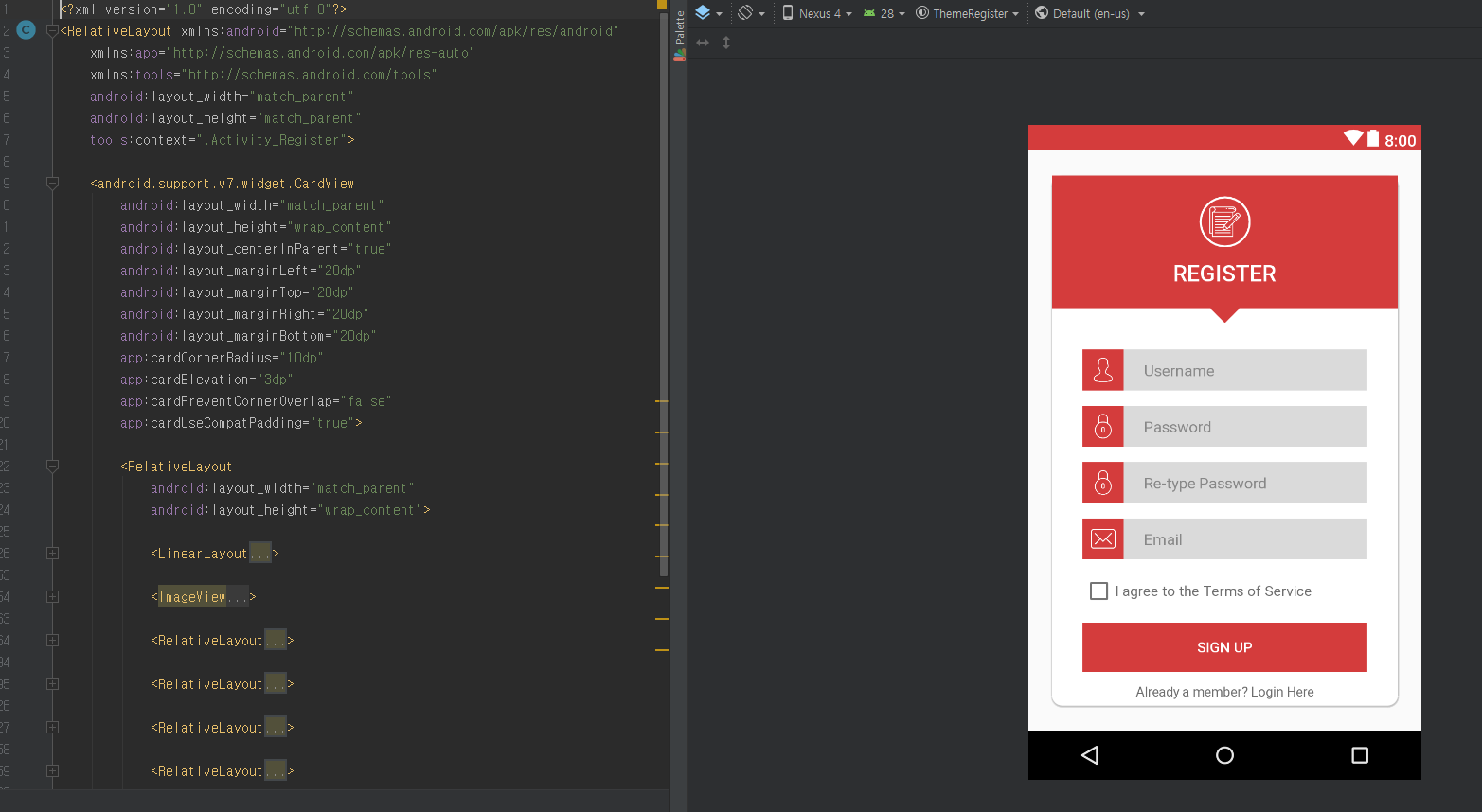
 

**Required API,php,mysql,XML and JAVA Codes :**

**php:** Register\_Gym.php for registering member in member database.

