

# Oracle - Database Programming: Grades 11, 12

Adopted 2006

## Number, Date, and Conversion Functions

### 1.1 Define terminology

1. Prepare a list of terms with definitions [1.1.1](#)
- 

### 1.2 Discuss the difference between single row and multiple row subqueries

1. Demonstrate the difference between single row and multiple row subqueries [1.2.1](#)
- 

### 1.3 Identify single row functions that perform case conversions

1. Apply single row functions that perform case conversions [1.3.1](#)
- 

### 1.4 Select and apply character case-manipulation functions LOWER, UPPER, and INITCAP in an SQL query

1. Prepare the code to run in HTML DB [1.4.1](#)
- 

### 1.5 Select and apply character case-manipulation functions CONCAT, SUBSTR, LENGTH, INSTR, LPAD, RPAD, TRIM, and REPLACE in an SQL query

1. Prepare the code to run in HTML DB [1.5.1](#)
- 

### 1.6 Select and apply the single-row number functions ROUND, TRUNC, and MOD in an SQL query

1. Prepare the code to run in HTML DB [1.6.1](#)
- 

### 1.7 Distinguish between the results obtained when TRUNC is applied to a numeric value and ROUND is applied to a numeric value

1. Discuss the results obtained when TRUNC is applied to a numeric value and ROUND is applied to a numeric value [1.7.1](#)
- 

### 1.8 State the implications for business when applying TRUNC and ROUND to numeric values

1. Discuss the difference between TRUNC and ROUND [1.8.1](#)
- 

### 1.9 Analyze and understand IT career options and education requirements based on interests, abilities, aptitudes, and accomplishments

1. Research IT career options and education [1.9.1](#)

---

**1.10 Demonstrate skills for locating, evaluating, and interpreting IT career information**

1. Prepare a chart showing the IT career information [1.10.1](#)
- 

**1.11 Apply concepts learned as a result of student's own work and academic experiences, and evaluate the application of skills to career options and the world of work**

1. Explain the student concepts and evaluate the applications [1.11.1](#)
- 

**1.12 Give an example of an explicit data-type conversion and an implicit data-type conversion**

1. Discuss an example of an explicit data-type conversion and an implicit data-type conversion [1.12.1](#)
- 

**1.13 Explain why it is important, from a business perspective, for a language to have built-in data-conversion capabilities**

1. Prepare an example of why it is important, from a business perspective, for a language to have built-in data-conversion capabilities [1.13.1](#)
- 

**1.14 Construct a SQL-query that correctly applies TO\_CHAR, TO\_NUMBER, and TO\_DATE single row functions to produce a desired result**

1. Prepare the code to run in HTML DB [1.14.1](#)
- 

**1.15 Apply the appropriate date and/or character format model to produce a desired output**

1. Prepare the code to run in HTML DB [1.15.1](#)
- 

**1.16 Explain and apply the use YYYY and RRRR to return the correct year as stored in the database**

1. Write the code to use a DUAL [1.16.1](#)
- 

**1.17 Explain the evaluation of a nested function**

1. Demonstrate the evaluation of a nested function [1.17.1](#)
- 

**1.18 List at least four general functions that work with any data type and relate to handling null values**

1. Create a chart showing the four functions [1.18.1](#)
- 

**1.19 Explain the use of COALESCE and the NVL functions**

1. Demonstrate the use of COALESCE and the NVL functions [1.19.1](#)
- 

**1.20 Construct and execute a SQL query that correctly applies NVL, NVL2, NULLIF, and COALESCE single-row functions**

1. Write the code to use a HTML DB [1.20.1](#)

---

**1.21 Compare and contrast the DECODE and CASE functions**

1. Demonstrate the use of DECODE and CASE functions [1.21.1](#)
- 

**1.22 Construct and execute a SQL query that correctly uses the DECODE and CASE functions**

1. Write the code to use a HTML DB [1.22.1](#)
- 

**1.23 Construct and execute two methods for implementing IF-THEN-ELSE conditional logic**

1. Write the code to use a HTML DB [1.23.1](#)
- 

**Joins****2.1 Define terminology**

1. Prepare a list of terms with definitions [2.1.1](#)
- 

**2.2 Identify the purpose of joins conditions**

1. Demonstrate the purpose of joins conditions [2.2.1](#)
- 

**2.3 Discuss how to construct and execute a SELECT statement that results in a cartesian product**

1. Create a SELECT statement that results in a cartesian product [2.3.1](#)
- 

**2.4 Discuss how to construct and execute a SELECT statement to access data from more than one table using an equijoin**

