Assignment

LEARNING OUTCOMES

- Build a moderate to advanced stand-alone GUI applications using Java concepts(C3,PL02)
- Demonstrate the use of Java concepts and their functionalities in the existing system(A3,PL05)

Title

Resort Room Booking System

Section A: Program Specification

Design and develop room booking system for a resort. The main purpose of this system is to simulate room booking in a week for a small resort with two views (Jungle & Sea) with each view consist of 10 rooms. The charges per room is RM350.00 per night. The system will be used by resort staff.

The following characteristics are important to be included in your system:

- Staff
  - sign in to use the system
  - book room for customer (Add)

    - system allow staff to choose room based on day. Next the system will show all the available room(s) for the selected day. Now the staff can specify the room to book.

    - Booking details such as customer name, IC/Passport, contact number, email, room id, days of stay etc should be included while booking.
  - manage room booking details (Modify/Delete/Search/View).
  - view receipt which consist details of customer, booked room and total charges including taxes.

    - taxes imposed are service tax (10% of total room charges) plus tourism tax (RM10 per night)

Once the booking is done, the details should be stored in text file and the room(s) will be blocked for the selected day(s).

GUIs should be implemented for user interactions. The system should be running continuously unless an exit command is issued.

The program submitted should compile and be executed without errors. Besides, validation should be done for each entry from the users in order to avoid logical errors.

Section B: Deliverables

This is an individual assignment. Each individual is required to submit:

1. A softcopy of the program coded in Java . The program should include the following:
    - Basic Java concepts such as displaying and reading of text, variables, and assignment of values, comments – to explain various parts of the program, selection control and iteration structures, and arrays – single/double scripted.
    - Object-oriented concepts incorporated using Java such as definition of classes, creation of objects / arrays of objects, constructors, method overloading, method overriding, inheritance etc.
    - Any other aspects of Java.
2. A documentation of the system, that incorporates basic documentation standards such as header and footer, page numbering, and which includes:
    - Cover page
    - Table of contents
    - Sample outputs when the program is executed with some explanation of the outputs/sections of the program
    - Sample code to discuss the OO concepts and Java features used in system
    - Additional features which have been incorporated in the solution in terms of Java codes
    - Assumptions
    - All references must be made using the Harvard Naming Convention

Submission of both report and system is via Webspace(Moodle)

Section C: Component Weighting

Coding : 50%

Documentation : 30%

Presentation : 20%

Plagiarism is a serious offence and will be dealt with according to APU regulations on plagiarism.

Section D: Performance Criteria

Distinction

- This grade will be assigned to work which solution meets more than 75% of the basic requirements.
- The program should compile and run with no errors.
- Clear evidence of appropriate usage of advanced concepts of Java such as interfaces, packages, and abstract classes. Work at this level has to show appropriate use of basic programming concepts with appropriate use of features not presented in class.
- Program must be a unique solution.
- All documentation requirements must be met professionally with referencing done appropriately.
- During presentation, the student should be able to open and execute the program. Student should also be able to demonstrate and rationalize the need for certain codes. Also be able to answer the questions correctly with detailed explanation.

Credit

- This grade will be assigned to work which solution meets more than 65% of the basic requirements.
- The program should compile and run with no errors.
- Clear evidence of appropriate usage of basic programming concepts such as looping, control structure, and array.
- Program must be a unique solution.
- All basic documentation requirements met. Referencing was done but with errors.
- During presentation, the student should be able to open and execute the program. Student should also be able to explain most of the work produced. Also be able to answer the questions correctly.

Pass

- This grade will be assigned to work which is considered to be of average standard and which meets more than 50% of the basic requirements listed above.
- The program should compile with no errors or run when executed but with some errors.
- Work at this level must provide clear evidence of appropriate usage of basic programming concepts such as looping, control structure, and arrays.
- Referencing was done but with errors.

- During presentation, the student should be able to open and execute the program. Student should also be able to explain the work produced. Also be able to answer most questions correctly.

Marginal Fail

- Work at this level will generally be of low standard where it may even fail to meet less than 50% of the basic requirements listed above.
- The program should compile with no errors and run when executed but with some major errors.
- Work at this level must provide clear evidence of some usage of basic programming concepts such as looping, control structure, and arrays.
- No referencing was done.
- During presentation, the student should be able to open and execute the program. Student barely able to explain the work produced and could not answer most questions correctly.

Fail

- Work at this level will generally be of low standard where it may even fail to meet less than 40% of the basic requirements listed above.
- The program does not compile and/or run when executed but with some major errors.
- Work at this level must show at least little usage of basic programming concepts such as looping, control structure, and arrays.
- Barely any documentation done.
- During presentation, the student not able to open and execute the program. Student also not able to explain the work produced and could not answer any of the questions asked.