CSCI3170 (2015-2016 1st term)
Introduction to Database Systems
Project – Sales System

Group Registration Deadline: 23:59 9th October 2014
Phase 1 Deadline: 23:59 30th October 2015
Phase 2 Deadline: 23:59 27th November 2015

1 Introduction

You are required to implement a sales system for a computer part store so that all information about transactions, computer parts and salespersons is stored. The system has to support interactive inquiries from users. You are required to use Java JDBC API to access the database and implement a Java application to satisfy all system functions defined in this specification.

There are two phases. In phase 1, you are required to design the database for the sales system (including an ER-diagram and a relational schema which doesn't contain redundant fields and tables). After the deadline of Phase 1, a suggested solution will be provided. You are required to use the suggested solution as a guideline to complete Phase 2. In Phase 2, you are required to implement the sales system as a Java command-line program. Our tutors will give tutorials on how to connect to Oracle database system through Java JDBC API and deploy your work on the required platform.

This is a group project and each group consists of three members. Each group is required to submit ONLY one solution for each project phase. Please sign the group registration form on the course homepage before the group registration deadline.

2 Milestones

Preparation

- Read the document thoroughly and make sure you understand all the assumptions and regulations stated in Section 4.

Phase 1

- According to the data specifications in Section 3, design an ER-diagram and transform it into a relational schema which doesn't contain redundant fields and tables.

Phase 2

- According to the suggested solution of Phase 1, implement a Java application that fulfills all requirements stated in Section 5.
- Debug your system with different datasets and user inputs.
- Write a readme file to describe the compilation and deployment of your system.

3 Data Specifications

All data files for the system are in UNIX text file format (i.e. Newline character is \n) encoded in ASCII. Your Java application is required to read records stored in the files and insert them into appropriate tables in the provided Oracle DBMS via JDBC API. There are five input files, a list of categories, a list of manufacturers, a list of parts, a list of salesperson and a list of transaction records.

3.1 Category

Each computer part belongs to a category. In this system, there are several categories and each of them has its own name.

Item Name	Format	Description
Category ID	Non-empty positive integer with exactly 1 digit.	A unique identifier for a category.
Category Name	Non-empty string with at most 20 characters.	The name of a category.

3.2 Manufacturer

Each computer part is produced by a manufacturer. Each manufacturer has its name, address and phone number.

Item Name	Format	Description
Manufacturer ID	Non-empty positive integer with at most 2 digits.	A unique identifier for a manufacturer.
Manufacturer Name	Non-empty string with at most 20 characters.	The name of a manufacturer.
Manufacturer Address	Non-empty string with at most 50 characters.	The address of a manufacturer.
Manufacturer Phone Number	Non-empty positive integer with exactly 8 digits.	The phone number of a manufacturer.
Warranty Period	Non-empty positive integer with exactly 1 digit.	The length of the warranty period of a manufacturer in term of years.

3.3 Part

Each computer part has its name, manufacturer, category and available quantity.

Item Name	Format	Description
Part ID	Non-empty positive integer with at most 3 digits.	A unique identifier for a part.
Part Name	Non-empty string with at most 20 characters.	The name of a part.
Part Price	Non-empty positive integer with at most 5 digits.	The price of a part.
Part Manufacturer ID	Non-empty positive integer with at most 2 digits.	The manufacturer ID of a part.
Part Category ID	Non-empty positive integer with exactly 1 digit.	The category ID of a part.
Part Available Quantity	Non-empty positive integer with at most 2 digits.	The quantity of a part available for sale.

3.4 Salesperson

Salespersons are responsible for selling computer parts in a store. Each salesperson has its name, address and phone number.

Item Name	Format	Description
Salesperson ID	Non-empty positive integer with at most 2 digits.	A unique identifier for a salesperson.
Salesperson Name	Non-empty string with at most 20 characters.	The name of a salesperson.
Salesperson Address	Non-empty string with at most 50 characters.	The address of a salesperson.
Salesperson Phone Number	Non-empty positive integer with exactly 8 digits.	The phone number of a salesperson.

3.5 Transaction Records

There is a record for each transaction done in the sales system. It records the part sold, salesperson involved and the date of the transaction.

Item Name	Format	Description
Transaction ID	Non-empty positive integer with at most 4 digits.	The ID of a transaction record.
Part ID	Non-empty positive integer with at most 3 digits.	The ID of the part sold.
Salesperson ID	Non-empty positive integer with at most 2 digits.	The ID of the salesperson.
Transaction Date	Non-empty date in the format of DD/MM/YYYY.	The date of the transaction.

4 Assumptions and Regulations

4.1 System

- All numerical values will not be larger than the maximum integer value that can be handled by Java.
- The system is case sensitive.
- Every date has the following format: [DD]/[MM]/[YYYY] and has the same time zone as Hong Kong (GMT+8). (Note: Y=year, M=month, D=day)
- There is no duplicate row in any input and output.
- There is no empty row in any input and output.
- The current date is the system date of the Oracle DBMS server.
- Your Java program may assume that any value entered into any input field is correct **in format only**.
- Your Java program may assume that any data file inputted into it is correct **in format and content**.