1. Create a SELECT statement to access data from more than one table using an equijoin [2.4.1](#)
- 

**2.5 Discuss how to construct and execute a SELECT statement that add search conditions using the AND operator**

1. Create a SELECT statements that add search conditions using the AND operator [2.5.1](#)
- 

**2.6 Discuss how to apply the rule for using column aliases in a join statement**

1. Create a column alias in a join statement [2.6.1](#)
- 

**2.7 Discuss why it is it important, from a business perspective, for a language to be able to combine information from multiple data sources**

1. Demonstrate why is it important, from a business perspective, for a language to be able to combine information from multiple data sources [2.7.1](#)
- 

**2.8 Discuss how to construct and execute a SELECT statement to access data from more than one table using a nonequijoin**

1. Create a SELECT statement to access data from more than one table using a nonequijoin [2.8.1](#)

---

**2.9 Discuss how to SELECT and execute a statement to access data from more than one table using an outer join**

1. Create a SELECT statement to access data from more than one table using an outer join [2.9.1](#)
- 

**2.10 Illustrate positive associations between learning and work**

1. Discuss examples of positive associations between learning and work [2.10.1](#)
- 

**2.11 Discuss how to construct and execute a SELECT statement to join a table to itself using a self-join**

1. Create a SELECT statement to join a table to itself using a self-join [2.11.1](#)
- 

**2.12 Identify factors that contribute to the changing nature of work**

1. Discuss examples of factors that contribute to the changing nature of work [2.12.1](#)
- 

**2.13 Illustrate how to incorporate into a career plan skills to stay up-to-date in anticipation of the changing nature of work**

1. Demonstrate how to incorporate into a career plan skills to stay up-to-date in anticipation of the changing nature of work [2.13.1](#)
- 

