**EXERCISE 6**

**Question 1: Create a Table for Employee Information**

You are working on a database to store employee information for a company. Create a table named EmployeeInfo with the following columns:

* **employee\_id:** An auto-incremented primary key of type NUMBER.
* **first\_name:** A field of type VARCHAR2(50) to store the first name of the employee. This field should not allow null values.
* **last\_name:** A field of type VARCHAR2(50) to store the last name of the employee. This field should not allow null values.
* **date\_of\_birth:** A field of type DATE to store the date of birth of the employee. This field should not allow null values.
* email: A field of type VARCHAR2(100) to store the email address of the employee.
* **phone\_number:** A field of type VARCHAR2(20) to store the phone number of the employee.
* hire\_date: A field of type DATE to store the date when the employee was hired. This field should not allow null values.
* **department:** A field of type VARCHAR2(50) to store the department or division where the employee works.

**Question 2: Create a Table for Product Inventory**

You are designing a database to manage the inventory of a retail store. Create a table named ProductInventory with the following columns:

* **product\_id:** An auto-incremented primary key of type NUMBER.
* **product\_name:** A field of type VARCHAR2(100) to store the name of the product. This field should not allow null values.
* **description:** A field of type VARCHAR2(200) to store a brief description of the product.
* **price:** A field of type NUMBER with precision 10 and scale 2 to store the price of the product. This field should not allow null values.
* **quantity\_in\_stock:** A field of type NUMBER with precision 5 to store the quantity of the product in stock. This field should not allow null values.
* **manufacturer:** A field of type VARCHAR2(50) to store the name of the manufacturer or supplier.
* **category:** A field of type VARCHAR2(50) to store the category or type of the product.
* **date\_added:** A field of type DATE to store the date when the product was added to the inventory. This field should not allow null values.

**Question 3: Create a Table for Library Books**

You are tasked with creating a table to manage the books in a library's collection. Create a table named LibraryBooks with the following columns:

* **book\_id:** An auto-incremented primary key of type NUMBER.
* **title:** A field of type VARCHAR2(200) to store the title of the book. This field should not allow null values.
* **author:** A field of type VARCHAR2(150) to store the name of the author of the book. This field should not allow null values.
* **publication\_date:** A field of type DATE to store the publication date of the book. This field should not allow null values.
* **isbn:** A field of type VARCHAR2(20) to store the ISBN (International Standard Book Number) of the book.
* **genre:** A field of type VARCHAR2(50) to store the genre or category of the book.
* **available\_copies:** A field of type NUMBER with precision 5 to store the number of copies available in the library.
* **total\_copies:** A field of type NUMBER with precision 5 to store the total number of copies owned by the library.

**Question 4: Create a Table for Customer Orders**

You are building a database for tracking customer orders for an online store. Create a table named CustomerOrders with the following columns:

* **order\_id:** An auto-incremented primary key of type NUMBER.
* **customer\_name:** A field of type VARCHAR2(100) to store the name of the customer. This field should not allow null values.
* **order\_date:** A field of type DATE to store the date when the order was placed. This field should not allow null values.
* **total\_amount:** A field of type NUMBER with precision 10 and scale 2 to store the total amount of the order. This field should not allow null values.
* **shipping\_address:** A field of type VARCHAR2(200) to store the shipping address for the order.
* **payment\_method:** A field of type VARCHAR2(50) to store the payment method used for the order.
* **status:** A field of type VARCHAR2(20) to store the status of the order (e.g., "Processing," "Shipped").
* **tracking\_number:** A field of type VARCHAR2(30) to store the tracking number for shipped orders.

**Question 1: Altering the EmployeeInfo Table**

You are tasked with making structural changes to the EmployeeInfo table. Write SQL queries to perform the following actions:

* Add a new column named address of type VARCHAR2(200) to store employee addresses.
* Modify the email column to allow null values.
* Rename the column phone\_number to contact\_number.
* Delete the department column from the table.
* Add a primary key constraint on the employee\_id column.
* Remove the primary key constraint from the employee\_id column.

**Question 2: Altering the ProductInventory Table**

You are working on the ProductInventory table and need to make structural changes. Write SQL queries to accomplish the following tasks:

* Increase the precision of the price column to allow prices up to $9999.99.
* Add a new column named manufacturer\_location of type VARCHAR2(100) to store the location of the manufacturer.
* Rename the column category to product\_category.
* Set a default value of 0 for the quantity\_in\_stock column.
* Add a primary key constraint on the product\_id column.
* Remove the primary key constraint from the product\_id column.