4.2 Category

- The ID and the name of a category are both unique.
- All categories are identified by their ID.

4.3 Manufacturer

- The ID of a manufacturer is unique and all manufacturers are identified by their ID.
- Some manufacturers may share the same name, phone number or address.

4.4 Part

- The ID of a part is unique and all parts are identified by their ID.
- Some parts may have the same name, price, available quantity, manufacturer ID or category ID.

4.5 Salesperson

- The ID of a salesperson is unique and all salespersons can be identified by their ID only.
- Some salespersons may share the same name, phone number or address.

4.6 Transaction Records

- The ID of a transaction is unique and all transactions can be identified by their ID only.
- Each transaction corresponds to the sale of one part.
- Some salesperson may never sell any part and some parts may have never been sold.
- A part can be sold only if it is still available (i.e. available quantity > 0).
- A salesperson can sell the same part more than once by having more than one transaction.
- There may be more than one transaction in one day.
- After a salesperson sells a part, the system should reduce the available quantity of that part by one and add a transaction record accordingly.

5 Function Requirements

You are required to write a simple command line application in Java. After performing a function specified in any of the following sub-sections, the program should go back to the topmost level of menu. Any error or informative message of the Java program should be displayed in a new line. The Java program consists of the following functions:

5.1 Administrator

The functions that can be used by an administrator are:

- **Create table schemas in the database**: This function creates all tables for the sales system in the Oracle DBMS based on the relational schema given.

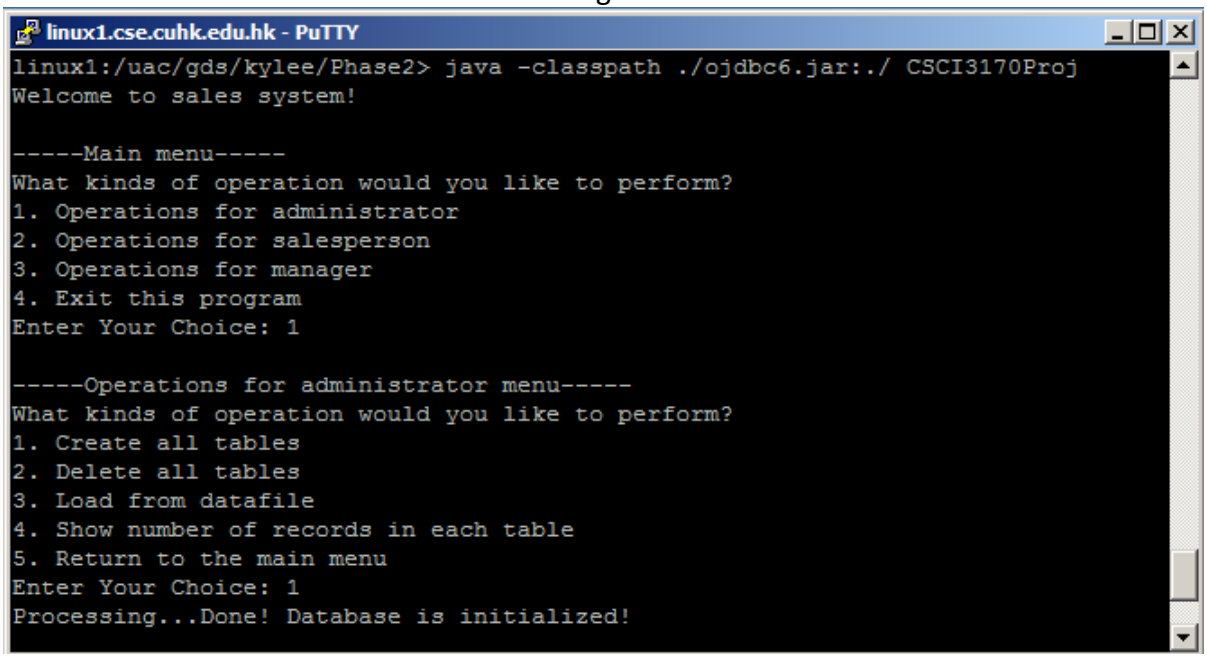

Figure 1: Expected interactive input and output while creating table schemas in Oracle DBMS.

- **Delete table schemas in the database:** This function deletes all existing tables of the sales system from Oracle DBMS.