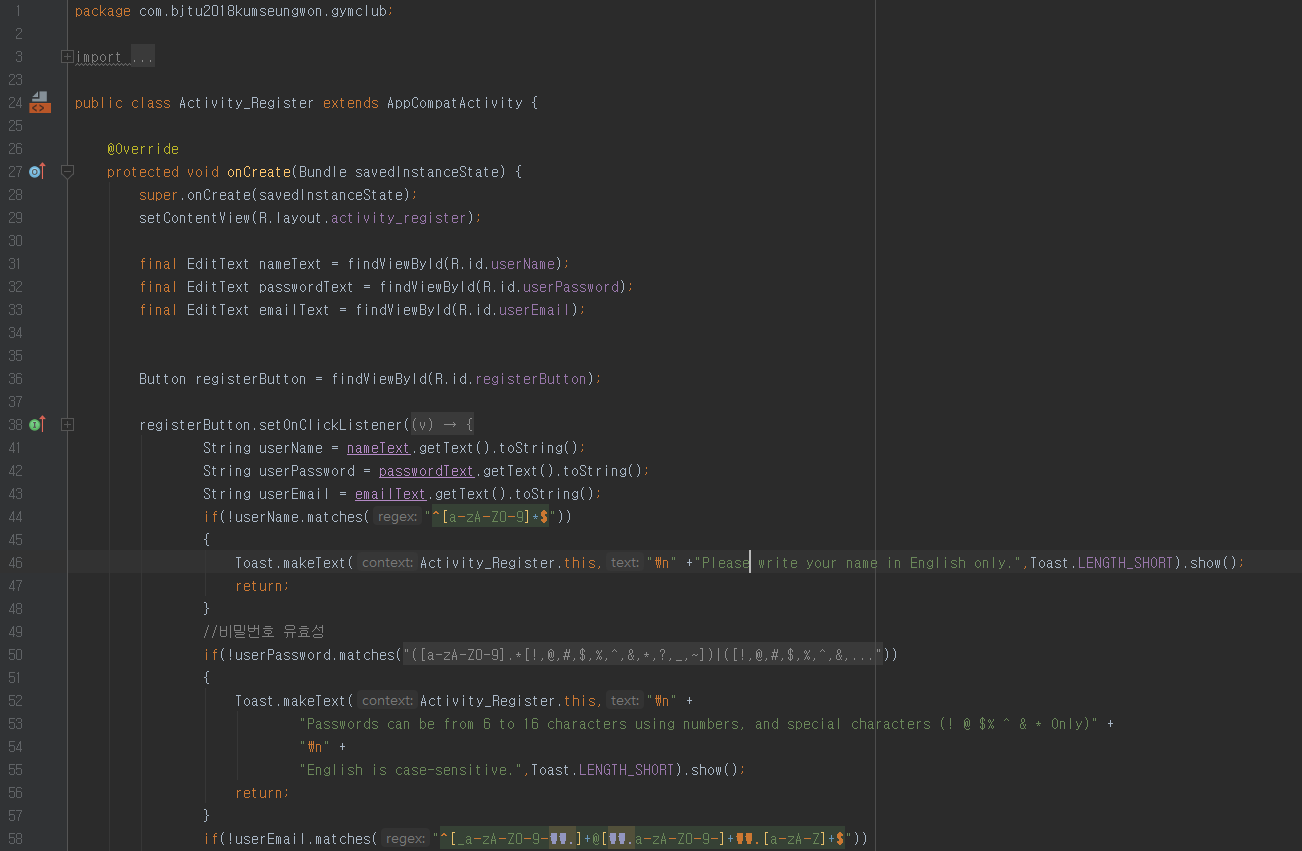


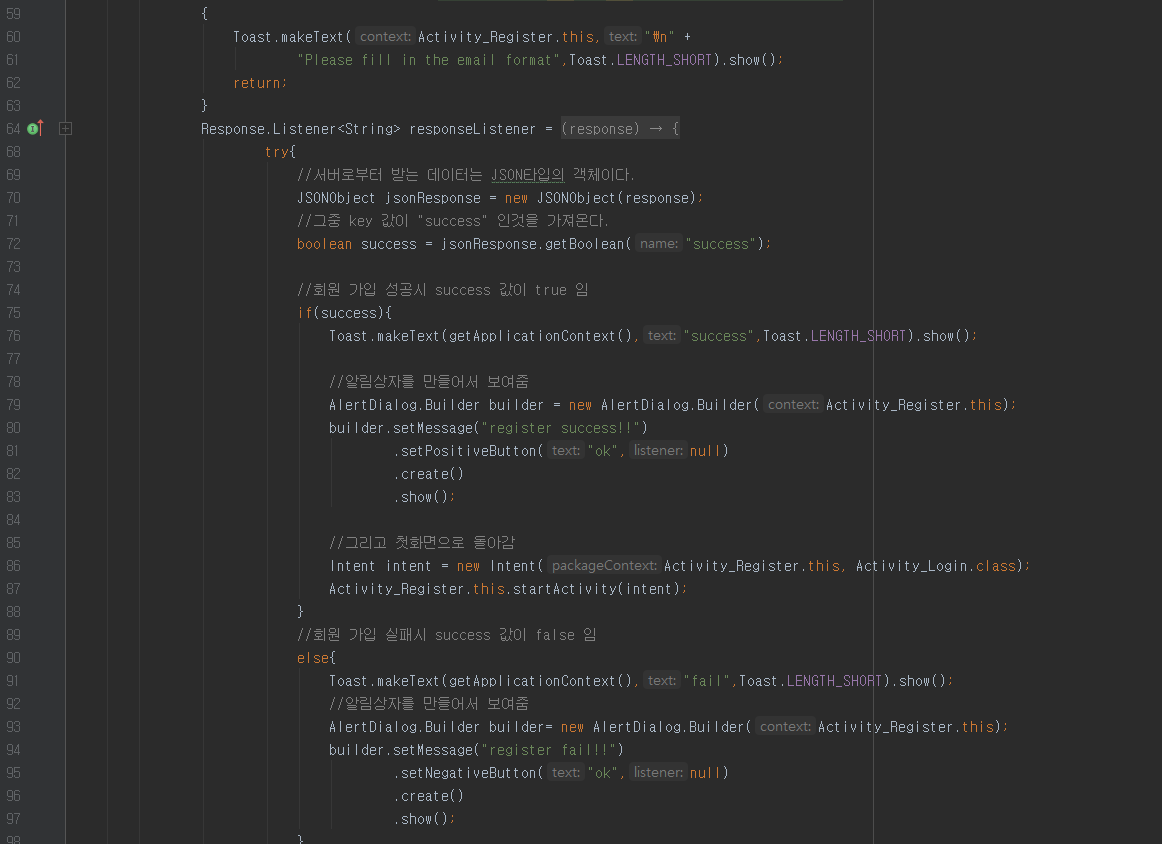
**xml :** activity\_register.xml ( Same as the code used in assignment1)



