In this assignment you will build an Android App for course registration. This will be a variant of the Javascript version of the registration page which you built earlier. We will provide the servlet for the backend work, and you have to handle the frontend work.

NOTE: There's lots of new stuff to learn. Feel free to discuss how to do things with friends. But the code you submit must be written by you.

1. The backend will be a set of servlets, accessed using HTTP, and returning JSON. Your app will make use of the JSON that is returned by the servlets.

2. Servlet code to handle your backend work is partially provided to you (modified from Govind Lahoti's lab submission). We have provided a few servlets that return JSON.
    1. The servlet login.java should be used for authentication. If user is authenticated user then the servlet returns a string "true", otherwise it returns a string "false". Your Android app will read this flag and perform the corresponding action.
    2. Registration backend functionality is handled by the servlet register.java.   The servlet returns a json array of registered courses.
    3. We have provided a method "ResultSetConverter" in DbHandler.java which takes resultset object as a parameter and returns a jsonarray object. You can use this for your backend JSON creation.
       NOTE: You need to download java-json.jar from moodle, and add it to your build path to use this method.
    4. You can modify other servlets which are provided, which currently return html results, to return json objects instead, or write your own.
    5. NOTE: each servlet needs to ensure that the user is logged in, using session information as usual. In case the user is not logged in the servlet should return an appropriate status, which should result in the app taking the user back to the login screen.
    6. NOTE: You can use HttpURLConnection class in Android to connect to your servlet. Make sure you use the correct IP address of the machine where your servlet is running.
3. The app starts with a login activity (page).
    - You can optionally store the login/password so in future the user does not need to login again.

4. After logging in successfully, you are taken to the next activity which is the home screen.
    1. The home screen first Invokes a servlet that checks that the user is logged in, and the ID is a student ID; if this succeeds it returns a list of currently registered courses in the current year/semester .
    2. Your home screen should then show the list of registered courses (include course ID, section ID, title, department and credits). Invoke the servlet register.java to do this.
    3. The home screen should display along with each registered course an X icon, which can be used to drop the course. You should have a popup that verifies from the user that she wanted to drop the course, and then drops it by invoking the servlet addDelete.java.
    4. The home screen should have a button allows users to add courses. The button should take the user to an activity that does the following:
        1. Allows the user to type in any part of a course name, and display matching courses using autocomplete. The servlet course.java provides you a list of all courses not currently registered by the student.
        2. Note: you should show course ID, title, department and credits with the displayed course.
        3. Once a course is selected, it can be added to the list of registered courses by clicking on an Add course button,
        4. Clicking on the button invokes a servlet addDelete.java to add the course registration of the student to the database and sends back a status. The status is displayed as an alert. When the user clicks OK, the app goes back to the home screen activity, and displays an updated list of courses.
5. OPTIONAL: Define a 3rd activity for browsing courses. This activity allows the user to select a department name. On selecting a department name the list of course sections in the current year/sem offered by the department should be displayed.   Also include a tab view so the user can switch between the home screen and the browse screen.