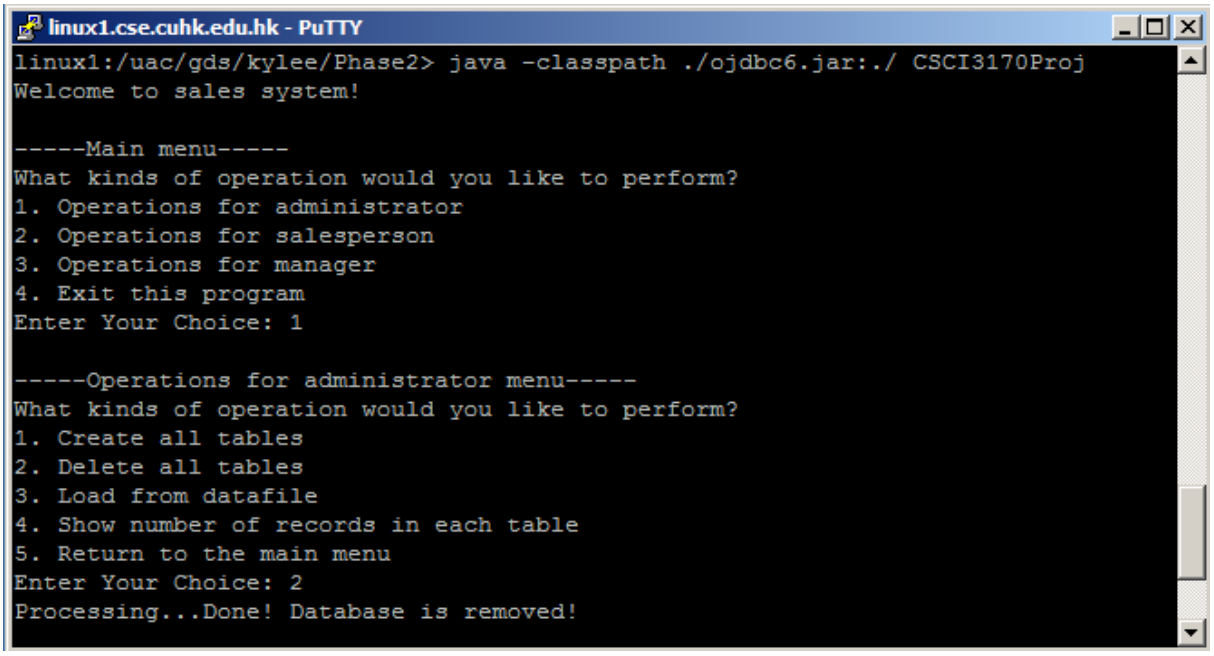

Figure 2: Expected interactive input and output while deleting table schemas from Oracle DBMS.

- **Load data from a dataset**: This function reads all data files from a user-specified folder and inserts the records into the appropriate table in the database. (Your program can assume that the user-specified folder must contain all five input files. These five input files are named *category.txt*, *manufacturer.txt*, *part.txt*, *salesperson.txt* and *transaction.txt*. Each data file stores the data corresponds to its filename.)

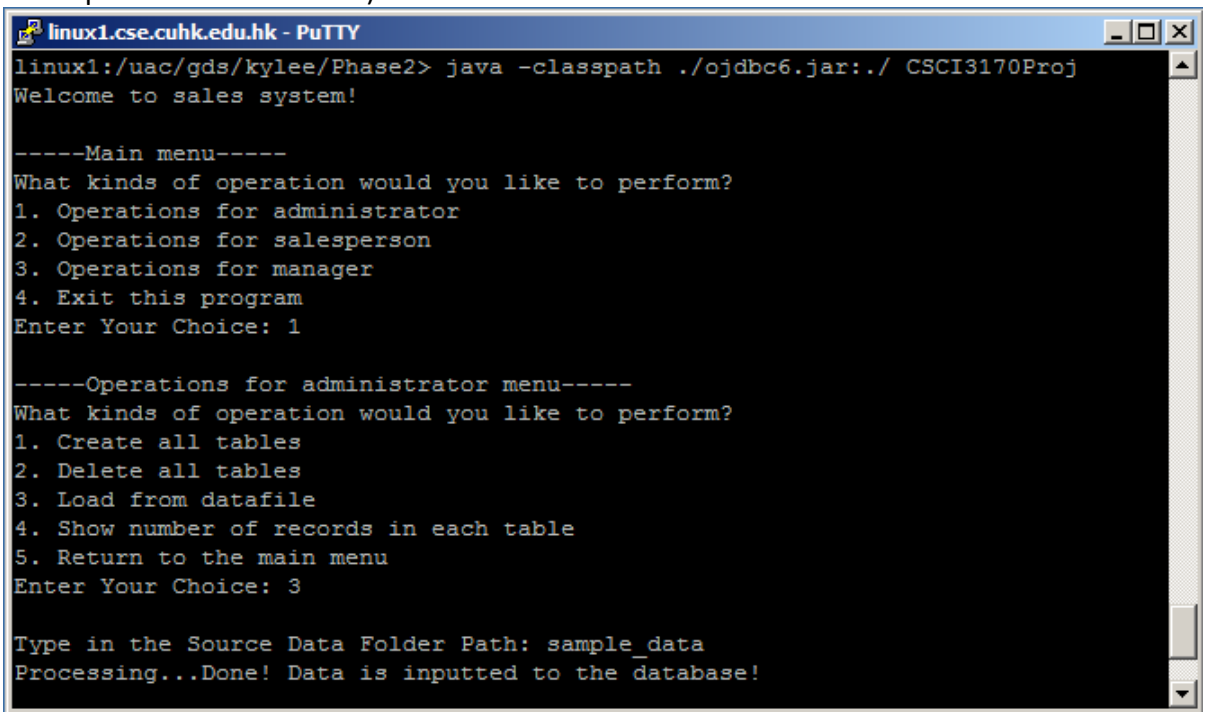

Figure 3: Expected interactive input and output while loading data from data files to the table schemas in Oracle DBMS.

