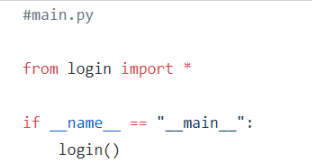
Database Homework 4

박준완, 박윤빈, 마리암

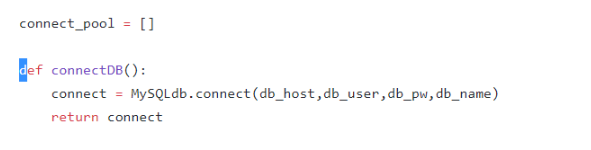
1. Main

By running application, initially run the login.py

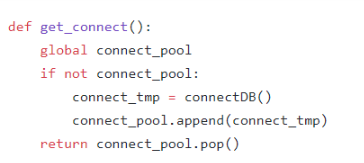


1. DBconnection

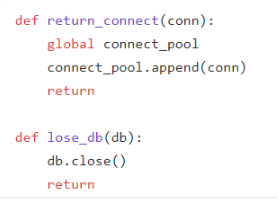
Initial a connection to the database



and return a connection object for every user

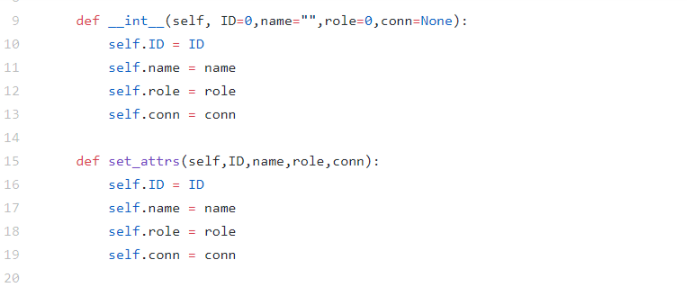


Return connection from user and close the database connection



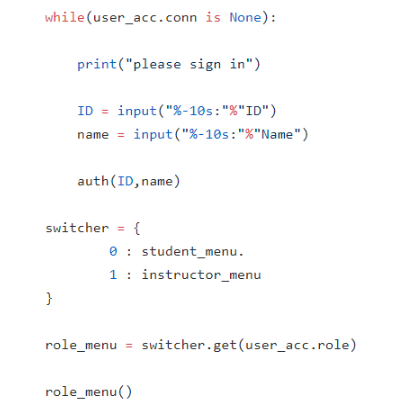
1. UserAcc

Inital a user with null information and provide a method to set the user information in login.py and return user object



1. Login process

Request ID and name from the user, next check id exists in student or instructor table then prints out the corresponding menu

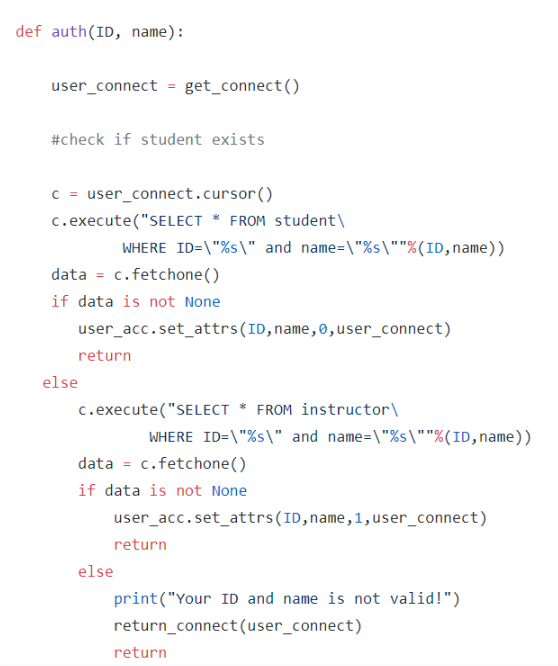


In auth function first we create a connection object to the database

Next use cursor function to execute a query to see if there is a student with the provided user id , if not then check in instructor table and again if not existing, then print out that the user id and name are wrong and close the connection.

But if the user id was found, then by using set\_attr function do assign ID, name,role,connection to the user\_acc

If ID and name was existing in the student table then set role as 0 and if it is existing in instructor then set it as 1.

