

3.10 Summary

- SQL is the most influential commercially marketed relational query language. The SQL language has several parts:
 - **Data-definition language** (DDL), which provides commands for defining relation schemas, deleting relations, and modifying relation schemas.
 - **Data-manipulation language** (DML), which includes a query language and commands to insert tuples into, delete tuples from, and modify tuples in the database.
- The SQL data-definition language is used to create relations with specified schemas. In addition to specifying the names and types of relation attributes, SQL also allows the specification of integrity constraints such as primary-key constraints and foreign-key constraints.
- SQL includes a variety of language constructs for queries on the database. These include the **select**, **from**, and **where** clauses, and support for the natural join operation.
- SQL also provides mechanisms to rename both attributes and relations, and to order query results by sorting on specified attributes.
- SQL supports basic set operations on relations including **union**, **intersect**, and **except**, which correspond to the mathematical set-theory operations \cup , \cap , and $-$.
- SQL handles queries on relations containing null values by adding the truth value “unknown” to the usual truth values of true and false.
- SQL supports aggregation, including the ability to divide a relation into groups, applying aggregation separately on each group. SQL also supports set operations on groups.
- SQL supports nested subqueries in the **where**, and **from** clauses of an outer query. It also supports scalar subqueries, wherever an expression returning a value is permitted.
- SQL provides constructs for updating, inserting, and deleting information.

Review Terms

- Data-definition language
- Data-manipulation language
- Database schema
- Database instance
- Relation schema
- Relation instance
- Primary key
- Foreign key
 - Referencing relation
 - Referenced relation

- Null value
- Query language
- SQL query structure
 - **select** clause
 - **from** clause
 - **where** clause
- Natural join operation
- **as** clause
- **order by** clause
- Correlation name (correlation variable, tuple variable)
- Set operations
 - **union**
 - **intersect**
 - **except**
- Null values
 - Truth value “unknown”
- Aggregate functions
 - **avg**, **min**, **max**, **sum**, **count**
 - **group by**
 - **having**
- Nested subqueries
- Set comparisons
 - { <, <=, >, >= } { **some**, **all** }
 - **exists**
 - **unique**
- **lateral** clause
- **with** clause
- Scalar subquery
- Database modification
 - Deletion
 - Insertion
 - Updating

Practice Exercises

- 3.1 Write the following queries in SQL, using the university schema. (We suggest you actually run these queries on a database, using the sample data that we provide on the Web site of the book, db-book.com. Instructions for setting up a database, and loading sample data, are provided on the above Web site.)
- a. Find the titles of courses in the Comp. Sci. department that have 3 credits.
 - b. Find the IDs of all students who were taught by an instructor named Einstein; make sure there are no duplicates in the result.
 - c. Find the highest salary of any instructor.
 - d. Find all instructors earning the highest salary (there may be more than one with the same salary).
 - e. Find the enrollment of each section that was offered in Autumn 2009.
 - f. Find the maximum enrollment, across all sections, in Autumn 2009.
 - g. Find the sections that had the maximum enrollment in Autumn 2009.