- **Show the number of records in each table:** This function shows the total number of records in each existing table.

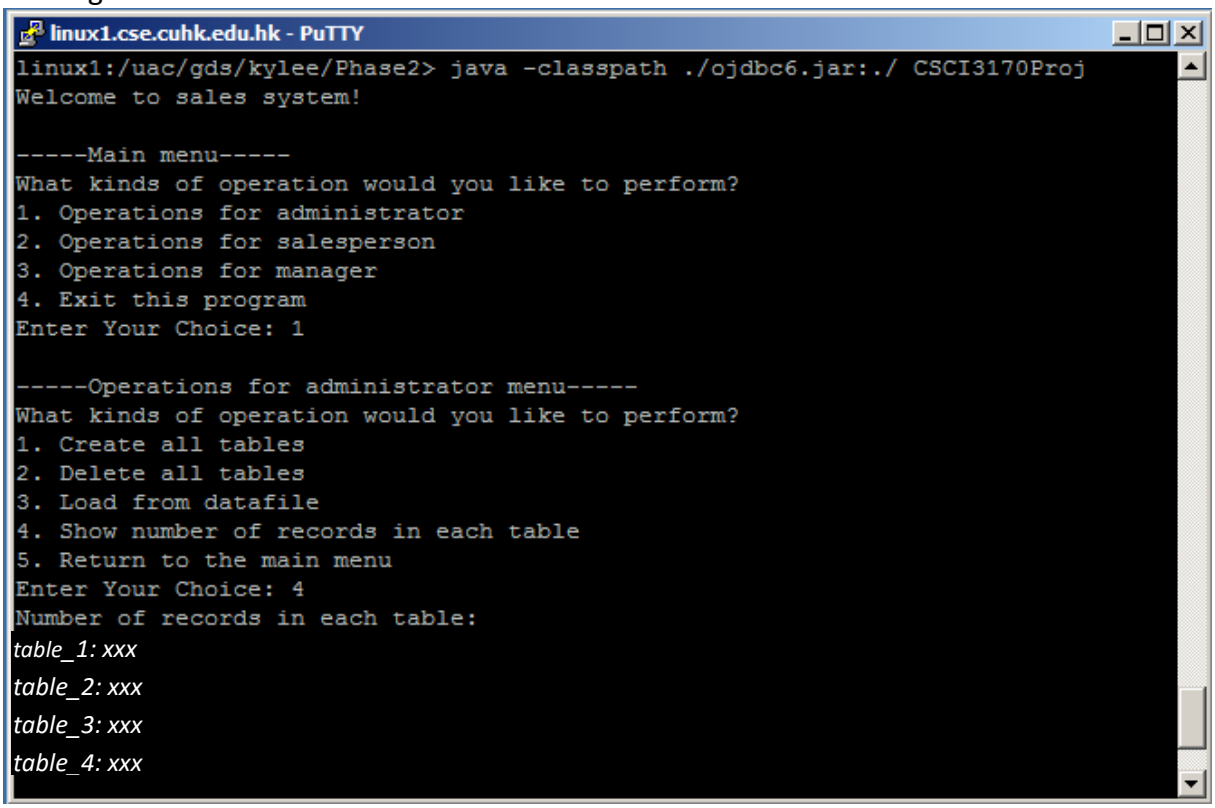

Figure 4: Expected interactive input and output while showing number of records in each table.

(Note: Please replace the words in Italic in the figure 4 with the tables in relational schema given in the suggested solution of phase 1. The number of tables may not be four as shown in Figure 4.)

5.2 Salesperson

The functions that can be used by a salesperson are:

- **Search for Parts**: The system has to provide an interface to allow a salesperson to search for computer parts available in the store based on any one of the two different search criterions below.
  - By Part Name (partial matching)
  - By Manufacturer Name (partial matching)

  You can assume that only one search criterion can be selected by the salesperson for each query. After he/she enters the search keyword or range of price, the program should perform the query and return all matching parts in terms of their *Part ID*, *Part Name*, *Manufacturer Name, Category Name, Available Quantity, Warranty Period and Part Price*. The salesperson can then choose any one of two different ways to sort the parts:
  - By price, ascending order
  - By price, descending order

  Finally, the result should be outputted as a table as follows:

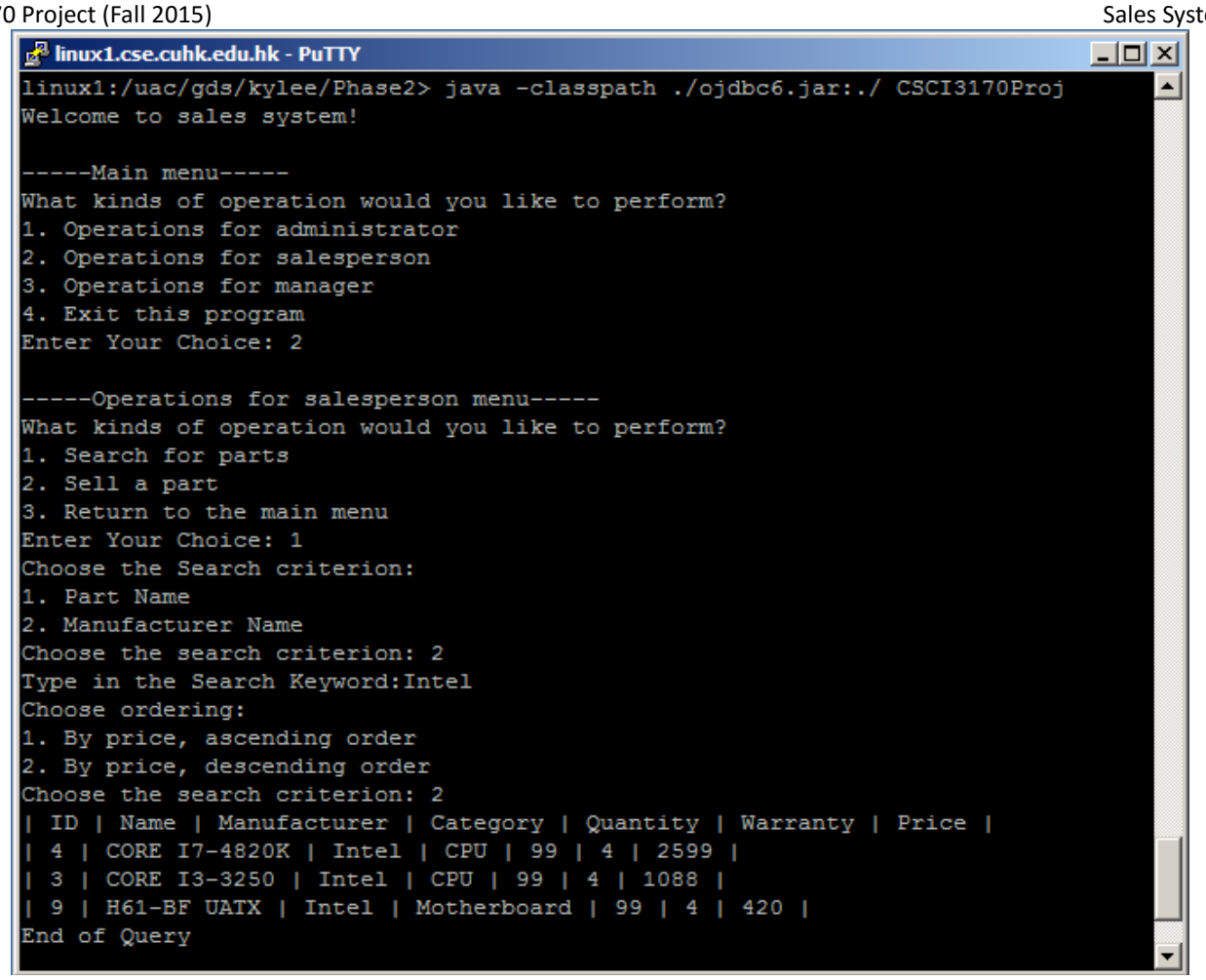

Figure 5: Expected interactive input and output while searching for parts

- **Perform Transaction:** After a salesperson helps a customer finding a part, he/she can then sell the part (i.e. perform a transaction) through the sales system. First, he/she needs to input part ID of the part being sold and his/her salesperson ID. Then the system should check whether that part is available (*Part Available Quantity* > 0). If the part is available, it is then sold and the database is updated accordingly. Finally there should be an informative message on remaining available quantity of the part sold. If the part cannot be sold, an error message should be shown.