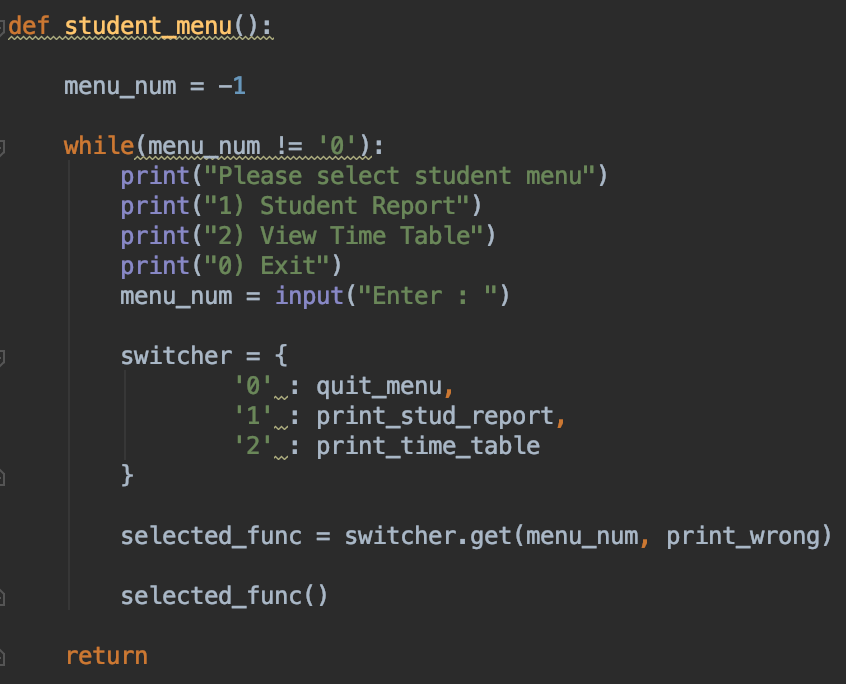


1. Student Menu

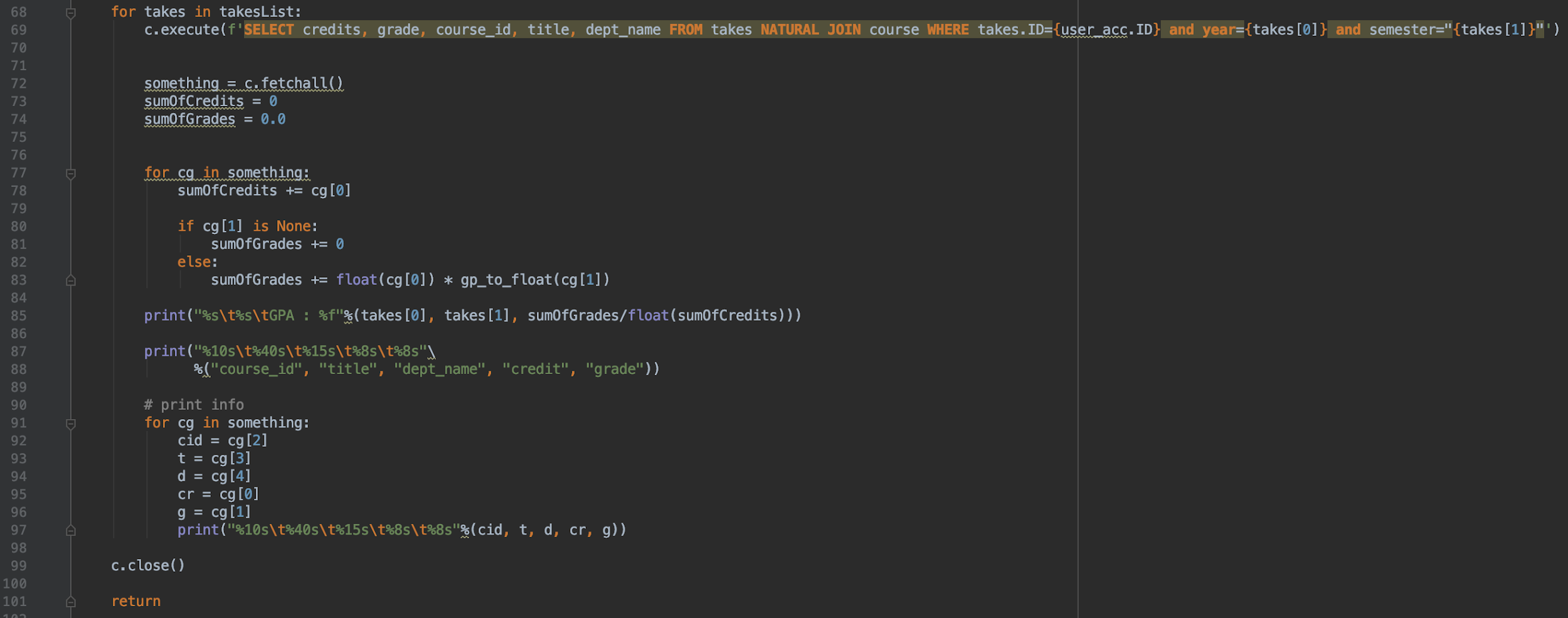
If the role of user be student then student\_menu will be printed and will request to choose one of the provided options