**2.14 Discuss how to compose and execute a natural join using SQL join syntax**

1. Create a query using SQL join syntax that will execute a natural join [2.14.1](#)
- 

**2.15 Discuss how to create a Cartesian product using SQL join syntax**

1. Create a query using SQL joint syntax that will result in a Cartesian product [2.15.1](#)
- 

**2.16 Discuss the relationship between a cross-join and a Cartesian product**

1. Demonstrate the relationship between a cross-join and a Cartesian product [2.16.1](#)
- 

**2.17 Discuss the relationship between a natural join and an equijoin**

1. Demonstrate the relationship between a natural join and an equijoin [2.17.1](#)
- 

**2.18 Discuss why it is important to have a standard for SQL as defined by ANSI**

1. Demonstrate why it is important to have a standard for SQL as defined by ANSI [2.18.1](#)
- 

**2.19 Discuss how to compose and execute a join with the USING and ON clauses**

1. Create a join with the USING and ON clauses [2.19.1](#)

---

**2.20 Discuss how to compose and execute an ANSI/SO SQL: 1999 query that joins three tables**

1. Create an ANSI/SO SQL: 1999 query that joins three tables [2.20.1](#)

---

**2.21 Name the Oracle proprietary joins and their ANSI/SO SQL: 1999 counterparts**

1. Demonstrate the Oracle proprietary joins and their ANSI/SO SQL: 1999 counterparts [2.21.1](#)

---

**2.22 Compare and contrast an inner and outer join**

1. Demonstrate inner and outer joins [2.22.1](#)

---

**2.23 Discuss how to construct and execute a query to use a left outer join**

1. Create a query that uses a left outer join [2.23.1](#)

---

**2.24 Discuss how to construct and execute a query to use a right outer join**

1. Create a query that uses a right outer join [2.24.1](#)

---

**2.25 Discuss how to construct and execute a query to use a full outer join**

1. Create a query that uses a full outer join [2.25.1](#)

---

**2.26 Construct and execute a query to use an inner join**

1. Write an HTML query that uses an inner join [2.26.1](#)

---

**Group Functions and Subqueries**

**3.1 Define terminology**

1. Prepare a list of terms with definitions [3.1.1](#)

---

**3.2 Explain the seven group functions: AVG, COUNT, MAX, MIN, STDEV, SUM, VARIANCE**

2. Research and give an example of the seven group functions [3.2.2](#)

---

**3.3 Discuss how to construct and execute a SQL query using SELECT, FROM, WHERE, GROUP BY, ORDER BY syntax using group functions**

1. Diagram a SQL query using SELECT, FROM, WHERE, GROUP BY, ORDER BY syntax using group functions [3.3.1](#)

---

**3.4 Discuss how to construct and execute group functions that operate only with numeric data types**

1. Construct and execute group functions that operate only with numeric data types [3.4.1](#)

---

**3.5 Discuss how to construct and execute group functions that operate to produce a single value**

1. Illustrate how to construct and execute a group function that will operate to produce a single value [3.5.1](#)

---

### **3.6 Compare and contrast the results set obtained from single-row functions versus group functions**

1. Discuss the results set obtained from single-row functions versus group functions [3.6.1](#)
- 

### **3.7 Discuss why is it important, from a business perspective, to be able to easily aggregate data (group)**

1. What business problems does this solve? [3.7.1](#)
- 

### **3.8 Discuss how to construct and execute a SQL query applying COUNT, DISTINCT, NVL group functions**

1. Give examples of how to construct a SQL query applying COUNT, DISTINCT, NVL group functions [3.8.1](#)
- 

### **3.9 Differentiate between different kinds of interviews and the purposes of each**

1. Demonstrate the different kinds of interviews and the purpose of each [3.9.1](#)
- 

### **3.10 List specific steps needed to better prepare for a job interview**

1. Create specific steps needed to better prepare for a job interview [3.10.1](#)
- 

### **3.11 Identify the types of questions that cannot be asked during an interview**

1. Research the types of questions that cannot be asked in an interview [3.11.1](#)
- 

### **3.12 Explain the importance of a first impression in the interview process**

1. Discuss the importance of the first impression in the interview process [3.12.1](#)
- 

## **Data Management Language**

### **4.1 Define terminology**

1. Prepare a list of terms with definitions [4.1.1](#)
- 

### **4.2 Construct a MERGE statement**

1. Execute a MERGE statement [4.2.1](#)
- 

### **4.3 List each of the number, character, and date data types**

1. Create a table with examples of the number, character, and date data types [4.3.1](#)
- 

### **4.4 Create a table applying the appropriate data type for each column**

1. Diagram the appropriate uses of each data type [4.4.1](#)
- 

### **4.5 Discuss tables incorporating TIMESTAMP, INTERVAL YEAR TO MONTH, AND INTERVAL DAY TO SECOND data types to columns**

1. Create a table incorporating TIMESTAMP, INTERVAL YEAR TO MONTH, AND INTERVAL DAY TO SECOND data types [4.5.1](#)

---

#### **4.6 Discuss how an organization use time stamps for time zones in business situations**

1. Research a list of examples of how businesses use time stamps for time zones [4.6.1](#)
- 

#### **4.7 Articulate the changing nature of work and its associated educational requirements**

1. Research the changes in work and educational requirements [4.7.1](#)
- 

#### **4.8 Discuss each of the DDL statements: ALTER, DROP, RENAME, and TRUNCATE and discuss the effect each has on tables and columns**

1. Give examples of each DDL statement: ALTER, DROP, RENAME, and TRUNCATE and show the effect each has on tables and columns [4.8.1](#)
- 