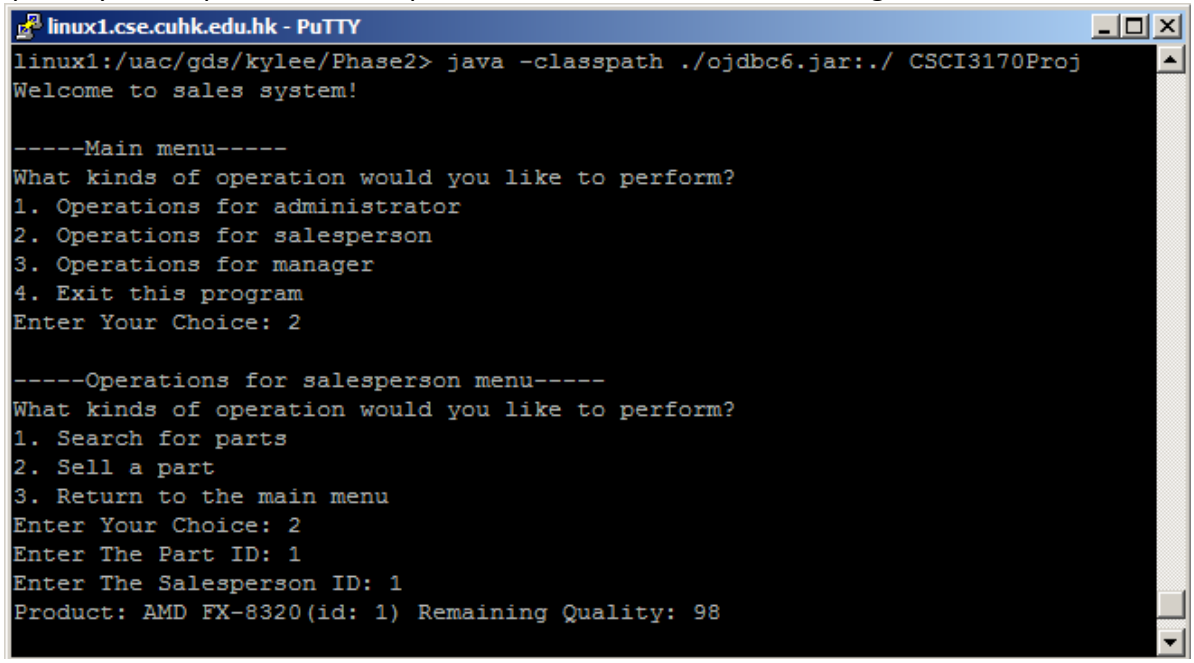

Figure 6: Expected interactive input and output while performing transaction

5.3 Manager

The functions that can be used by a manager are:

- **Show all transaction records of a salesperson within a period:** The system has to provide an interface to allow a manager to show all transaction records within a given period (e.g. from 01/09/2015 to 01/11/2015) of a salesperson with a given *Salesperson ID*. After he/she enters the given *Salesperson ID* and *period*, the program will perform the query and return all the matching transaction records in term of *Transaction ID, Part Name, Manufacturer Name, Part Price and Transaction Date*. These transaction records should be sorted in descending order of *transaction date* and outputted as a table as follows:

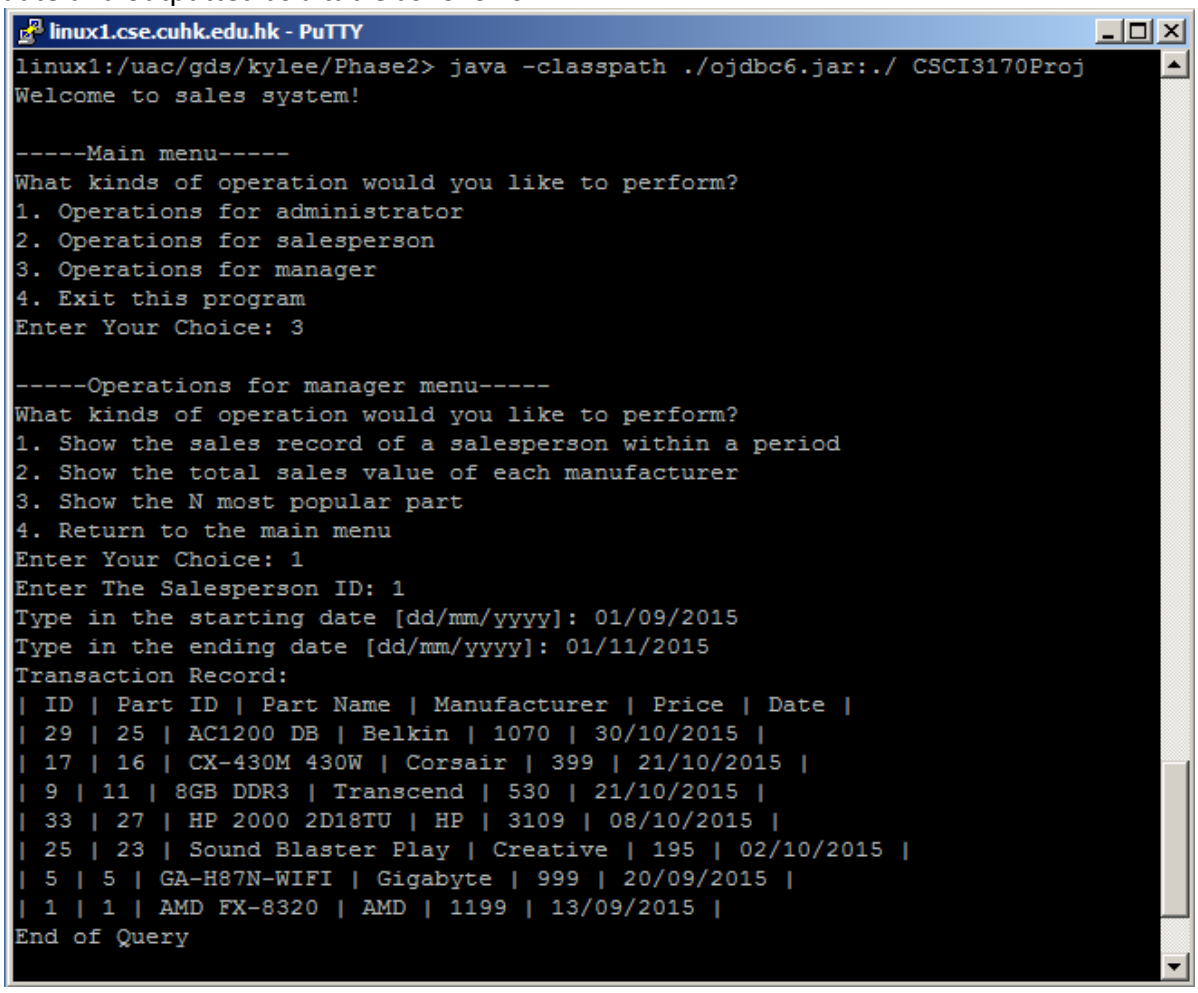

Figure 7: Expected interactive input and output while showing all transaction records of a salesperson within a period (from 01/09/2015 to 01/11/2015).

- **Sort and list the manufacturers in descending order of total sales value:** The system has to provide an interface to allow a manager to sort the manufacturers according to their total sale values. After the program performs the query, it returns the results in terms of *Manufacturer ID*, *Manufacturer Name* and *Total sales value* in descending order of *Total sales value* as a table as follows:

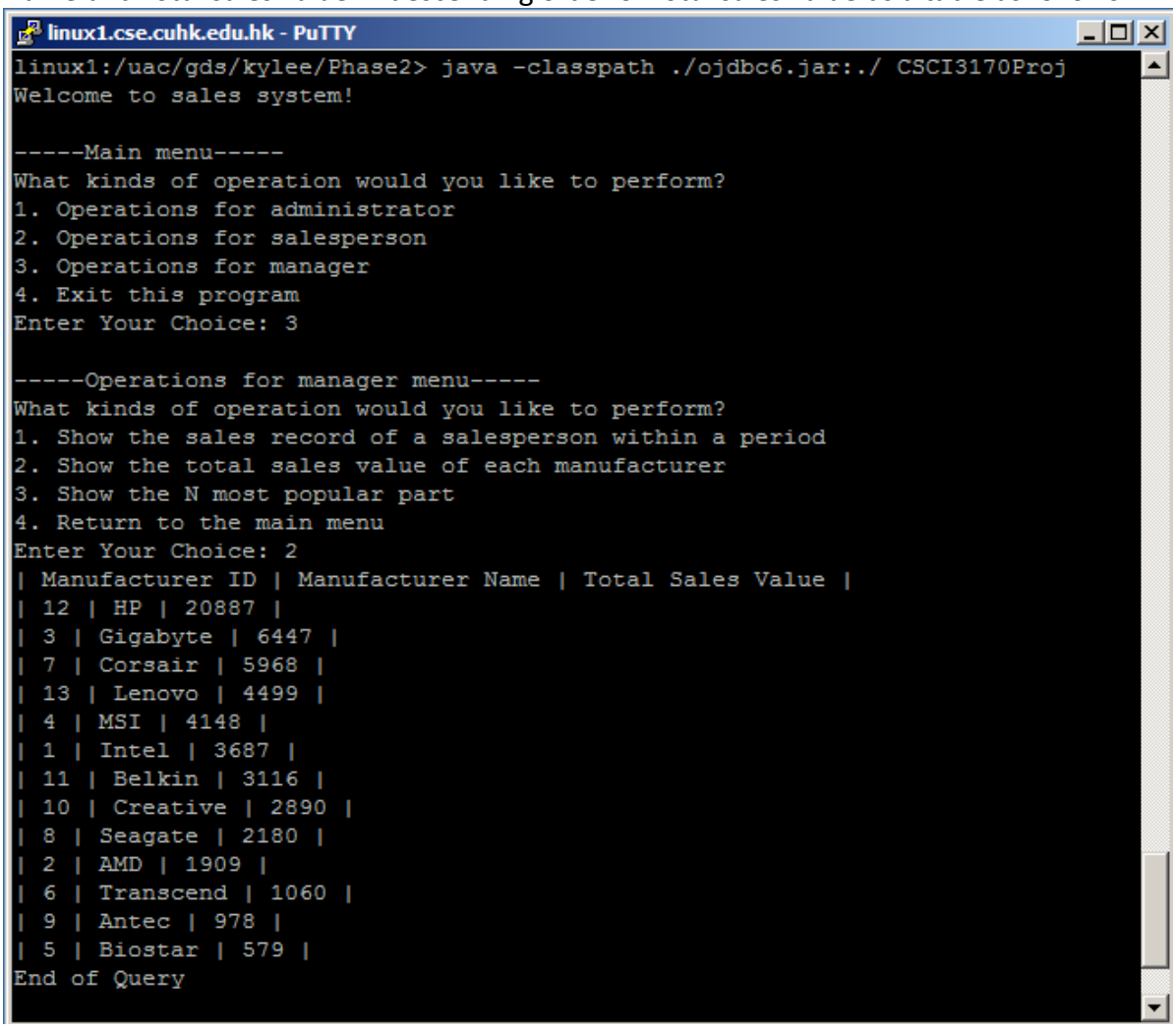

Figure 8: Expected interactive input and output while showing all manufacturers in ascending order of total sales value

- **Show the N most popular parts:** The system has to provide an interface to allow a manager to show the N parts that are most popular. After the manager enters the number of parts (N) that he/she wants to list, the program will perform the query and return the N parts that are most popular in terms of *Part ID*, *Part Name* and *Total Number of Transaction* in descending order of *Total Number of Transaction* as a table as follows.

