Assigned: 10/8/2025
Group Portion (Max 52.5) Due: 10/18/2025 at 11:59 PM on Canvas
Individual Portion (Max 35) Due: 10/20/2025 at 11:59 PM on Canvas
No late submission of either the group portion or the individual portion will be accepted.
See the submission instructions at the end of the document. Read the "Group Graded Homework Grading Policy" posted on Canvas

PROBLEM 1 (Group Question, No Individual Question)

GQ1 (22.5 points). Write a Java program using JDBC and Azure SQL Database to manage a relational database with the following schema, as provided in Problem 2 of HW 2:
- Passenger (pid: integer, pname: string, tier: string, age: integer)
- Pilot (plid: integer, plname: string, hours: real)
- Flight (fnum: string, origin: string, destination: string, dep_time: string, arrival_time: string, plid: integer)
- Booking (pid: integer, fnum: string)

The program should provide a menu-driven interface to perform the following operations on the Pilot relation:

1. **Insert a Pilot (Hours Based on Average Flight Time)**:
   Insert a new pilot into the Pilot table with attributes plid, plname, and hours. The hours value should be estimated as the average flight time (in hours) of all flights in the Flight table, calculated by subtracting dep_time from arrival_time. If no flights exist, set the hours to 0. Ensure the hours value is at least 0.
2. **Insert a Pilot (Hours Based on Passenger Tier)**:
   Insert a new pilot into the Pilot table with attributes plid, plname, and hours. The hours value should be estimated as the average flight time (in hours) of all flights piloted by pilots who have at least one passenger of a specified tier (e.g., Gold, Silver, or Bronze). If no such flights exist, set the hours to 0. Ensure the hours value is at least 0.
3. **Display All Pilots**:
   Display all records in the Pilot table, showing plid, plname, and hours, in a formatted manner sorted by plid.
4. **Quit**:
   Exit the program.

Requirements:
- The program must use JDBC to connect to an Azure SQL Database.
- The program must terminate only when the user selects Option 4.
- Options 1 and 2 must be implemented as Transact-SQL stored procedures in the Azure SQL Database. The stored procedures must accept plid, plname, and (for Option 2) tier as input parameters.

- The stored procedures must calculate flight time by subtracting dep_time from arrival_time (assuming times are in a format allowing time difference calculation, e.g., HH:MM) and convert to hours.
- For testing, execute Option 3 before and after each execution of Options 1 and 2 to display the Pilot table contents.
- Execute Options 1 and 2 at least three times each with different values for plid, plname, and (for Option 2) tier to demonstrate functionality.
- Execute Option 4 at least once to verify proper program termination.
- The Java program and Transact-SQL stored procedures must include comments explaining the code logic.
- Assume the database tables (Passenger, Pilot, Flight, Booking) are already created and populated as per HW 2, parts GQ3 and GQ4.

PROBLEM 2 (Group Question and Individual Question)

GQ2 (15 Points). Review the SQL file you created for Problem 2 in Graded Homework 2, choose one table that should be indexed, include SQL statement(s) to create an index on that table, and rerun the queries that need to access the table and its index. Provide detailed explanations as to why you chose that table and search key for indexing, whether that index is primary or secondary, and why you chose those queries to rerun.

Individual Question (35 Points)

Each group member will be assigned one version of the following question (focusing on query IQ1, IQ2, IQ3, or IQ4) randomly after the group portion of the homework is due (see submission instructions at the end of the document). You must be prepared to answer any of the versions of the following question individually for the individual portion of the homework:

Consider the case when one of the queries from the list below is executed far more frequently than the other queries from Problem 2 (GQ5) in Graded Homework 2:

- **IQ1)** Find the age of the oldest passenger who is either a Silver-tier member or is booked on a flight piloted by Smith.
- **IQ2)** Find the flight numbers of all flights that either depart from LAX or have five or more passengers booked.
- **IQ3)** Find the names of all passengers who are booked on two flights that depart at the same time.
- **IQ4)** Delete all Bronze-tier passengers (tier = 'Bronze').

Does this new information change your group's answer for **GQ2**? Provide detailed explanations as to why it does, or why it does not. If you argue for a change to the group's answer, make sure to explain which new table and/or which new search key should be used for indexing and whether the new index is primary or secondary. Provide the SQL statement(s) to create the new index. There is no need to execute the statement or any queries on Azure SQL Database.