#### **4.9 Discuss why it is important to be able to modify a table**

1. Demonstrate the importance of being able to modify tables [4.9.1](#)
- 

#### **4.10 Discuss how to construct a query using the ALTER TABLE commands ADD, MODIFY, and DROP**

1. Execute a query using the ALTER TABLE commands ADD, MODIFY, and DROP [4.10.1](#)
- 

#### **4.11 Discuss the rationale for using TRUNCATE vs. DELETE for tables**

1. Provide a list of differences between TRUNCATE vs. DELETE [4.11.1](#)
- 

#### **4.12 Discuss adding a comment to a table using the COMMENT ON TABLE command**

1. Create a table and add a comment using COMMENT ON TABLE command [4.12.1](#)
- 

#### **4.13 Identify the changes that can and cannot be made to modify a column**

1. Create a list of examples of changes that can and cannot be made to modify a column [4.13.1](#)
- 

#### **4.14 Discuss the guidelines for dropping a column when constraints are present**

1. Create a list of examples of dropping columns when constraints are present [4.14.1](#)
- 

#### **4.15 Discuss when and why the SET UNUSED statement is advantageous**

1. Diagram when and why the SET UNUSED statement is advantageous [4.15.1](#)
- 

#### **4.16 List the guidelines related to using a DROP TABLE statement**

1. Create a chart explaining each guideline [4.16.1](#)
- 

#### **4.17 Discuss the term constraint as it relates to data integrity**

1. Create a list of ways to use constraints [4.17.1](#)

---

**4.18 Discuss NOT NULL and a UNIQUE constraint**

1. Demonstrate NOT NULL and a UNIQUE constraint in a new table [4.18.1](#)

---

**4.19 Identify two reasons why constraints are incorporated into table definitions**

1. Create one example for each reason constraints are incorporated into table definitions [4.19.1](#)

---

**4.20 Discuss why it is important, from a business perspective, for a language to have built-in constraint-checking capability**

1. Research and provide evidence to support the assigned topic [4.20.1](#)
2. Research ways a business can use all of them together [4.20.2](#)

---

**4.21 Evaluate a business problem that would involve a new table with NOT NULL and UNIQUE constraints**

1. Write and display a code for both NOT NULL and UNIQUE constraints [4.21.1](#)

---

**4.22 Discuss PRIMARY KEY, FOREIGN KEY, and CHECK constraints**

1. Create a table using an example of PRIMARY KEY, FOREIGN KEY, and CHECK constraints [4.22.1](#)

---

**4.23 Discuss the purpose of defining PRIMARY KEY, FOREIGN KEY, and CHECK constraints**

1. Explain each term as it is used in a table [4.23.1](#)

---

**4.24 Discuss how to use the CREATE TABLE**

1. Demonstrate the creation of constraints at the column level and table level in a CREATE TABLE statement [4.24.1](#)

---

**4.25 Evaluate a business problem requiring the addition of a PRIMARY KEY and FOREIGN KEY constraint**

1. Write the code to execute the addition of a PRIMARY KEY and FOREIGN KEY; demonstrate them by creating a table [4.25.1](#)

---

**4.26 Differentiate between checking vs. constraint management**

1. Demonstrate the different ways of using constraints [4.26.1](#)

---

**4.27 List three different functions that the ALTER statement can perform on constraints**

1. Demonstrate how each ALTER statement works [4.27.1](#)

---

**4.28 Name a business function that would require a DBA to drop, enable, and/or disable a constraint or use the CASCADE**

