Computer Science

Coral S. Schmidt Montilla

#148830

Database Systems

FA 2024 CECS4202-81

Contents

[Part I. Review Question and Exercises from the book Database Systems: Design, Implementation, & Management by Carlos Coronel and Steven Morris (Use Edition 12 that is not so expensive) (50 points). 3](#_Toc180008659)

[Chapter 5. 3](#_Toc180008660)

[Chapter 6. 4](#_Toc180008661)

[Chapter 7. 8](#_Toc180008662)

[Chapter 8. 11](#_Toc180008663)

[Part II: SQL (60 points) 17](#_Toc180008664)

[A. Review Questions (Module Quiz) Chapter 5-8 of the book *A Guide to SQL (2015)*: Odd Review Questions. Copy the question in bold face above the answer.  Do not use bold face for the answers. 17](#_Toc180008665)

[Chapter 5: 17](#_Toc180008666)

[Chapter 6: 19](#_Toc180008667)

[Chapter 7: 21](#_Toc180008668)

[Chapter 8: 23](#_Toc180008669)

[B. Show the complete question, query used, and print the query results. 25](#_Toc180008670)

[Chapter 5: 25](#_Toc180008671)

[Chapter 6: 33](#_Toc180008672)

[Chapter 7: 37](#_Toc180008673)

[Chapter 8: 45](#_Toc180008674)

# Part I. Review Question and Exercises from the book Database Systems: Design, Implementation, & Management by Carlos Coronel and Steven Morris (Use Edition 12 that is not so expensive) (50 points).

## Chapter 5.

Review Questions: 1-6: Page 189

1. **What is an entity supertype, and why is it used?**

An entity supertype is a generic entity type that contains common characteristics shared by one or more subtypes. It avoids redundancy by allowing common attributes and relationships to be defined in the supertype. At the same time, unique characteristics of subtypes are captured in the subtypes. This approach allows for efficient organization and representation of data in a database.

1. **What kinds of data would you store in an entity subtype?**

Entity subtypes store data that is unique to each subtype. For example, in the case of employees, while common attributes like name and hire date are stored in the supertype, a subtype such as a pilot would store specific attributes like flight hours or certifications.

1. **What is a specialization hierarchy?**

A specialization hierarchy is a top-down organizational structure that arranges a higher-level entity supertype and its related subtypes. It depicts the relationships between supertypes and subtypes, often represented by "is-a" relationships, where each subtype inherits common attributes and relationships from its supertype.

1. **What is a subtype discriminator? Give an example of its use.**

A subtype discriminator is an attribute in the supertype that determines to which subtype an instance belongs. For example, the EMP\_TYPE attribute might be used in an employee database to differentiate between pilots, mechanics, and accountants based on the value stored in this attribute.

1. **What is an overlapping subtype? Give an example.**

An overlapping subtype is a situation where an entity instance can belong to more than one subtype. For example, in a university, a person can be both an employee and a student. Therefore, the PERSON supertype might have overlapping subtypes of STUDENT and EMPLOYEE.

1. **What is the difference between partial completeness and total completeness?**

Partial completeness indicates that some instances of the supertype may not belong to any subtype. In contrast, total completeness means that every instance of the supertype must be a member of at least one subtype.

## Chapter 6.

Review Questions: 1-6: Pages 235

1. **What is normalization?**

Normalization is a process used in database design to organize data to reduce redundancies and improve data integrity. It involves breaking down large tables into smaller, well-structured tables by ensuring each table has a clear purpose and adheres to the rules of normal forms.

1. **When is a table in 1NF?**

A table is in first normal form (1NF) when all key attributes are defined, there are no repeating groups, and all attributes depend solely on the primary key.

1. **When is a table in 2NF?**

A table is in second normal form (2NF) when it is already in 1NF, and all non-key attributes are fully dependent on the entire primary key, with no partial dependencies.

1. **When is a table in 3NF?**

A table is in the third normal form (3NF) when it is in 2NF, and there are no transitive dependencies, meaning no non-key attribute depends on another non-key attribute.

1. **When is a table in BCNF?**

A table is in Boyce-Codd Normal Form (BCNF) if it is in 3NF and every determinant is a candidate key. BCNF handles anomalies that might exist in 3NF when a non-key attribute serves as a determinant.