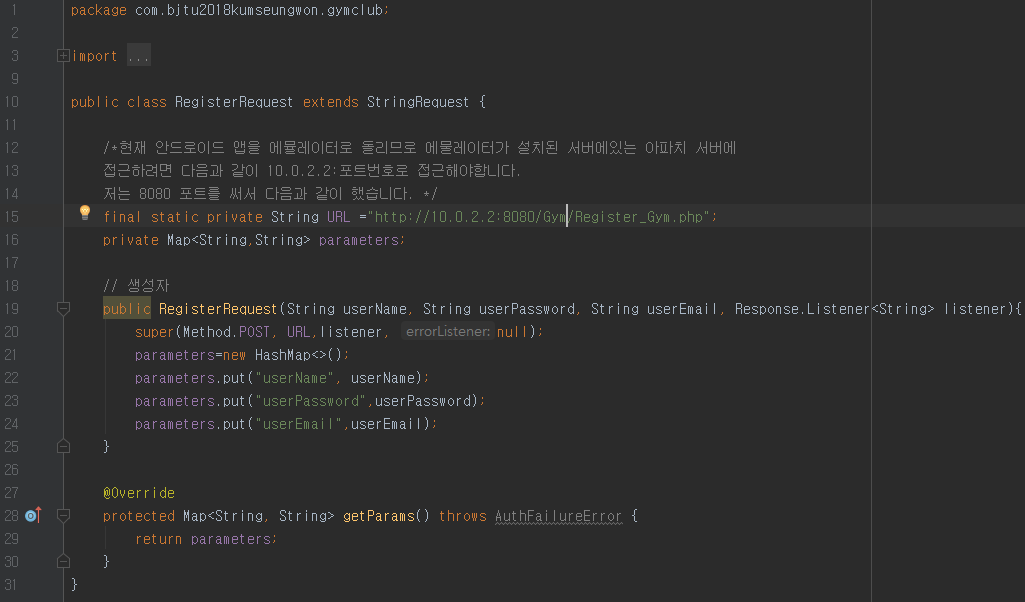
**Java codes :** Activity\_Register.java ,RegisterRequest.java

In "Activity\_Register.java", we send the information entered in "editText" to the server via "RegisterRequest.java"