1. Identify the function that requires DBA rights [4.28.1](#)

---

**4.29 Evaluate a business problem to modify an existing table with new constraints**

1. Write the code to resolve the problem [4.29.1](#)
- 

**4.30 Discuss the different activities that a database administrator might perform with regard to constraints**

1. Research evidence to support the assigned topic [4.30.1](#)
- 

**Transforming from the Conceptual to the Physical****5.1 Define terminology**

1. Prepare a list of terms with definitions [5.1.1](#)
- 

**5.2 Identify entity relationship models and database models**

1. Distinguish entity relationship models from database models [5.2.1](#)
- 

**5.3 Compare and contrast the conceptual and physical data models**

1. Describe the terminology mapping between a conceptual model and a relational database model [5.3.1](#)
- 

**5.4 Discuss the rule of basic mapping to transform an entity into a table**

1. Apply the rule of basic mapping to transform an entity into a table [5.4.1](#)
- 

**5.5 Recall the rule of Oracle naming conventions for tables and columns used in relational models**

1. Apply the rule of Oracle naming conventions for tables and columns used in relational models [5.5.1](#)
- 

**5.6 Recall the rule of relationship mapping to correctly transform one to many and barred relationships**

1. Apply the rule of relationship mapping to correctly transform one to many and barred relationships [5.6.1](#)
- 

**5.7 Recall the rule of relationship mapping to correctly transform many-to-many relationships**

1. Apply the rule of relationship mapping to correctly transform many-to-many relationships [5.7.1](#)
- 

**5.8 Recall the rule of relationship mapping to correctly transform one-to-one relationships**

1. Apply the rule of relationship mapping to correctly transform one-to-one relationships [5.8.1](#)
- 

**5.9 Recall the rule of relationship mapping to correctly transform relationships in an arc**

1. Apply the rule of relationship mapping to correctly transform relationships in an arc [5.9.1](#)
-

---

**5.10 Recall the table, column, identifiers, relationship, and integrity restraint rules for mapping supertype implementations**

1. State and apply the table, column, identifiers, relationship, and integrity restraint rules for mapping supertype implementations [5.10.1](#)

---

**5.11 Recall the table, column, identifiers, relationship, and integrity restraint rules for mapping subtype implementations**

1. State and apply the table, column, identifiers, relationship, and integrity restraint rules for mapping subtype implementations [5.11.1](#)

---

**5.12 Recall the table, column, identifiers, relationship, and integrity restraint rules for mapping supertype and subtype arc implementations**

1. State and apply the table, column, identifiers, relationship, and integrity restraint rules for mapping supertype and subtype arc implementations [5.12.1](#)

---

**5.13 Discuss how to create a table in HTML DB using a provided SQL script**

1. Create a table in HTML DB using a provided SQL script [5.13.1](#)

---

**5.14 Discuss entering data into an existing table using a provided SQL script**

1. Enter sample data into an existing table using a provided SQL script [5.14.1](#)
2. Query a table to view data using a provided SQL script [5.14.2](#)