1. **Given the dependency diagram shown in Figure Q6.6, answer Items 6a−6c.**

**A diagram of a diagram

Description automatically generated**

1. **Identify and discuss each of the indicated dependencies.**

Each arrow in the diagram represents a functional dependency. For example, the dependency between C1 and C2 shows that the value of C1 determines the value of C2, indicating a direct functional relationship.

1. **Create a database whose tables are at least in 2NF, showing the dependency diagrams for each table.**

|  |  |  |
| --- | --- | --- |
| # | C1 | C2 |
| 1 | **1** | **A** |
| 2 | **2** | **B** |
| 3 | **3** | **C** |

|  |  |  |
| --- | --- | --- |
| # | C1 | C3 |
| 1 | **1** | **X** |
| 2 | **2** | **Y** |
| 3 | **3** | **Z** |

|  |  |  |
| --- | --- | --- |
| # | C4 | C5 |
| 1 | **P** | **100** |
| 2 | **Q** | **200** |
| 3 | **R** | **300** |

1. **Create a database whose tables are at least in 3NF, showing the dependency diagrams for each table.**

|  |  |  |
| --- | --- | --- |
| # | C1 | C2 |
| 1 | **1** | **A** |
| 2 | **2** | **B** |
| 3 | **3** | **C** |

|  |  |  |
| --- | --- | --- |
| # | C1 | C2 |
| 1 | **1** | **X** |
| 2 | **2** | **Y** |
| 3 | **3** | **Z** |

|  |  |  |
| --- | --- | --- |
| # | C4 | C5 |
| 1 | **P** | **100** |
| 2 | **Q** | **200** |
| 3 | **R** | **300** |

## Chapter 7.

Review Questions: 1-12: Page 306-307

1. **In a SELECT query, what is the difference between a WHERE clause and a HAVING clause?**

The WHERE clause filters records before any groupings are made by the GROUP BY clause, operating on individual rows. In contrast, the HAVING clause filters records after the GROUP BY operation, applying conditions to the grouped rows. This means WHERE cannot be used with aggregate functions, but HAVING can.

1. **Explain why the following command would create an error and what changes could be made to fix the error:**

**SELECT V\_CODE, SUM(P\_QOH) FROM PRODUCT;**

The command produces an error because the V\_CODE column is not part of an aggregate function or the GROUP BY clause. To fix it, you need to add a GROUP BY V\_CODE clause, which will group rows by V\_CODE before applying the aggregate function SUM(P\_QOH).

1. **What type of integrity is enforced when a primary key is declared?**

Declaring a primary key enforces entity integrity. This ensures that each row in a table is uniquely identifiable, and that no part of the primary key can contain null values.

1. **Explain why it might be more appropriate to declare an attribute that contains only digits as a character data type instead of a numeric data type.**

It is more appropriate to use a character data type for digits that do not require arithmetic operations. For example, zip codes or phone numbers are better represented as characters to avoid misinterpretation of leading zeros or non-numerical operations.

1. **What is the difference between a column constraint and a table constraint?**

A column constraint is applied directly to a specific column, while a table constraint applies to the table as a whole. For instance, a UNIQUE constraint on a column ensures its values are unique, while a table-level constraint could enforce that a combination of columns is unique.

1. **What are “referential constraint actions”?**

Referential constraint actions enforce referential integrity between tables, ensuring that a foreign key either has a null value or matches a primary key in the referenced table. Common actions include CASCADE, SET NULL, and RESTRICT.

1. **Rewrite the following WHERE clause without the use of the IN special operator: WHERE V\_STATE IN ('TN', 'FL', 'GA')**

WHERE (V\_STATE = 'TN' OR V\_STATE = 'FL' OR V\_STATE = 'GA')

1. **Explain the difference between an ORDER BY clause and a GROUP BY clause.**

The ORDER BY clause arranges the result set in a specified order, either ascending or descending, based on one or more columns. The GROUP BY clause, on the other hand, groups rows that share a common value into summary rows, typically used with aggregate functions.

1. **Explain why the following two commands produce different results:**

**SELECT DISTINCT COUNT (V\_CODE) FROM PRODUCT;**

**SELECT COUNT (DISTINCT V\_CODE) FROM PRODUCT;**

The first query SELECT DISTINCT COUNT(V\_CODE) applies the DISTINCT modifier to the result of COUNT, meaning it eliminates duplicate counts, whereas the second query SELECT COUNT(DISTINCT V\_CODE) counts only distinct V\_CODE values, producing a different result .

