

LAB5 – PRO192

Type:
Code:
LOC:
Slot(s):

ShortAssignment
LAB5
70
1

Exercise 1 (Score 9):

Title

Create a Java console program to manage students.

Background Context

Write a program to manage information of student. The program implements terminology of Object Oriented Programming (OOP) paradigm. OOP is one of the best choosing ways to design software program.

In this assignment, we will use **ArrayList** to store list of student. In fact, ArrayList is popular used to manipulate with data. ArrayList provides some useful methods, such as: add(), remove(), sort() ., etc.

Program Specifications

A student information consists of ID, Student Name, Address, Phone. **A student has only one ID and this key is unique for a student.** Student Name, Address, Phone is not empty. The program allows use to create list of student, display. On the other hand, use can search student by student id.

1. Main Screen as below:

WELCOME TO STUDENT MANAGEMENT

1. Create
2. View all student information
3. Search
4. Exit

(Please choose 1 to Create, 2 to View all student's information, 3 to Search, 4 to Exit program).

2. Function details:

There are 4 functions in Student Management Screen, as bellow:

1. **Create:** user creates student by **inputting information from keyboard**. After creating the one, the program main screen.
2. **View all student information:** Display all student's information
3. **Searching:** Find and display student (by id)

Technical Requirements

1. Using Object-Oriented programming style
2. Use only core Java functions and classes.

Exercise 2 (Score 1):

Title

Letter count.

Background Context

Knowing the number of **words** in a **document** can be important. For example, if the author is required to write a minimum or maximum amount of words for an article or paper, word counter can help them know if their article meets these requirements.

Program Specifications

Design a program that allows users to **input** the **string value**. **Count** and **display** the number of **letters**:

Expectation of User interface:

```
Enter your content:  
hello world, hello everybody  
{everybody=1, world,=1, hello=2}
```

Guidelines

To tokenize a string, you can use **StringTokenizer** in java.util package