Online queries and their frequencies for the job-shop accounting system:

1. Enter a new customer (30/day).
2. Enter a new department (infrequent).
3. Enter a new process-id and its department together with its type and information relevant to the type (infrequent).
4. Enter a new assembly with its customer-name, assembly-details, assembly-id, and date-ordered and associate it with one or more processes (40/day).
5. Create a new account and associate it with the process, assembly, or department to which it is applicable (10/day).
6. Enter a new job, given its job-no, assembly-id, process-id, and date the job commenced (50/day).
7. At the completion of a job, enter the date it completed and the information relevant to the type of job (50/day).
8. Enter a transaction-no and its sup-cost and update all the costs (details) of the affected accounts by adding sup-cost to their current values of details (50/day).
9. Retrieve the total cost incurred on an assembly-id (200/day).
10. Retrieve the total labor time within a department for jobs completed in the department during a given date (20/day).
11. Retrieve the processes through which a given assembly-id has passed so far (in date-commenced order) and the department responsible for each process (100/day).
12. Retrieve the jobs (together with their type information and assembly-id) completed during a given date in a given department (20/day).
13. Retrieve the customers (in name order) whose category is in a given range (100/day).
14. Delete all cut-jobs whose job-no is in a given range (1/month).
15. Change the color of a given paint job (1/week).

Task 5. Write SQL statements for all queries (1-15) defined in part I. Write a Java application program that uses JDBC and Azure SQL Database to implement all SQL queries (options 1-15), two additional queries for import and export (options 16-17), and the "Quit" option (option 18) as specified in the menu given below. You are free to pick any file format you wish to use for file import and export options. The program will stop execution only when the user chooses the "Quit" option; otherwise, all options must be available for the user to choose at all times. Your program must be commented properly.

WELCOME TO THE JOB-SHOP ACCOUNTING DATABASE SYSTEM

(1) Description of query 1
(2) Description of query 2
.
.
.
(15) Description of query 15
(16) Import: enter new customers from a data file until the file is empty (the user must be asked to enter the input file name).
(17) Export: Retrieve the customers (in name order) whose category is in a given range and output them to a data file instead of screen (the user must be asked to enter the output file name).
(18) Quit

Task 6. Run the program created for Tasks 5 to test its correctness.

- To populate the database, perform 5 queries for each type (1, 2) and 10 queries for each type (3, 4, 5, 6, 7, 8) and show the contents of the affected tables after the 5 queries of each type (1, 2) are completed and after the 10 queries for each type (3, 4, 5, 6, 7, 8) are completed.
- To show database access is possible, perform 3 queries for each type (9, 10, 11, 12, 13, 14, 15).
- To show the import and export facilities are available, run each option (16-17) once.
- To show the Quit option is available, run option (18) at least once.
- To demonstrate that Azure SQL Database can detect errors, you also need to perform 3 queries of different types that contain some errors.

Important Notes for the Java + JDBC + Azure SQL Database Part (Tasks 5 and 6):
Data manipulation and error checking **must be done** by Azure SQL Database. Your program is only to create the menu, accept choices, form queries, submit them to Azure SQL Database for execution, and display results or error messages.

Task 7. Write a Web database application using Azure SQL Database and JSP (or PHP or Python if you are familiar with it) which provides the Web pages for query 1 and query 13. Since both queries take the input data from the user, there should be two Web pages for each query as follows: for query 1, one Web page to allow the user to enter the input data and one to display a message confirming the successful execution of the insertion; and for query 13, there should be one Web page to allow the user to enter the input data and one to display the retrieval results with appropriate headings. To show that your Web application works correctly, run the Web application so that queries 1 and 13 will be executed in this order: first query 13, then query 1, and then query 13 again, making sure that the results of query 1 will change the results of query 13 that follow query 1.