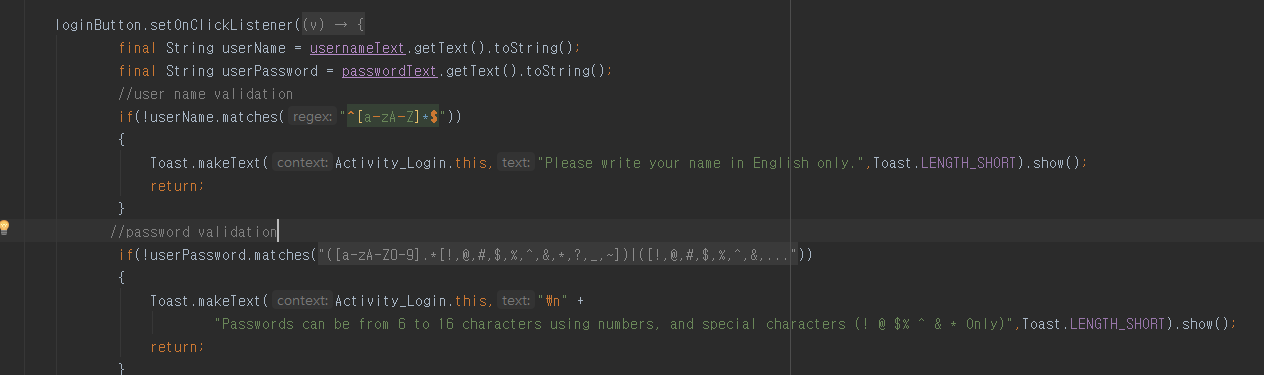
In "RegisterRequest.java", it is possible to send member information, receive the value whether or not it has been registered in the server, and send whether or not "Activity\_Register.java" succeeded.



**3. Validation of user entries**

Added validation to login and sign-up functions.

in Activiy\_Login.java



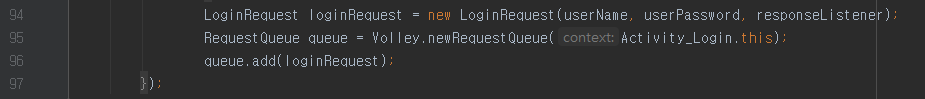
in Activity\_Register.java

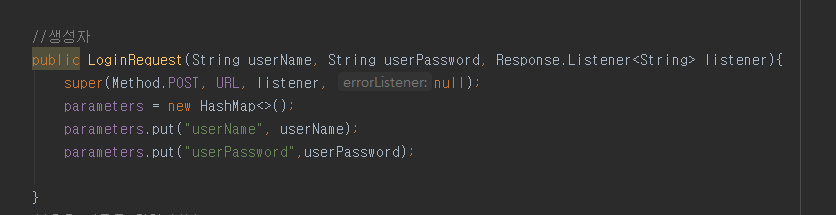


**4.POST the form information to back end with http request**

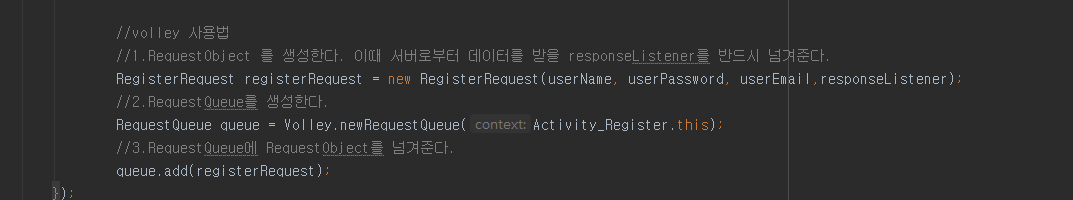
Both login and sign-up functions request information to post in the way to http requset.