1. **What is the difference between the COUNT aggregate function and the SUM aggregate function?**

The COUNT function tallies the number of non-null values in a column, while SUM adds up all the values in a column. COUNT can work on any data type, but SUM is used only for numeric data.

1. **Explain why it would be preferable to use a DATE data type to store date data instead of a character data type.**

Using a DATE data type ensures that dates are stored in a standard format and allows for efficient use of date functions (e.g., adding or subtracting dates). Storing dates as strings would prevent these operations and may lead to inconsistencies.

1. **What is a recursive join?**

A recursive join occurs when a table is joined to itself. This is often used to represent hierarchical data, such as employee-manager relationships, where each row in the table refers to another row in the same table.

## Chapter 8.

Review Questions: Odd numbers: Page 417-418

1. **What is a CROSS JOIN? Give an example of its syntax.**

A CROSS JOIN produces the Cartesian product of two tables. This means every row from the first table is paired with every row from the second table. The syntax is:

SELECT \* FROM T1 CROSS JOIN T2;

1. **Using tables named T1 and T2, write a query example for each of the three join types you described in Question 2. Assume that T1 and T2 share a common column named C1.**

INNER JOIN:

SELECT T1.C1, T1.column2, T2.column3

FROM T1

INNER JOIN T2 ON T1.C1 = T2.C1;

LEFT JOIN:

SELECT T1.C1, T1.column2, T2.column3

FROM T1

LEFT JOIN T2 ON T1.C1 = T2.C1;

CROSS JOIN:

SELECT \*

FROM T1

CROSS JOIN T2

1. **What are the three types of results that a subquery can return?**

**After running the following UNION query:**

A subquery can return (1) a single value (scalar subquery), (2) a list of values (multi-row subquery), or (3) a table of values (multi-column subquery).

1. **Explain the difference between a regular subquery and a correlated subquery.**

A regular subquery runs once independently, while a correlated subquery references values from the outer query and is executed for each row processed by the outer query.

1. **The relational set operators UNION, INTERSECT, and EXCEPT (MINUS) work properly only when the relations are union-compatible. What does union-compatible mean, and how would you check for this condition?**

Union-compatibility means two relations must have the same number of columns and the columns must have compatible data types. To check for union-compatibility, compare the schema of both relations to ensure they have the same structure.

1. **Suppose you have two tables: EMPLOYEE and EMPLOYEE\_1. The EMPLOYEE table contains the records for three employees: Alice Cordoza, John Cretchakov, and Anne McDonald. The EMPLOYEE\_1 table contains the records for employees John Cretchakov and Mary Chen. Given that information, list the query output for the UNION query.**

SELECT EMP\_LNAME, EMP\_FNAME FROM EMPLOYEE

UNION

SELECT EMP\_LNAME, EMP\_FNAME FROM EMPLOYEE\_1;

The output would be:

Alice Cordoza

John Cretchakov

Anne McDonald

Mary Chen

1. **Given the employee information in Question 11, list the query output for the INTERSECT query.**

SELECT EMP\_LNAME, EMP\_FNAME FROM EMPLOYEE

INTERSECT

SELECT EMP\_LNAME, EMP\_FNAME FROM EMPLOYEE\_1;

The output would be:

John Cretchakov

1. **Why does the order of the operands (tables) matter in an EXCEPT (MINUS) query but not in a UNION query?**

In an EXCEPT (or MINUS) query, the order matters because it subtracts the result set of the second query from the first. Therefore, reversing the order will yield different results. In contrast, a UNION combines the results from both queries and eliminates duplicates, so the order does not affect the final result.

1. **What Oracle function should you use to calculate the number of days between your birth date and the current date?**

In Oracle, the SYSDATE function can be used to get the current date, and subtracting a date from SYSDATE returns the number of days between the two dates. For example:

SELECT SYSDATE - BIRTH\_DATE AS days\_between FROM EMPLOYEE;

1. **What string function should you use to list the first three characters of a company’s EMP\_LNAME values? Give an example using a table named EMPLOYEE. Provide examples for Oracle and SQL Server.**

Oracle:

SELECT SUBSTR(EMP\_LNAME, 1, 3) FROM EMPLOYEE;

SQL Server:

SELECT SUBSTRING(EMP\_LNAME, 1, 3) FROM EMPLOYEE;

|  |  |
| --- | --- |
| EMP\_LNAME | EMP\_FNAME |
| Cretchakov | John |
| Cordoza | Alice |
| McDonald | Anne |
| Chen | Mary |

Output (first three characters of EMP\_LNAME):

Cre

Cor

McD

Che

1. **What is a trigger, and what is its purpose? Give an example.**

A trigger is a special kind of stored procedure in a database that automatically runs when a specified event occurs, such as an INSERT, UPDATE, or DELETE operation on a table. Its purpose is to enforce business rules, maintain data integrity, or audit changes. For example, a trigger can be set up to log any changes to an employee's salary in a separate audit table whenever an update is made to the salary column.

1. **What is embedded SQL and how is it used?**

Embedded SQL refers to SQL commands written directly inside a programming language such as C, Java, or Python. It is used to run SQL queries within a program, allowing the program to interact with the database. Embedded SQL is usually processed by pre-compilers or APIs to allow smooth communication between the application code and the database.

# Part II: SQL (60 points)

## A. Review Questions (Module Quiz) Chapter 5-8 of the book *A Guide to SQL (2015)*: Odd Review Questions. Copy the question in bold face above the answer.  Do not use bold face for the answers.

### Chapter 5:

Odd Numbers: pgs. 168-169

1. **How do you join tables in SQL?**

You can join tables in SQL by listing the columns to display in the SELECT clause, then listing all tables involved in the FROM clause. In the WHERE clause, you specify the condition that restricts the data to be retrieved, which is based on matching values in corresponding columns from the tables being joined.

1. **List two operators that you can use with subqueries as an alternate way of performing joins.**

Two operators you can use with subqueries are IN and EXISTS. These operators can retrieve data from multiple tables without performing an explicit join.

1. **What is an alias? How do you specify an alias in SQL? Why would you use an alias?**

An alias in SQL is an alternate name for a table or column. You specify an alias by following the table or column name with a space and the alias. Aliases simplify queries, particularly when working with complex queries or self-joins. For example, you might alias the CUSTOMER table as C to make the query more concise.

1. **What command would you use to show all rows of two tables? How would you use it? What command would you use to show only common rows between two tables? How would you use it?**

To show all rows from two tables, you would use an OUTER JOIN. For example, a LEFT JOIN retrieves all rows from the first table and matching rows from the second. To show only common rows, you would use an INNER JOIN, which retrieves rows where there is a match between the two tables based on the join condition.

1. **How do you use the ALL operator with a subquery?**

The ALL operator is used with a subquery to ensure that the condition is true for all values returned by the subquery. For example, SELECT \* FROM CUSTOMER WHERE BALANCE > ALL (SELECT BALANCE FROM CUSTOMER WHERE REP\_ID = '10') finds customers whose balance is greater than the balance of all customers represented by a particular sales rep.

1. **Which rows are included in an inner join? What clause can you use to perform an inner join in SQL?**

An inner join includes only rows that have matching values in both tables based on the join condition. You can perform an inner join using the INNER JOIN clause. For example, SELECT \* FROM CUSTOMER INNER JOIN INVOICES ON CUSTOMER.CUST\_ID = INVOICES.CUST\_ID retrieves matching rows from both tables where customer IDs are the same.

1. **Which rows are included in a right outer join? What clause can you use to perform a right outer join in SQL?**

A right outer join includes all rows from the table listed on the right side of the join, and only matching rows from the left table. The RIGHT JOIN clause is used to perform a right outer join. For example, SELECT \* FROM CUSTOMER RIGHT JOIN INVOICES ON CUSTOMER.CUST\_ID = INVOICES.CUST\_ID retrieves all invoices, even if they don’t have a matching customer.

### Chapter 6:

Odd Numbers: pgs. 197-198

1. **Which command creates a new table?**

The CREATE TABLE command is used to define the structure of a new table. This command is followed by the table name and the definitions for the columns and their data types.

1. **How do you add data from an existing table to another table?**

You can add data from an existing table to another table by first creating the new table using CREATE TABLE and then inserting data using the INSERT INTO command with a SELECT statement to specify which data to move from the existing table.

1. **Which command removes rows from a table?**

The DELETE command is used to remove rows from a table. You specify which rows to delete by using a WHERE clause in conjunction with this command.

1. **Which command reverses updates? Which updates are reversed?**

The ROLLBACK command reverses updates that have not been committed. If no COMMIT has been made, ROLLBACK can undo all changes since the last commit.