1. Student report
2. View timeline
3. Quit

Next will switch to the corresponding function

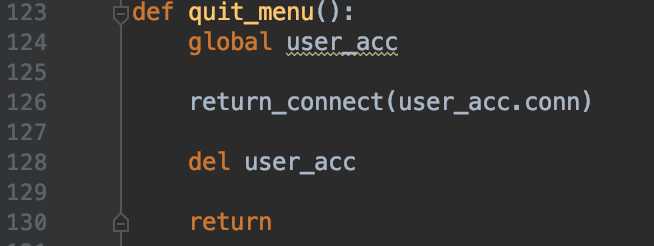


Student Report:

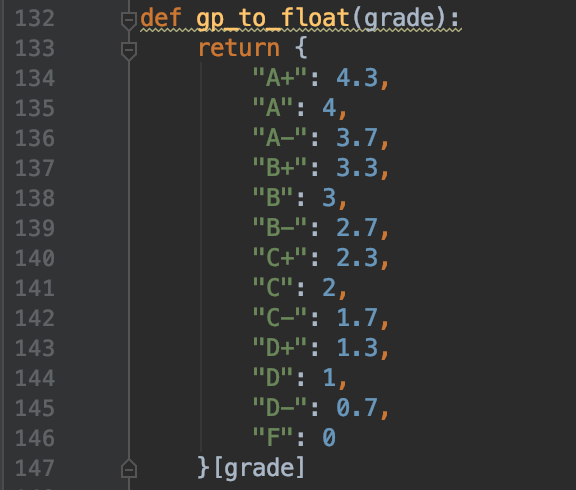


For the student report function, it is required that the semesters were compared to get the most recent semester. The tuples are compared in the order of Winter, Fall, Summer and Spring. After which, the year and semester were sorted in reverse order to list the most recent year and semester first. Then the takes and course tables were natural-joined to get the credits and grades the student received for the respective classes. The sum of the credits were calculated through the for loop, so as for the sum of the grades. Initially, I was ignorant of multiplying the sum of the grades by the number of credits after converting it to the respective scores, so the GPA was oddly small. However, after debugging the mistake, the GPA was converted to floating point and results turned out fine. The GPAs and student reports are printed for every semester. It was difficult having to consider the null values for the GPA calculation so flags were used. And finally they were printed according to the required format.

Quit menu:

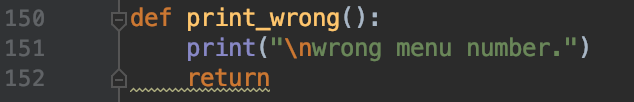


Gp\_to\_float:



The alphabet grades were converted to the respective numerical score for the grade for GPA aggregation.