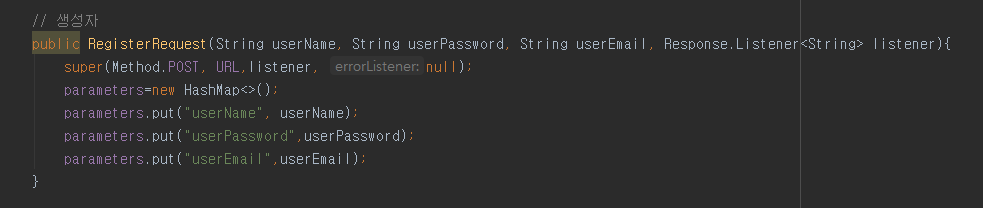
in login functions





in log-up functions

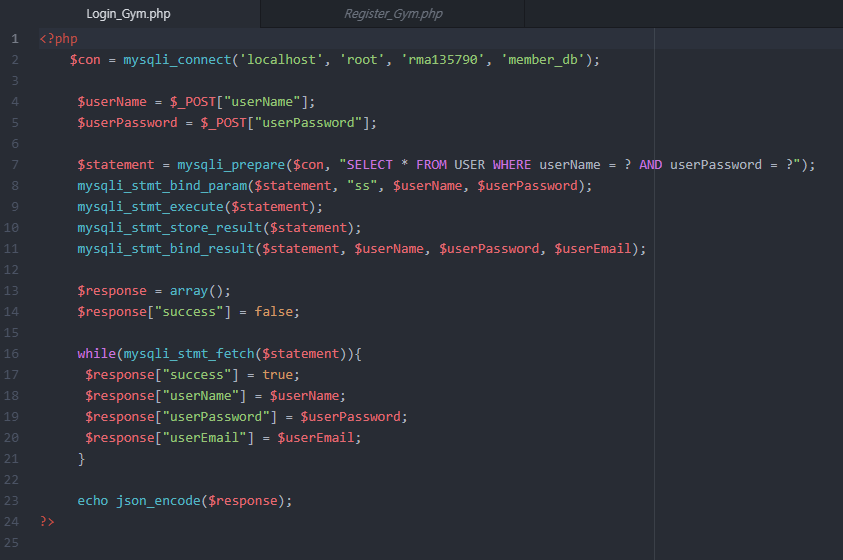




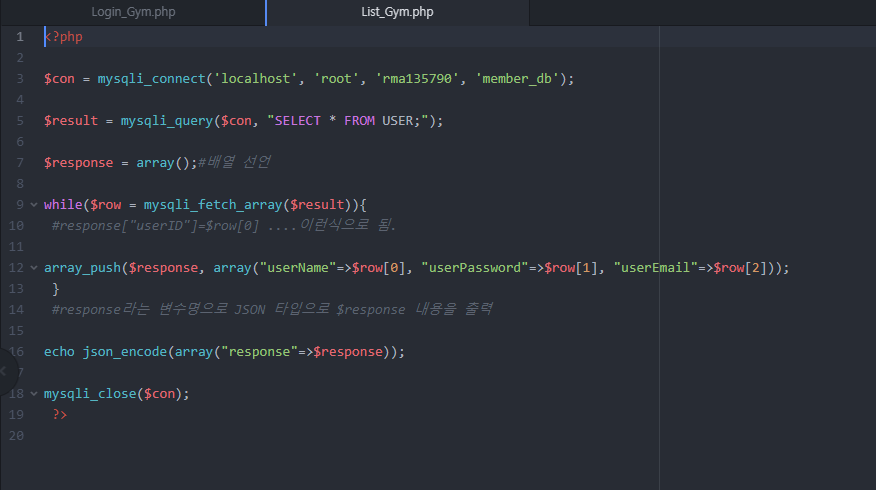
**5.Retrieve data from backend by calling a http request service.**

We have to manipulate the member database through the php language.