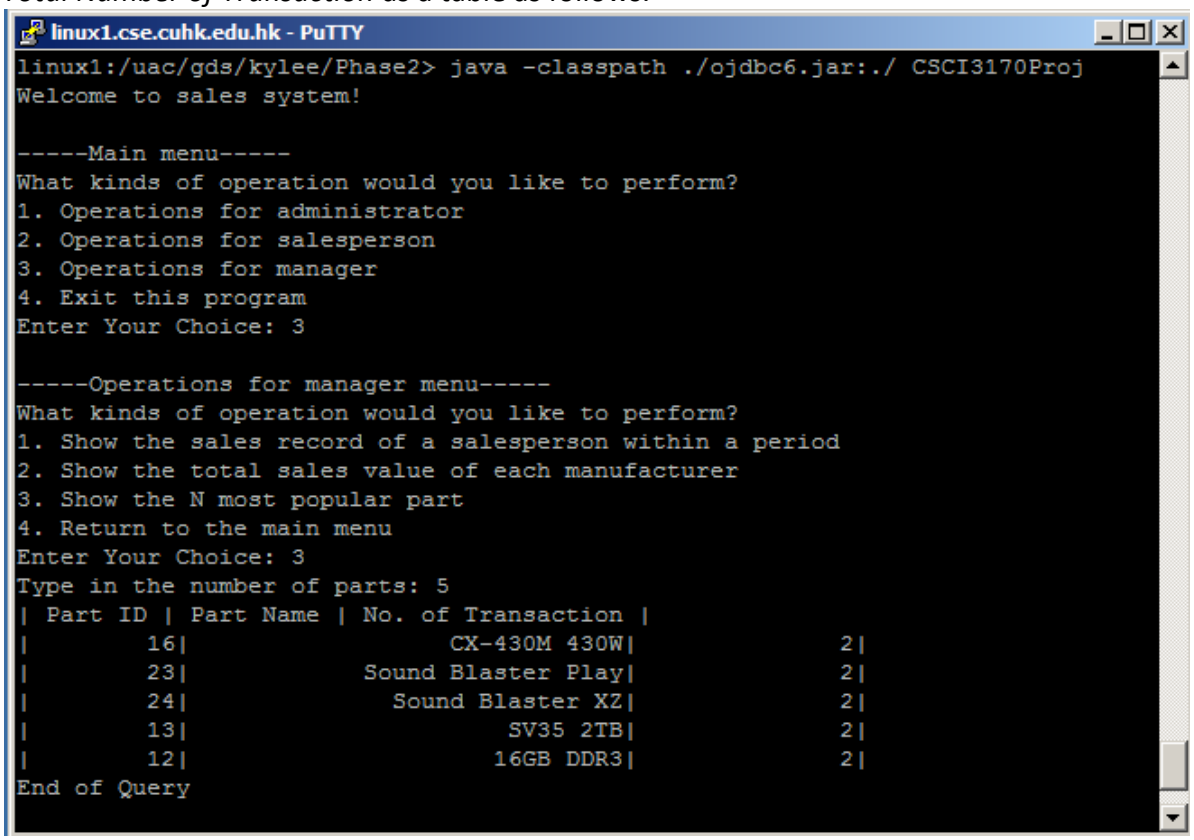

Figure 9: Expected interactive input and output while showing the N most popular parts.

(Note: N should be an integer larger than 0 and a part without any transaction record should not be shown in the table above.)

6 Grading Policy

The marks are distributed as follows:

Phase	Content	Mark Distribution
1	ER-diagram	10%
	Relational schema (based on your ER-diagram)	10%
2	Java application	80%

- There will be a mark deduction if your application is terminated unexpectedly during the demonstration.
- You are not allowed to modify any source code during the demonstration.
- All members in the same group will receive the same marks for the project. In order to encourage every student to participate in the project, a question about this project may be asked in the final examination.

7 Demonstration

- All groups need to sign up for a demonstration on their phase 2 implementation. The registration page would be posted on the course website later.
- All group members should attend the demonstration.
- The duration for the demonstration for each group is about 20 minutes.
- The Java application will be tested in a Linux 64bit machine in the CSE department.
- The dataset used in the demonstration may be different from the dataset provided.

8 Submission Methods

8.1 Phase 1

- Submit a PDF file (one copy for each group) to the collection box at eLearning platform.
- The PDF file should consist of your groups ER diagram, relational schema, the group number, the names and the student IDs of all group members of your group.

8.2 Phase 2

- Submit a ZIP file (one copy for each group) to the collection box at eLearning platform. The ZIP file should consist of all your source codes and a README file (README.txt), which contains:
  - The group number of your group
  - The name and the student ID of each group members of your group
  - List of files with description
  - Methods of compilation and execution