Print\_wrong:

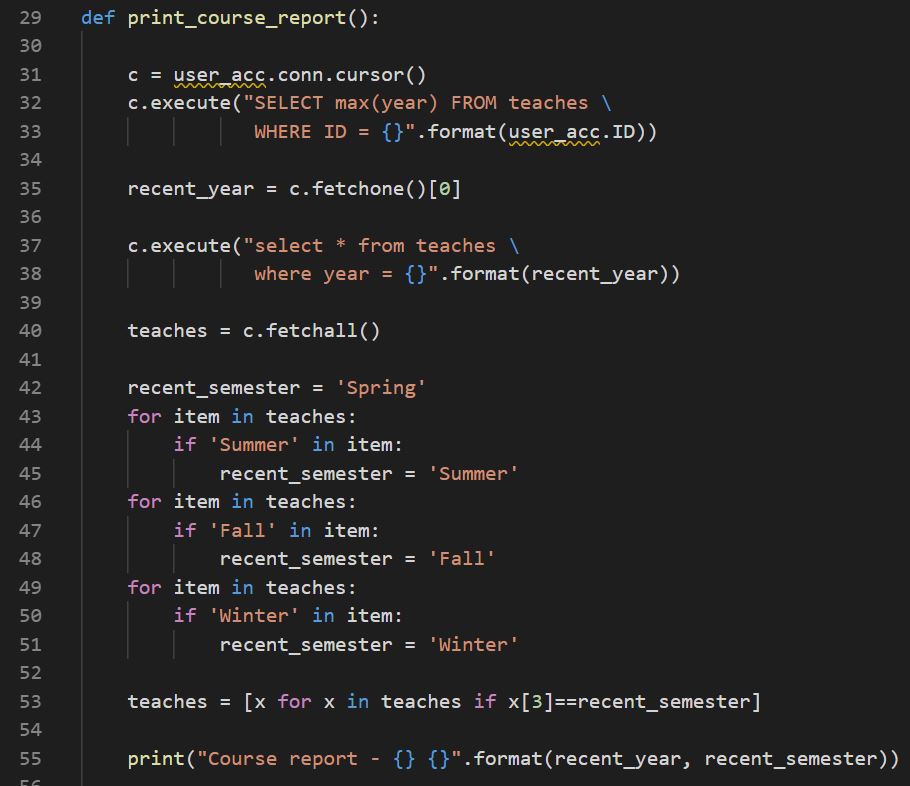


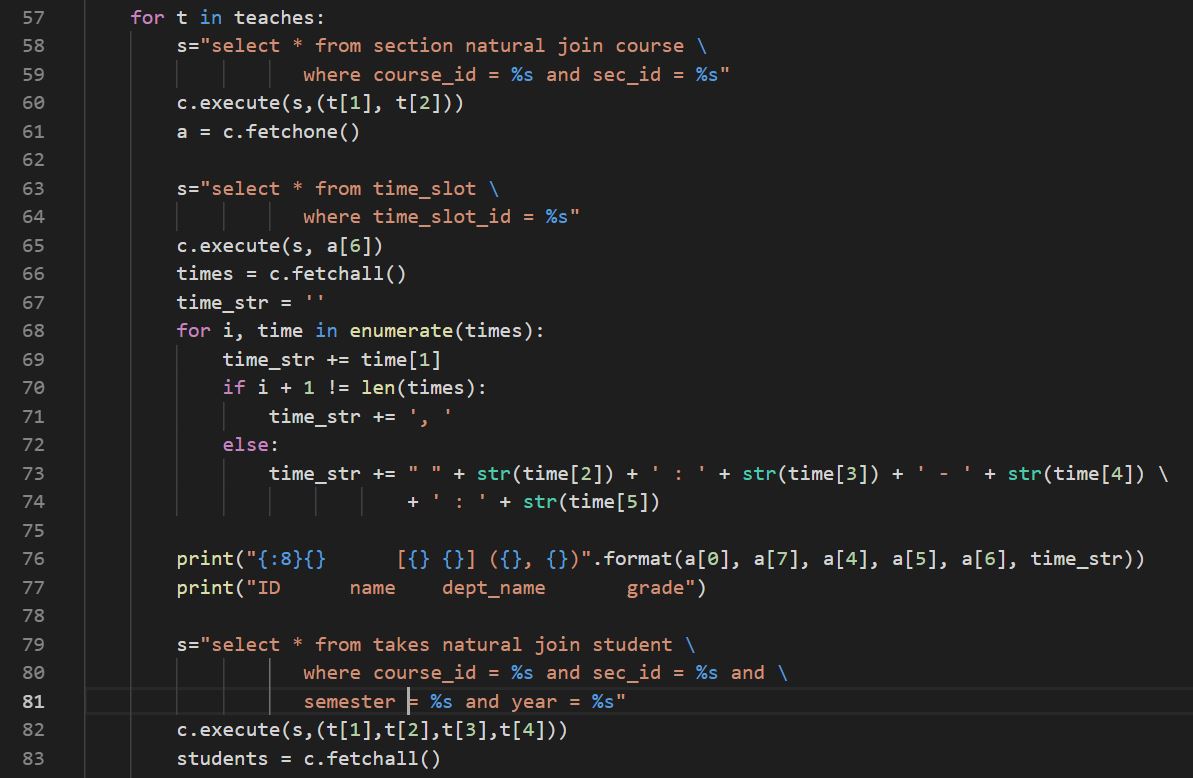
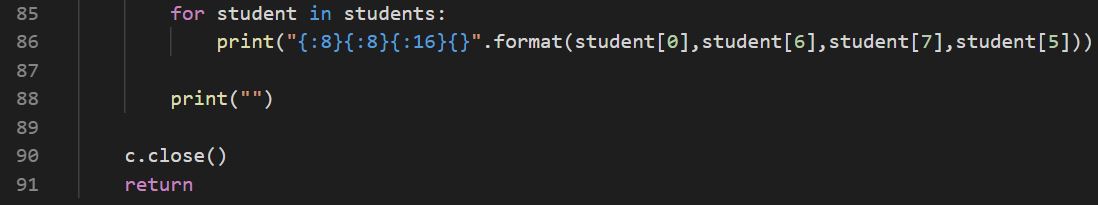
1. Instructor Menu