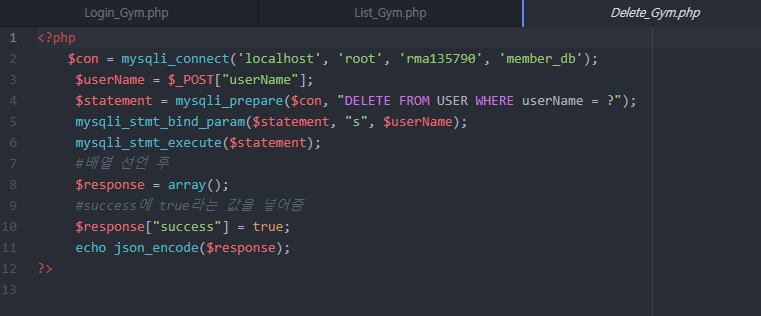
in Login\_Gym.php



in List\_Gym.php



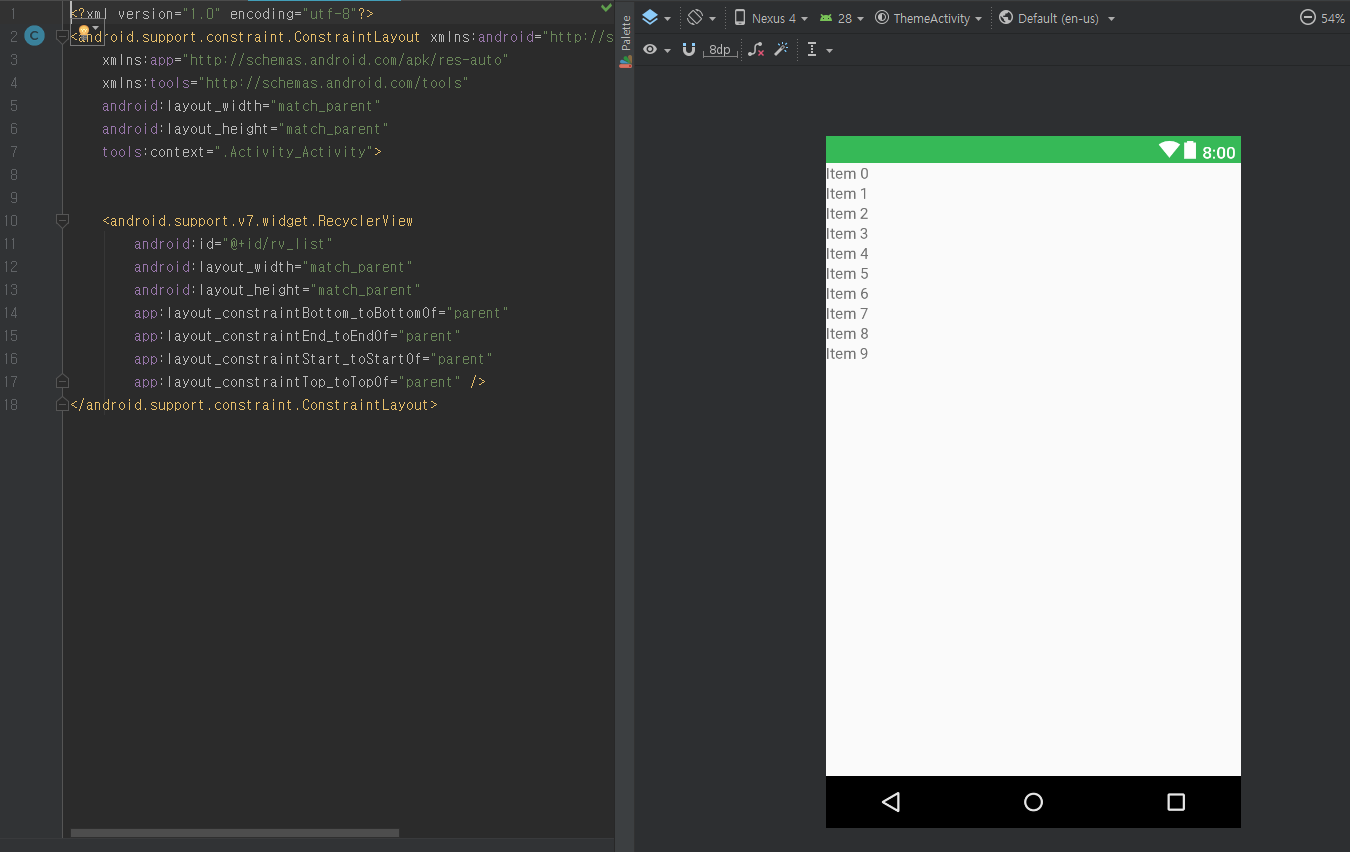
in Delete\_Gym.php