1. **What is the format of the SET clause that changes the value in a column to null in an UPDATE command?**

To change a value in a column to null, you use the UPDATE command with the SET clause in the format SET column\_name = NULL. If you only want to set specific rows to null, you would include a WHERE clause to filter the rows.

1. **Which command and clause changes the characteristics of an existing column in a table?**

You use the ALTER TABLE command along with the MODIFY clause to change the characteristics of an existing column in a table.

### Chapter 7:

Odd Numbers: pgs. 234-235

1. **What is a view?**

A view is a virtual table that consists of data from one or more tables. It does not store the data physically but allows users to query it as if it were a real table.

1. **What is a defining query?**

A defining query is the SQL command used to create a view. It specifies the rows and columns from the underlying base tables that will appear in the view.

1. **What are three advantages of using views?**

The three advantages of using views are:

Data Independence: Views continue to work even if the underlying database structure changes.

Customization: Different users can view the same data in different ways based on their needs.

Security: Views can restrict access to sensitive data by limiting the columns or rows a user can see.

1. **Which command deletes a view?**

The DROP VIEW command is used to delete a view when it is no longer needed.

1. **Which command terminates previously granted privileges?**

The REVOKE command is used to terminate previously granted privileges. For example, you can revoke a user's SELECT permission with REVOKE SELECT ON table FROM user.

1. **How do you create an index? How do you create a unique index? What is the difference between an index and a unique index?**

You create an index using the CREATE INDEX command, while a unique index is created with CREATE UNIQUE INDEX. The difference is that a unique index enforces a rule that the indexed values must be unique within the column(s).

1. **Does the DBMS or the user make the choice of which index to use to accomplish a given task?**

The DBMS automatically chooses which index to use for a given query to improve performance.

1. **The CUSTOMER table contains a foreign key, REP\_ID, that must match the primary key of the SALES\_REP table. What type of update(s) to the CUSTOMER table would violate the foreign key constraint?**

Any update to the REP\_ID in the CUSTOMER table that does not correspond to an existing REP\_ID in the SALES\_REP table would violate the foreign key constraint.

1. **How is the system catalog updated?**

The system catalog is updated automatically by the DBMS whenever changes are made to the database structure, such as creating or deleting tables, indexes, or views.

1. **How do you specify a general integrity constraint?**

A general integrity constraint is specified using the CHECK clause in a CREATE TABLE or ALTER TABLE statement, ensuring that only valid data is entered into the table.

1. **How do you specify a foreign key in MySQL?**

You specify a foreign key in MySQL using the FOREIGN KEY clause in the CREATE TABLE or ALTER TABLE statement, followed by the column that references the primary key of another table.

### Chapter 8:

Odd Numbers: pg. 283-284

1. **How do you display letters in uppercase in MySQL, Oracle, and SQL Server? How do you display letters in lowercase in MySQL, Oracle, and SQL Server?**

To display letters in uppercase, use the UPPER() function in all three systems (MySQL, Oracle, and SQL Server). For lowercase letters, use the LOWER() function.

1. **How do you add months to a date in MySQL, Oracle, and SQL Server? How do you add days to a date? How would you find the number of days between two dates?**

In MySQL, use DATE\_ADD(date, INTERVAL n MONTH) to add months; in Oracle, use ADD\_MONTHS(date, n); and in SQL Server, use DATEADD(MONTH, n, date). To add days, simply use addition in all three systems. To find the number of days between two dates, use DATEDIFF().

1. **How do you concatenate values in character columns in MySQL, Oracle, and SQL Server?**

In MySQL and SQL Server, use the CONCAT() function, while in Oracle, you can use || for concatenation.

1. **What are stored procedures? What purpose do they serve?**

Stored procedures are precompiled SQL queries that are stored in the database and can be executed repeatedly. They help improve performance, ensure reusability, and centralize business logic in the database.

1. **Where do you declare variables in MySQL and PL/SQL procedures?**

In MySQL, declare variables using the DECLARE statement in the body of the procedure. In PL/SQL, variables are also declared in the body using the same syntax.

1. **How do you place the results of a SELECT command into variables in MySQL and PL/SQL?**

Use the INTO clause after the SELECT statement to assign query results to variables.

