**Project 1: Student Management Program**

**Program Functions:**

Write a program to manage Student's test score information with the following objects:

+ Student: (Student ID), Full name, Date of birth, Gender, Address, Phone number,

Email.

+ Subject: (Course code), Course name

+ Test scores: (Student ID, Subject code) is a combination of primary keys created by 2 foreign keys: Process Score, Final Score

**Requirements:**

+ Create user interfaces in the terminal (users can move back and forth between screens):

- Main Menu screen: Display the list of program functions:

* Student Information Management screen
* Course Information Management screen
* Screen of Information Management Test Score

- Manage student information: Add/Edit/Delete/Search students

- Manage subject information: Add/Edit/Delete/Search for subjects

+ Manage exam score information: Enter scores/Edit scores/Delete scores/Look up scores by Student Code or Student's Full Name/Statistical list of Students by Score levels summation (A (90<=score<=100), B (70<=score<=90), C (50<=score<=70), D (score<=50)). Total score is calculated by this formula: Final score = (Process score + Final Score \* 2) / 3

+ Export the transcript to CSV file

Output information includes: Student ID, Full Name, Date of Birth, Gender, Address, Number

Phone, Email, Subject Name, Progress Score, Final Score, Total Score

Note:

+ 3 objects Students, Subjects, Test scores can be understood as 3 data tables

+ Data stored in .txt file. Each object is stored in a separate file. Example of storing the Student object in the file 'hocvien.txt'. Each record

on 1 line, the data fields are separated by the character |

+ Select the appropriate data type for each attribute field

+ Validate input data from keyboard, if not valid, the program will report an error and ask to re-enter

+ Properly handle exception