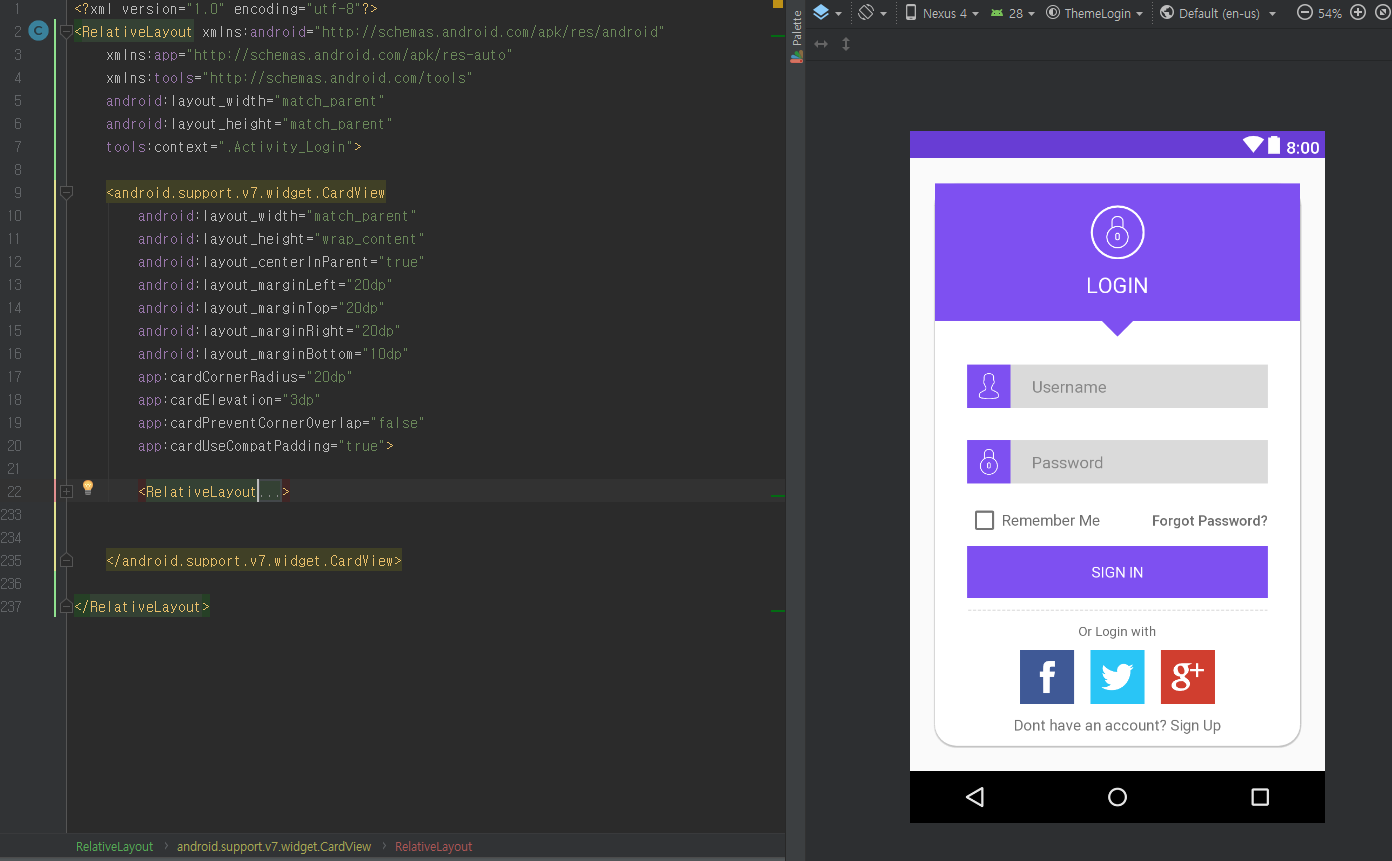
**6. Displaying a set of information(e.g, personal trainer list) with RecyclerView and CardView**

