All of your classes should exist in a *webd4201.lastnamefirstinitial* package (all lowercase).

Submission:

Be sure that you ZIP up your *lastnamefirstinitial* folder that is in the `Tomcat/webapps` folder, with a copy of your source code (i.e. your Eclipse `src` folder) and SQL scripts (in a *sql* folder) placed ***inside*** of it. Also, for this deliverable you are to use the Javadoc tool to generate the API website for your project

Any other compression schemes other than *.zip will be penalized. This *.zip file should be submitted into DC Connect on or before the due date.

Be sure that your generated html conforms to XHTML 1.0 Strict compliance; you must include the clickable image when compliance is met.

Demonstration Requirements:

You are to demo your page(s) during your lab period on the due date. Demo'ing after this date will have to be coordinated on a case-by-case basis, as the class ofter this deliverable is due is the test date for Term Test 2

Javadoc Requirements:

All classes, methods, constructors, attributes, class constants etc. need to have appropriate Javadoc comments and you are required to run the Javadoc tool to create the API website as part of this deliverable. For each case where something does not have appropriate Javadoc comments you will lose one mark out of 20, up to a maximum of 10 marks.

Database Requirements:

You are to continue to use a PostgreSQL database that runs on port 5432 (default), named `webd4201_db` that is owned by a user named `webd4201_admin` that has a password of `webd4201_password`

Your database should be preloaded with tables for `Schools`, `Programs`, `Courses` and `Marks` (scripts can be found in DC Connect). You will have to load the tables in that order. Note `Schools` and `Programs` tables needed to be generated from the provided `3_programs_schools.csv` file

NOTE: all statements are to be `PreparedStatements` in any/all Data Access classes.

When a student successfully logs in (i.e. authenticates) you are to attempt to retrieve all of their marks from the `Marks` table. Records retrieved should be placed into the `marks` attribute for a student, before the student is place on the session and redirected to their `dashboard.jsp`.

Servlet Requirements:

No new servlet is required.

`web.xml` Requirements:

As there is no new servlet, no modifications to the `web.xml` are required.

*.jsp Requirements:

Modify the dashboard.jsp so that it generates a table of the students marks (i.e. a tabled transcript. For full mark the student's cumulative GPA should be calculated and displayed. If the student does not have any marks (i.e. they are a first year, first semester student and their marks Vector is empty) a message stating that the student does not currently have any marks should be displayed on the page. Numbers should be formatted correctly.

NOTE: if someone is not logged in (i.e. there is not a Student object on the session), they should be re-directed off the page with an appropriate message to the `login.jsp`.