**Question 3: Altering the LibraryBooks Table**

Your task is to modify the structure of the LibraryBooks table. Write SQL queries to achieve the following:

* Add a new column named language of type VARCHAR2(50) to store the language of the book.
* Modify the isbn column to allow null values.
* Rename the column available\_copies to available\_quantity.
* Delete the total\_copies column from the table.
* Add a primary key constraint on the book\_id column.
* Remove the primary key constraint from the book\_id column.

**Question 4: Altering the CustomerOrders Table**

You are managing the CustomerOrders table and need to make structural adjustments. Write SQL queries to perform the following operations:

* Add a new column named delivery\_date of type DATE to store the expected delivery date of orders.
* Modify the shipping\_address column to allow longer addresses (e.g., VARCHAR2(250)).
* Rename the column payment\_method to payment\_type.
* Set a default value of 'Processing' for the status column.
* Add a primary key constraint on the order\_id column.
* Remove the primary key constraint from the order\_id column

**Update Queries for EmployeeInfo Table:**

* You need to update the email address of an employee with employee\_id 101. Set their email to 'newemail@example.com'
* An employee with employee\_id 203 recently changed their last name to 'Johnson.' Update their last name in the EmployeeInfo table accordingly. Write the SQL query to make this change.
* An employee with employee\_id 305 has been promoted and their salary needs to be increased by $5,000.
* Due to a system error, the hire date of an employee with employee\_id 402 was recorded incorrectly as '2023-06-15' instead of '2023-06-01'. Write an SQL query to correct this hire date.
* You need to assign a new department to an employee with employee\_id 507. Update their department to 'Marketing'.

**Update Queries for ProductInventory Table:**

* The price of a product with product\_id 101 has been increased by 10%. Write an SQL query to update the price accordingly.
* A product with product\_id 205 has been discontinued, and its quantity in stock should be set to 0. Write an SQL query to update the quantity\_in\_stock for this product.
* The manufacturer of a product with product\_id 303 has changed their location. Update the manufacturer\_location to 'New York' for this product.
* Update the category of products with names containing the word 'Electronics' to 'Electrical Appliances'.
* A product with product\_id 408 has been recalled and is no longer available. Update its quantity\_in\_stock to -1 to mark it as unavailable.

**Update Queries for LibraryBooks Table:**

* A book with book\_id 101 has received a new edition, and its title needs to be updated to 'The New Book Title'. Write an SQL query to update the book's title.
* Correct the publication date of a book with book\_id 203, which was mistakenly recorded as '2021-05-15' instead of '2021-05-01'.
* Update the genre of all books published before the year 2000 to 'Classics'.
* The total copies of a book with book\_id 305 should be increased by 5 due to high demand. Write an SQL query to update the total\_copies accordingly.
* A book with book\_id 402 has been removed from the library's collection. Set its available\_quantity to 0 to mark it as unavailable.

**Update Queries for CustomerOrders Table:**

* An order with order\_id 101 has a new shipping address due to a customer's recent move. Update the shipping\_address for this order.
* Change the payment type of an order with order\_id 203 from 'Credit Card' to 'PayPal'.
* An order with order\_id 303 is delayed and will be delivered two days later. Update its delivery\_date accordingly.
* Update the total amount of an order with order\_id 405 to $175.50, reflecting a change in the order's contents.
* An order with order\_id 501 has been canceled. Change its status to 'Canceled' and remove the tracking number.

**Delete Queries for EmployeeInfo Table:**

* You need to remove an employee with employee\_id 101 who has left the company. Write an SQL query to delete this employee's record from the EmployeeInfo table.
* Delete all employees with a hire\_date before '2020-01-01' who are no longer with the company. Write an SQL query to remove these records.

**Delete Queries for ProductInventory Table:**

* A product with product\_id 201 is discontinued and should be removed from the inventory. Write an SQL query to delete this product's record from the ProductInventory table.
* Delete all products with a price greater than $500.00 that are no longer in stock (quantity\_in\_stock = 0). Write an SQL query to remove these records.

**Delete Queries for LibraryBooks Table:**

* A book with book\_id 102 has been permanently removed from the library's collection. Write an SQL query to delete this book's record from the LibraryBooks table.
* Remove all books published before the year 1990 that have less than 5 available copies. Write an SQL query to remove these records.

**Delete Queries for CustomerOrders Table:**

* An order with order\_id 301 was mistakenly duplicated in the system and needs to be deleted. Write an SQL query to remove one of the duplicate orders.
* Delete all orders with a total\_amount less than $50.00 that are in 'Canceled' status. Write an SQL query to remove these records.