The member database list was created as a “listview”, and the trainer and activity lists were created as "RecyclerView". And also login and sign-up used “cardview”.

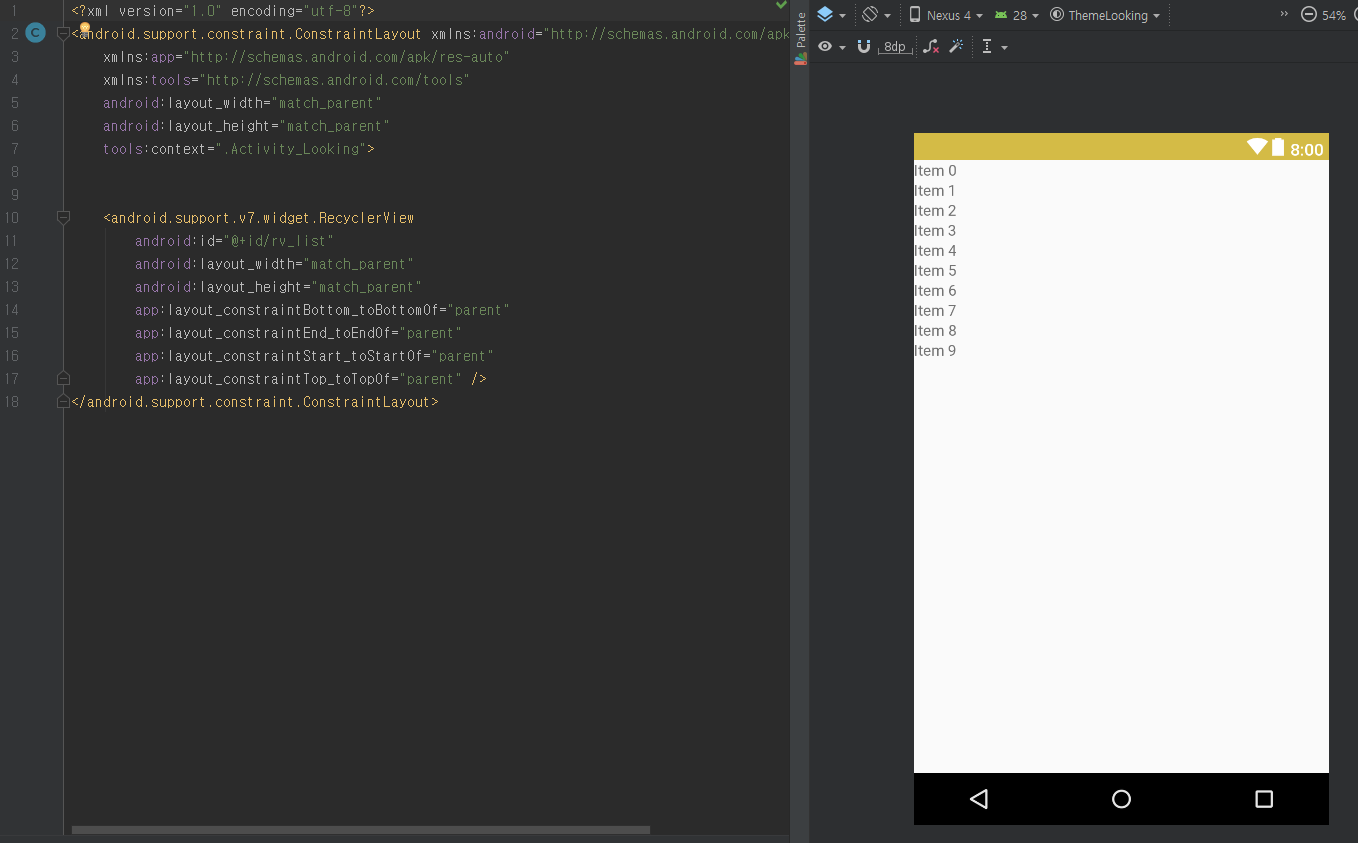
in activity\_activity.xml



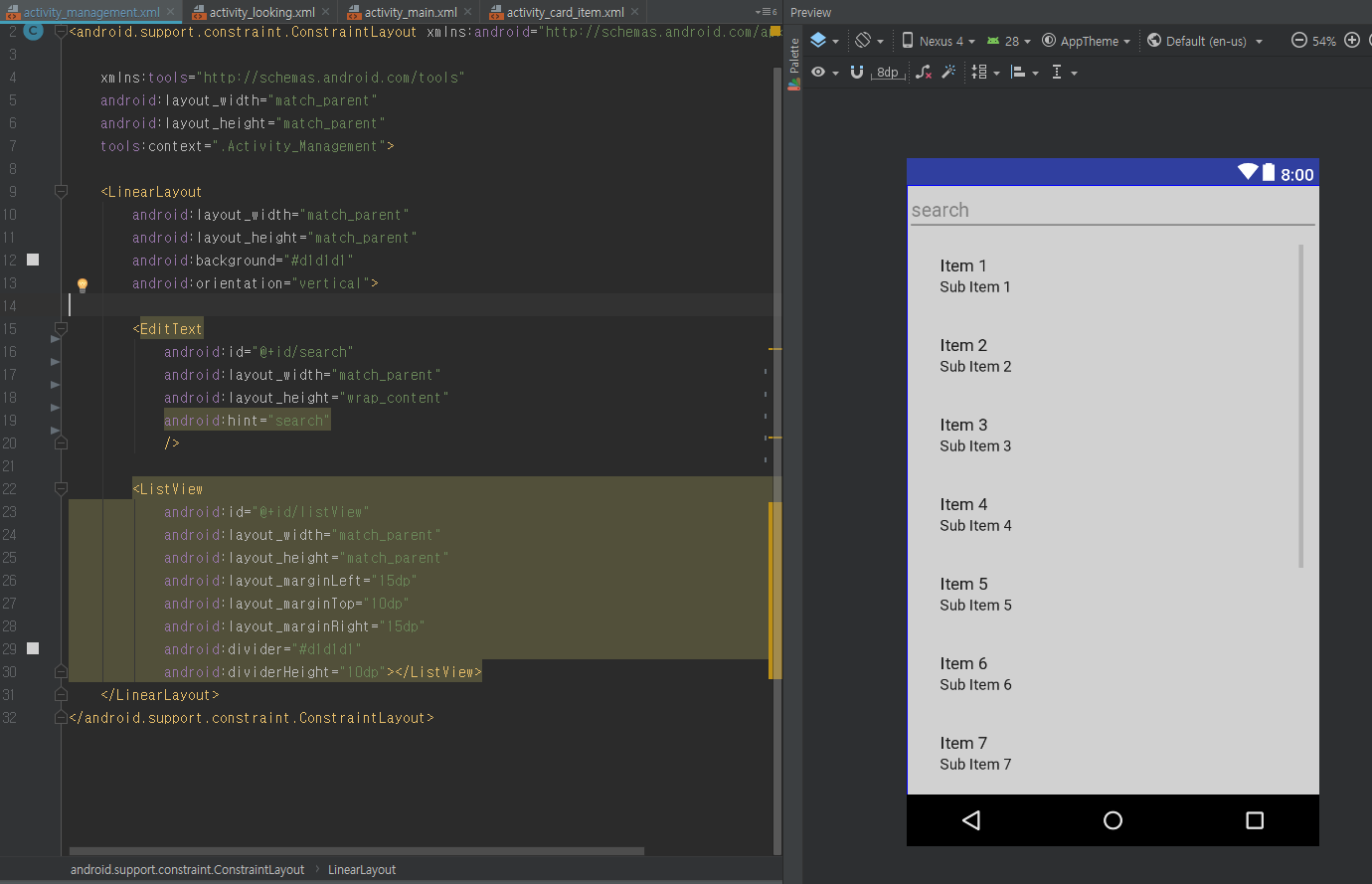
in activity\_login.xml



in activity\_looking.xml



in activity\_management.xml



in activity\_register.xml