---

**Knowledge**

**6.1 Define terminology**

1. Prepare a list of terms with definitions [6.1.1](#)

---

**6.2 Explain Integrity Rule as it relates to database tables**

1. Create a table following the Integrity Rule [6.2.1](#)

---

**6.3 Identify table, row, column, primary key, unique key, and foreign key given in a diagram containing them**

1. Printout a table and identify the rows, columns, UID, FK [6.3.1](#)

---

**6.4 Identify violations of data-integrity rules**

1. Correct violations of data-integrity rules [6.4.1](#)

---

**6.5 Recall the rules of SQL to display all columns of a table**

1. Apply the rules of SQL to display all columns of a table [6.5.1](#)

---

**6.6 Recall the rules of SQL to display a subset of the columns of a table specified by criteria**

1. Apply the rules of SQL to display a subset of the columns of a table specified by criteria [6.6.1](#)

---

**6.7 Explain how to add new data to a table containing 4 columns**

1. Apply the rules of SQL to add new data to a table containing 4 columns [6.7.1](#)
- 

**6.8 Explain how to add a new column to an existing table**

1. Apply the rules of SQL to add a new column to an existing table [6.8.1](#)
- 

**6.9 Discuss applications of DELETE**

1. Use the DELETE and ALTER TABLE [6.9.1](#)
- 

**6.10 Illustrate a data-modeling project**

1. Use a data-modeling project to solve a business information need [6.10.1](#)
- 

**6.11 Discuss solutions to business problems using database technology**

1. Work as a group to solve business problems using database technology [6.11.1](#)
- 

**6.12 Illustrate a database solution to a business problem**

1. Create and present a database solution to a business problem [6.12.1](#)
- 

**6.13 Explain the different stages of the system development lifecycle**

1. List and describe the different stages of the system development lifecycle [6.13.1](#)
- 

**6.14 Explain the role of data modeling in the system development lifecycle**

1. Explain the purpose of data modeling [6.14.1](#)
- 

**6.15 Discuss the relationship between project tasks to the different stages of the system development lifecycle**

1. Correctly apply relationships to List of Values (LOV) in an organizational hierarchy [6.15.1](#)
- 

**6.16 Recall how to implement tables from an ERD**

1. Use HTML DB to implement tables from an ERD [6.16.1](#)
- 

**6.17 Recall how to issue SQL queries in HTML DB**

1. Produce query output using HTML DB [6.17.1](#)
- 

**6.18 Explain the features and benefits that Oracle Database Environment provides for businesses**

1. Identify & describe the efficiency, accuracy, flexibility, workflow, & reporting qualities of Oracle [6.18.1](#)
- 

**6.19 Compare and contrast application software and system software**

1. Identify key differences between application software and system software [6.19.1](#)

---

**6.20 Identify the appropriate SQL functions to perform projection, selection, and join**

1. Use the correct syntax to perform projection, selection, & join [6.20.1](#)
- 

**6.21 Discuss the correct syntax to perform arithmetic expressions on the columns of a query**

1. Use the correct syntax to perform arithmetic expressions on the columns of a query [6.21.1](#)
- 

**6.22 Recall correct operator precedence to display desired results**

1. Formulate queries using correct operator precedence to display desired results [6.22.1](#)
- 

**6.23 Compare and contrast the concepts of null, zero and an empty string**

1. Categorize the concepts of null, zero, and an empty string [6.23.1](#)
- 

**6.24 Recall the effect null values have in arithmetic expressions**

1. Demonstrate the effect null values have in arithmetic expressions [6.24.1](#)
- 

**6.25 Discuss how to use a column alias**

1. Construct a query using a column alias [6.25.1](#)
- 

**6.26 Recall how to use the concatenation operator**

1. Apply the concatenation operator to link column values and expressions to create a character expression [6.26.1](#)
- 

**6.27 Discuss use of literal values of type character, number, and date**

1. Use literal values of type character, number, and date in a SQL SELECT statement [6.27.1](#)
- 

**6.28 Define and use DISTINCT to eliminate duplicates in query results**

1. Apply DISTINCT to eliminate duplicates in query results [6.28.1](#)
- 

**6.29 Discuss the structure of a table using DESCRIBE**

1. Display the structure of a table using DESCRIBE [6.29.1](#)
- 

**6.30 Illustrate the use of HTML DB to run, edit, and save SQL statements**

1. Edit, execute, and save SQL statements in HTML DB [6.30.1](#)
- 

**6.31 Discuss how to access self test software to review for certification exam**

1. Demonstrate the ability to log into the self test software [6.31.1](#)
- 

**6.32 Discuss how to use WHERE clause to restrict rows returned in a SQL query**

1. Apply the WHERE clause to restrict rows returned in a SQL query [6.32.1](#)

---