1. **How do you use a SELECT command that retrieves more than one row in a procedure?**

When you expect more than one row, you must define a cursor, and then use FETCH to retrieve rows one at a time.

1. **Which command selects the next row in a cursor?**

The FETCH command selects the next row from a cursor.

1. **What are triggers? What purpose do they serve?**

A trigger is a procedure that is automatically executed in response to certain events on a table, such as INSERT, UPDATE, or DELETE. It helps maintain the integrity and automation of the database.

## B. Show the complete question, query used, and print the query results.

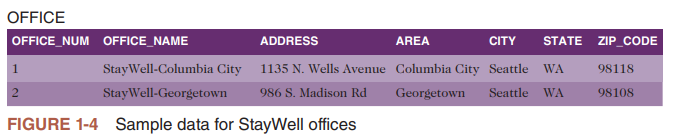
### Chapter 5:

(A Guide to SQL) **Use the Staywell student Accommodation database only.**

Case Exercises: Odd number Case Exercises, pg. 170-171

**Use SQL and the StayWell Student Accommodation database (see Figures 1-4 through 1-9 in Module 1) to complete the following exercises. If directed to do so by your instructor, use the information provided with the Module 3 Exercises to print your output or save it to a document.**

A screenshot of a data table

Description automatically generated

A purple and black box

Description automatically generated A purple and white screen with text

Description automatically generated with medium confidence

A table of names with numbers

Description automatically generated

1. **For every property, list the management office number, address, monthly rent, owner number, owner’s first name, and owner’s last name.**

**A screenshot of a computer

Description automatically generated**

1. **For every service request for furniture replacement, list the property ID, management office number, address, estimated hours, spent hours, owner number, and owner’s last name.**

**A screenshot of a computer

Description automatically generated**

1. **Repeat Exercise 4, (List the first and last names of all owners who own a two-bedroom property. Use the IN operator in your query.) but this time use the EXISTS operator in your query.**

**A screenshot of a computer

Description automatically generated**

1. **List the square footage, owner number, owner last name, and owner first name for each property managed by the Columbia City office.**

**A screenshot of a computer

Description automatically generated**

1. **List the office number, address, and monthly rent for properties whose owners live in Washington state or own two-bedroom properties.**

**A screenshot of a computer

Description automatically generated**

1. **List the office number, address, and monthly rent for properties whose owners live in Washington state but do not own two-bedroom properties.**

**A screenshot of a computer

Description automatically generated**

1. **Find the service ID and property ID for each service request whose estimated hours are greater than the number of estimated hours on every service request on which the category number is 5.**

**A screenshot of a computer

Description automatically generated**

1. **Repeat Exercise 14(List the address, square footage, owner number, service ID, number of estimated hours, and number of spent hours for each service request on which the category number is 4.), but this time be sure each property is included regardless of whether the property currently has any service requests for category 4.** **A screenshot of a computer

   Description automatically generated**

### Chapter 6:

(A Guide to SQL) **Use the Staywell student Accommodation database only.**

Case Exercises: Odd Number Case Exercises, pg. 199-200

**Use SQL to make the following changes to the StayWell Student Accommodation database (Figures 1-4 through 1-9 in Module 1). After each change, execute an appropriate query to show that the change was made correctly. If directed to do so by your instructor, use the information provided with the Module 3 Exercises to print your output or to save it to a document.**

1. **A purple and white rectangular object with black text

   Description automatically generated with medium confidenceCreate a LARGE\_PROPERTY table with the structure shown in Figure 6-29. (Hint: If you have trouble creating the primary key, see Figure 3-36 in Module 3.)**

A screenshot of a computer

Description automatically generated

1. **StayWell has increased the monthly rent of each large property by $150. Update the monthly rents in the LARGE\_PROPERTY table accordingly.**

A screenshot of a computer

Description automatically generated

1. **Insert a row into the LARGE\_PROPERTY table for a new property. The office number is 1, the address is 2643 Lugsi Dr, the number of bedrooms is 3, the number of floors is 2, the monthly rent is $775, and the owner number is MA111.**

A screenshot of a computer

Description automatically generated

1. **The property in managed by Columbia City with the address 105 North Illinois Rd is in the process of being remodeled and the number of bedrooms is unknown. Change the bedrooms value in the LARGE\_PROPERTY table to null.**

A screenshot of a computer

Description automatically generated

1. **Change the OCCUPIED column in the LARGE\_PROPERTY table to N for property ID 9.**