PROBLEM 3 (Group Question, No Individual Question)

GQ3 (15 points) Given the following relational database table:

VeterinaryClinic (vet_name, license_no, clinic_city, fee_per_visit)
The following insertions are performed on the table VeterinaryClinic:
- `Insert record <Smith, 12, Tulsa, $30>`
- `Insert record <Brown, 45, OKC, $25>`
- `Insert record <Wilson, 23, Norman, $20>`
- `Insert record <Taylor, 78, OKC, $25>`
- `Insert record <Davis, 34, Edmond, $30>`
- `Insert record <Clark, 67, Enid, $35>`
- `Insert record <Lewis, 89, OKC, $25>`
- `Insert record <Walker, 56, Yukon, $30>`
- `Insert record <Harris, 90, Tulsa, $35>`

Assume each block in the VeterinaryClinic file can store up to 3 veterinarian records, and Veterinary Clinic is organized as a sequential file with **vet_name** as the ordering field. Do the following:
1. Show the contents (i.e., the data values as well as the associated block/bucket/record addresses) of the file after the last insertion.
2. Assuming that VeterinaryClinic is an index-sequential file on the search key **vet_name**, show the contents of the dense primary index and the secondary index on **fee_per_visit** after the last insertion.
3. Assuming that a B+-tree index file on **license_no** with order 3 is created for VeterinaryClinic and no two veterinarians have the same **license_no**, using the definition of the B+-tree index given in class, show the content of the B+-tree index file after the last insertion.

SUBMISSION INSTRUCTIONS:

Group Portion:
- The homework answers (one submission per group) must be submitted to **Canvas by 11:59 PM, Saturday, 10/18/2025.** The submission includes the following:
  - Solutions for Problem 1: three files: a) a Java file (extension .java) containing the Java source program; b) an SQL file (extension .sql) containing the Transact SQL Stored Procedures; and c) a PDF file that shows the steps indicating that you have compiled and executed the program successfully (the output must be included). Use the file name convention HW3_Problem1_Group X where X is your group number.
  - Solutions for Problem 2: two files: a) the SQL file that shows the required explanations written as in-line comments, the SQL statement(s) for index creation, and the SQL queries that you chose to rerun; and b) The PDF file that shows the screenshots of the Azure SQL creation of the index and the Azure SQL execution of the SQL queries that you chose to rerun. Use the file name convention HW3_Problem2_Group X where X is your group number;

- Solutions for Problem 3: A single PDF file with typed solution for the Problem 3. No Azure SQL is required to solve Problem 3. Use the file name convention HW3_Problem 3_Group X where X is your group number.
- Attach to your PDF for Problem 1 a cover page that contains the following information:
  GROUP NUMBER: <write your group number here>
  GROUP MEMBERS: <list the names of all members here>
  GRADED HOMEWORK NUMBER: 3
  COURSE: CS/DSA-4513 - DATABASE MANAGEMENT
  SECTION: 002
  SEMESTER: FALL 2025
  INSTRUCTOR: EGAWATI PANJEI
  SCORE: <<We will record the total score of your group for both Problems 1, 2, and 3 here>>

Individual Portion:

- After the submission deadline of the group portion of this graded homework, and before the submission deadline of the individual portion of this graded homework, you will have to take a quiz on Canvas. The quiz will be open from **12:00 AM, Sunday, 10/19/25** to **11:59 PM, Monday,10/20/25**. The quiz will contain one of the versions of the Individual Question for Problem 2. Once you open the quiz, you will have 60 minutes to submit your answer. You will have to upload one PDF document as your answer. The quiz will also ask you for your feedback on your group members (i.e. the scores you give to each of your group members on the group portion of this graded homework) as outlined in the "Group Graded Homework Grading Policy" document available on Canvas. If you do not provide the scores for your group members, then we assume that you give the same points to each of your group members.

NOTES:
- The instructions for using JDBC and Transact-SQL are available on Canvas;
- If you have questions concerning your Azure SQL account, using JDBC, Transact SQL, or other questions concerning this homework, see one of your TAs during their office hours or email them. Their office hours and contact information are in the Announcements page on Canvas.
- Start this project early to avoid last minute system problems. No late submission will be accepted.