**6.33 Explain why it is important to be able to easily limit data retrieved from a table**

1. Construct a query that limits or restricts a column or row [6.33.1](#)
- 

**6.34 Explain the use of logical comparisons to restrict the rows returned based on two or more conditions**

1. Evaluate logical comparisons to restrict the rows returned based on two or more conditions [6.34.1](#)
- 

**6.35 Explain the rules of precedence by which expressions are evaluated and calculated**

1. Apply the rules of precedence to determine the order in which expressions are evaluated and calculated [6.35.1](#)
- 

**6.36 Identify a query to sort a result set in ascending or descending order**

1. Construct a query to sort a result set in ascending or descending order [6.36.1](#)
- 

**6.37 Identify a query to order a result set using a column alias**

1. Construct a query to order a result set using a column alias [6.37.1](#)
- 

**6.38 Identify a query to order a result set for single or multiple columns**

1. Construct a query to order a result set for single or multiple columns [6.38.1](#)
- 

**6.39 Identify appropriate applications of single-row functions in query statements**

1. Use single-row functions in a query [6.39.1](#)
- 

**6.40 Identify a function as a single row or multiple row function**

1. Classify a function as a single row or multiple row function [6.40.1](#)
- 

**6.41 Compare and contrast the results returned by single row and multiple row functions**

1. Categorize the results returned by a function as single row or multiple row [6.41.1](#)
- 

**HTML DB**

**7.1 Define terminology**

1. Prepare a list of terms with definitions [7.1.1](#)

---

## **7.2 Discuss using SQL concepts to create a functional database appropriate for a small business**

1. Using SQL you will form a database for a small business [7.2.1](#)
  2. Using SQL form a database for a small business with a new list of values [7.2.2](#)
  3. Using HTML DB SQL input data for the businesses customers, subjects, publishers, and item types [7.2.3](#)
  4. Create and produce reports for customers orders using ORDER in the input form and report page [7.2.4](#)
- 

## **7.3 Discuss how to use Oracle HTML DB SQL Workshop to create table components and layouts using a wizard**

1. Use Oracle's HTML DB SQL to make a table using layouts from a wizard [7.3.1](#)
- 

## **7.4 Use Oracle HTML DB SQL Workshop to create the application's pages, page style, and popup list of values (LOVs) using a wizard**

1. Using Oracle's HTML DB SQL create application pages, different page styles, and a list of values [7.4.1](#)
- 

## **7.5 Use Oracle HTML DB SQL Workshop to create input forms using a wizard**

1. Using Oracle's HTML DB SQL create input forms [7.5.1](#)
- 

## **7.6 Describe how to input data into the CUSTOMERS, SUBJECTS, PUBLISHERS, and ITEM\_TYPES tables using a wizard**

1. Create input forms to report areas for applications such as orders and items [7.6.1](#)
- 

## **7.7 Discuss how to create input forms and report areas for the application's ORDERS and ITEMS pages to enable use of the LOVs for inputting data using a wizard**

1. Format the report pages [7.7.1](#)
- 

## **7.8 Discuss how to create and produce reports about customer orders using the ORDER input form and report page**

1. Create charts to display summary data [7.8.1](#)
- 

## **7.9 Explain how to format report pages**

1. Present to class functional database created along with data, reports, and forms [7.9.1](#)
- 

## **Database Operations**

### **8.1 Define terminology**

1. Prepare a list of terms with definitions [8.1.1](#)
- 

### **8.2 List three advantages of the COMMIT, ROLLBACK, and SAVEPOINT statements**

1. List three advantages using these statements [8.2.1](#)

---

**8.3 Explain why it is important, from a business perspective, to be able to control the flow of transaction processing**

1. Explain the importance of controlling the flow of transaction processing 8.3.1
- 

**8.4 Explain the difference between system security and data security as it relates to a database**

1. Explain differences between system security and data security 8.4.1
- 

**8.5 Discuss why it is important, from a business perspective, to be able to set up user accounts with different types of access permissions**

1. Research and provide evidence to support the assigned topic 8.5.1
- 

**8.6 List 5 system privileges and explain their functions**

1. List five system privileges with function explanation 8.6.1
- 

**8.7 Write a statement to create a user**

1. Write a statement to create a user 8.7.1
- 

**8.8 Write a statement to GRANT privileges such as CREATE SESSION, CREATE TABLE, CREATE SEQUENCE, CREATE VIEW, and CREATE PROCEDURE**

1. Write a statement to grant these privileges 8.8.1
- 

**8.9 Define and explain the advantages of a role**

1. Define with explanation of the advantages of a role 8.9.1
- 

**8.10 Define a database link and explain the object privileges that apply with a remote database**

1. Define database links with explanation of object privileges 8.10.1