A screenshot of a computer

Description automatically generated

1. **Delete the LARGE\_PROPERTY table from the database.**

A screenshot of a computer

Description automatically generated

### Chapter 7:

(A Guide to SQL) **Use the Staywell student Accommodation database only.**

Case Exercises: Odd Numbers: pgs. 237-238

**Use SQL to make the following changes to StayWell Student Accommodation database (Figures 1-4 through 1-9 in Module 1). After each change, execute an appropriate query to show that the change was made correctly. If directed to do so by your instructor, use the information provided with the Module 3 Exercises to print your output or save it to a document. For any exercises that use commands not supported by your version of SQL, write the command to accomplish the task.**

**1. Create a view named SMALL\_PROPERTY. It consists of the property ID, office number, bedrooms, floor, monthly rent, and owner number for every property whose square footage is less than 1,250 square feet.**

**a. Write and execute the CREATE VIEW command to create the SMALL\_PROPERTY view.** A screenshot of a computer

Description automatically generated

**b. Write and execute the command to retrieve the office number, property ID, and monthly rent for every property in the SMALL\_PROPERTY view with a monthly rent of $1150 or more.**

A screenshot of a computer

Description automatically generated

**c. Write and execute the query that the DBMS actually executes.** A screenshot of a computer

Description automatically generated

**d. Does updating the database through this view create any problems? If so, what are they? If not, why not?**

Updating the database through this view might create problems because views based on multiple tables, or with specific conditions like the one defined for `SMALL\_PROPERTY`, are often not updatable. In this case, Oracle would not allow you to update the `SMALL\_PROPERTY` view directly because it includes a condition based on square footage (`SQR\_FT < 1250`), which restricts the view. Additionally, if there are columns like calculated fields or complex joins, these may prevent updates as well. Therefore, any attempt to update such a view could fail or require special permissions and triggers to handle updates.

**3. Create a view named MONTHLY\_RENTS. It consists of two columns: The first is the number of bedrooms, and the second is the average monthly rent for all properties in the PROPERTY table that have that number of bedrooms. Use AVERAGE\_RENT as the column name for the average monthly rent. Group and order the rows by number of bedrooms.**

**a. Write and execute the CREATE VIEW command to create the MONTHLY\_RENTS view.**

A screenshot of a computer

Description automatically generated

**b. Write and execute the command to retrieve the square footage and average fee for each square footage for which the average fee is greater than $1,100.** A screenshot of a computer

Description automatically generated

**c. Write and execute the query that the DBMS actually executes.** A screenshot of a computer

Description automatically generated

**d. Does updating the database through this view create any problems? If so, what are they? If not, why not?**

Updating the database through the `MONTHLY\_RENTS` view is not possible because this view contains aggregated data, specifically the average monthly rent grouped by the number of bedrooms. Aggregated fields, like `AVG(MONTHLY\_RENT)`, do not directly map back to rows in the underlying table, making it impossible to perform updates, inserts, or deletes through this view. Therefore, any attempt to modify data via this view will result in an error. Only non-aggregated, simple views that directly map to the original table can be updated.