instructor\_menu

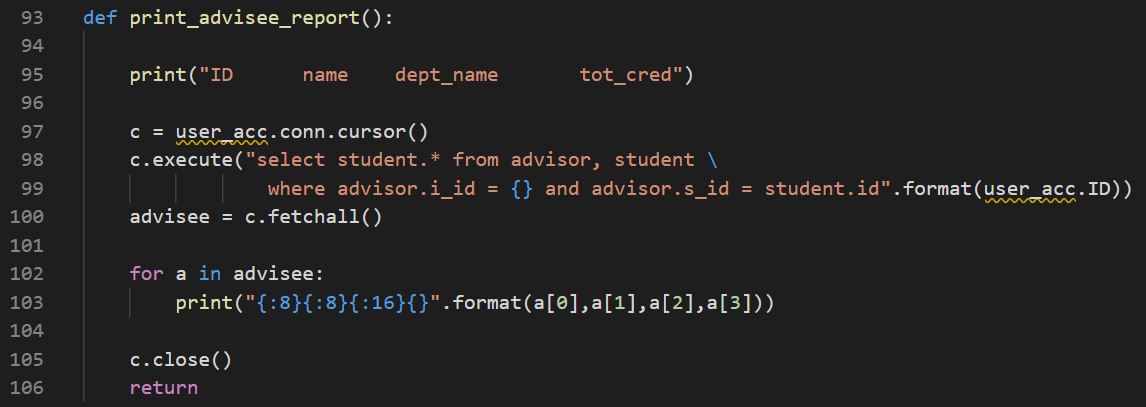
instructor menu to select among course\_report, advisee\_report, and quit.

print\_course\_report

prints course report. Select max(year) from teaches to get the last year of class. Then get the last semester by searching each classes.

then select from section natural join course to print the course information. Then select from takes natural join student to print the student information who takes the class. Repeat the for loop because there could be several classes in that last semester.

print\_advisee\_report

print advisee report. Select student information from advisor, student to get the advisee of the instructor. Then print the information.

1. Contribution

마리암: outline and syntax matching(UserAcc.py,DBconnection.py, main.py), login.py, student-menu function

박윤빈: student\_report, quit\_menu, gp\_to\_float, print\_wrong

박준완: instructor\_menu, print\_course\_report, print\_advisee\_report