

Project #1 Description

Advanced Programming, 14013103-4

Week #9, Sunday 7 May 2023

(Total: 30 Marks)

Due Date: Week #11, Saturday 20 May 2023

"Products Information System" application

Please follow these instructions:

- This is a group project. So every group must complete it in their own and by themselves.
- **Every group could include two students at maximum.**
- The project must be submitted before the deadline and any late projects will receive **ZERO mark**.
- Copying any deliverables from other groups or the Internet is not allowed. Otherwise, all cheating groups will receive **ZERO marks**.
- Include a report PDF document that includes all the assumptions and several screenshots of your Netbeans IDE application as well as the output screenshot results in cases of successes and errors.
- Submit your Java code as .zip compressed file via the online Blackboard system while **including a backup of the whole MySQL database of the project as well**. Name of the compressed file should include your names, student numbers and group numbers.

The project description:

As a programmer, you have been asked to write a Java console application, using OOP and JDBC concepts, for a **Products Information System** with the following requirements:

-The database name (in MySQL DBMS) should include your last names of the group members **such as: ProductsDB_AIDosary_AlHarby**

-Create a database table using a name that contains the name ProductsTBL and your first names **such as: ProductsTBL_Aly_Ahmed**

-The database table should include the following fields with their data types and constraints:

- ID**: Integer, Not Null, Auto Increment, Primary Key
- Type**: Varchar(20), Not Null (such as: TV, Refrigerator, or PC)
- Model**: Varchar(40), Not Null (such as: Samsung QN90B QLED, LG 645 Liter Side By Side Refrigerator with Multi Air Flow Technology, or HP Envy 14)
- Price**: Float, Not Null
- Count**: Integer, Not Null
- DeliveryDate**: Date, Not Null

-Create a Java Console Project that reads input from the user to add/read all/search/delete products records in the database and display the results to the user. Therefore the application should provide the following menu options to the user in a continuous loop until the user enter the value 4 (i.e. exit):

1- Add a product – To add a new product record including all fields in the corresponding table columns. The application should display a suitable error message (if necessary) and ask the user to type a correct value. If the record is inserted successfully to the database table, present to the user a suitable success message stating that the data was added successfully. The application should present a failure error if the product record was not added for any possible error case.

2- Search for products by model or type – If the user chooses this option, ask the user for the product **model or type** value to search by. The retrieved products records should be presented in a well-organized table form. If no records match the search criteria, present to the user a message stating that there is no records available for this search criteria. Note: the match of the **model or type** should include the search name criteria as any part of the **model or type** values (using LIKE operation and using any letters cases), not exact match.

Note: All records should be displayed if the user provides empty search text.

3- Delete a product by ID – This function should delete the corresponding record to the ID input value provided by the user. The application should only delete the record if the user accepts the confirmation (Y/N) request later on. The application should display a success message upon successful deletion. The application should show unavailable record error message if the user provides non-existing product ID. If the user entered “Y” or “y”, the corresponding record should be removed from the table and a deletion success message should be displayed to the user. If the user entered “N” or “n” the corresponding record should not be removed from the table and a cancelation message should be displayed to the user.

4- Exit – to exit from the application and display a “finishing...” message to the user

Try to make your application as **commented**, organized and modular as possible.

Project deliverables:

- The members of the project group should submit only one .zip file to the blackboard system. Name of the compressed .zip file should include your names, student numbers and group numbers. The .zip file should include the following contents:
 - Include a text file (call it as **names.txt**) that shows the names of the participating students, their student numbers, group numbers and email addresses (**in English**).
 - Include a report PDF document **FILE** (name it as **project1_report.pdf**) that includes several screenshots of your Netbeans IDE application output screenshot results (for all possible success, exception and error cases). Add screenshots of the folder and files structures of the Netbeans project. Provide all assumptions that you have made during design and implementation.
- Note: The program output screenshots should also include details about the group students: IDs, Names, Group Number and Email Addresses on the top of program output.**
- Include the whole Netbeans Java project **FOLDER** with all source codes. Write as much **comments** as possible to document all your source codes.
 - Include a full backup of the database as part of the project submission .zip file.

Hint: to backup your MySql database (schema and data) to one filename.sql file, look at the “5.1 Backup via “mysqldump” Utility Program” section in the MySql RDBMS tutorial at URL: https://www.ntu.edu.sg/home/ehchua/programming/sql/MySQL_HowTo.html
OR URL: <https://blog.winhost.com/using-mysqldump-to-backup-and-restore-your-mysql-databasetables/>