**5. Write, but do not execute, the command to revoke all privileges from user Adams.**

A screenshot of a computer

Description automatically generated

**7. Delete the OWNER\_INDEX 3 index from the OWNER table.**

A screenshot of a computer

Description automatically generated

**9. Add the OWNER\_NUM column as a foreign key in the PROPERTY table.**

A screenshot of a computer

Description automatically generated

### Chapter 8:

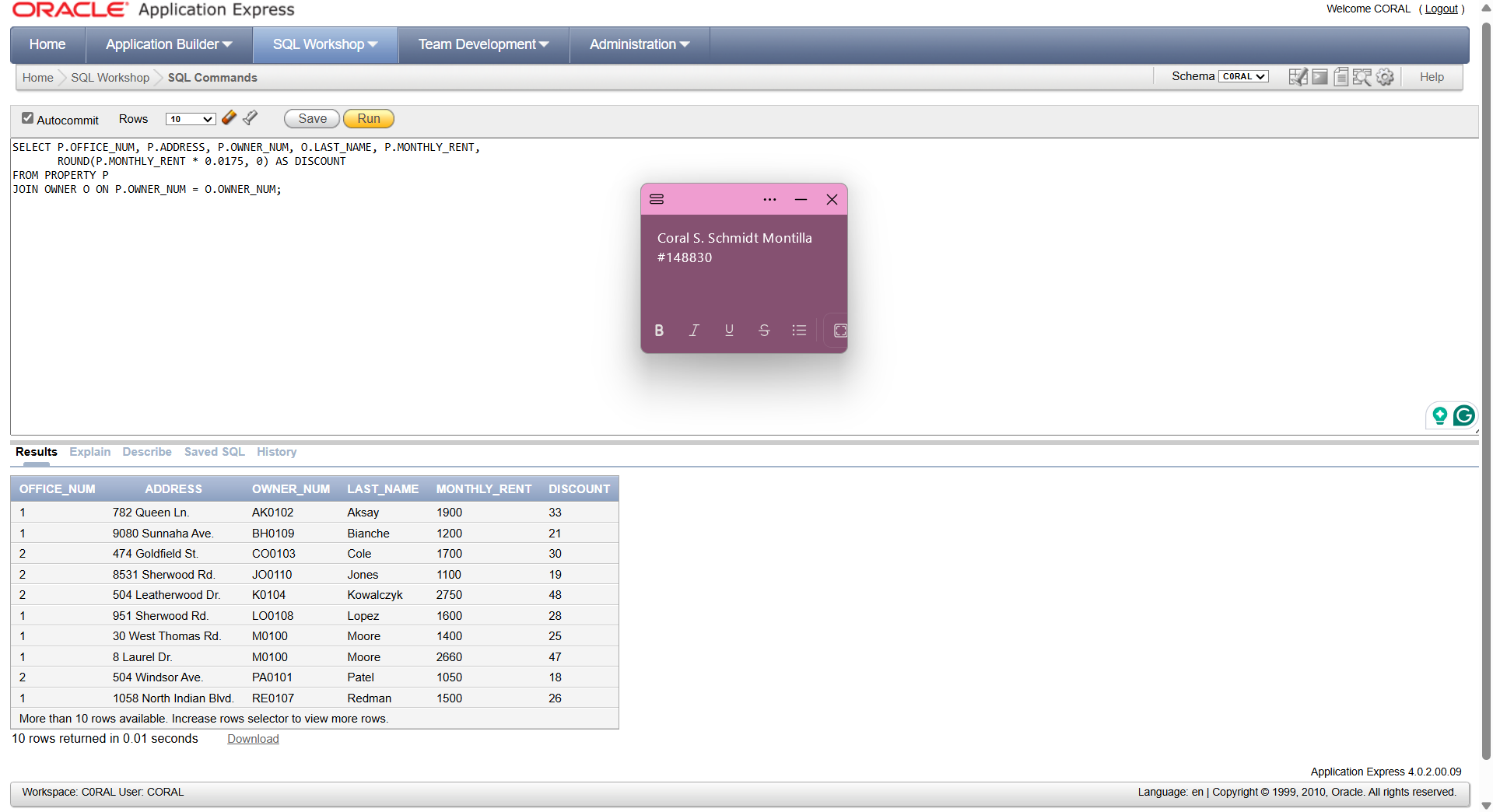
(A Guide to SQL) **Use the Staywell student Accommodation database only.**

Case Exercises: Odd Numbers: pgs. 286-287

**Use the StayWell Accommodation Database (see Figures 1-4 through 1-9 in Module 1) to complete the following exercises. If directed to do so by your instructor, use the information provided with the Module 3 Exercises to print your output or save it to a file.**

**1. List the owner number, first name, and last name for all owners. The first name should appear in uppercase letters and the last name should appear in lowercase letters.** **A screenshot of a computer

Description automatically generated**

**3. StayWell is offering a monthly discount for residents who pay their rent on a quarterly basis. The discount is 1.75 percent of the monthly fee. For each property, list the office number, address, owner number, owner’s last name, monthly rent, and discount. The discount should be rounded to the nearest dollar.** ****

**5. Write PL/SQL or T-SQL procedures to retrieve and output the office number, address, monthly rent, and owner number for every property whose square footage is equal to the square footage stored in I\_SQR\_FT.** **A screenshot of a computer

Description automatically generated**

**7. Write a stored procedure in PL/SQL or T-SQL that changes the monthly rent of a property with a given address and office number. How would you use this stored procedure to change the monthly rent for the property with the address “782 Queen Ln.” and office number 1 to $1,100?**

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated