Bookshop Management System

The Book Corner

K.F.RISHADHA	MADSE221F-007
A.A.NIZAR	MADSE221F-009
A.M.M.LAKSHANI	MADSE221F-029
K.L.S. DILSHAN	MADSE221F-040

ABSTRACT

This report presents an abstract on the implementation of an Oracle database for The Book Corner, a real-world independent bookstore. The database consists of two related tables aimed at enhancing the bookstore's management processes. Additionally, the report showcases the implementation of specific queries using Oracle PL/SQL to retrieve relevant information from the database. The abstract begins by describing the two tables created in the Oracle database for The Book Corner. The first table, "Books," stores information about the books available in the store, including attributes such as book ID, title, author, genre, and price. The second table, "Customers," captures customer data, including customer ID, name, contact information, and purchase history. The relationship between these tables is established through a foreign key constraint, where the "customer ID" attribute in the "Customers" table references the corresponding primary key in the "Books" table.

The report then proceeds to outline the PL/SQL queries implemented for efficient bookstore management. These queries cover various aspects of the business, such as inventory management, customer analysis, and sales reporting. Examples of implemented queries include retrieving the total number of books in stock, identifying customers who have made a specific purchase, calculating the total revenue generated from book sales, and generating a list of the top-selling books within a given time frame. To demonstrate the effectiveness of these queries, the report includes screenshots showcasing the PL/SQL code and the corresponding output obtained from the Oracle database. The screenshots illustrate how the implemented queries provide valuable insights into inventory levels, customer behavior, and sales performance, enabling The Book Corner's management team to make informed decisions and optimize their operations.

In conclusion, this report highlights the successful implementation of an Oracle database for The Book Corner, featuring two related tables and a range of PL/SQL queries tailored to address key aspects of bookstore management. By leveraging the power of the database and the flexibility of PL/SQL, The Book Corner can efficiently manage its inventory, analyze customer data, and generate meaningful reports to drive business growth and enhance customer satisfaction.

TABLE OF CONTENTS

TABLE OF FIGURES

1.1 INTRODUCTION

In this report, we will explore The Book Corner, a real-world bookshop that our group has selected for the task of creating an Oracle database and implementing various queries using PL/SQL.The Book Corner is a renowned and beloved bookshop located in the heart of a vibrant city. With a passion for literature and a commitment to promoting the joy of reading, The Book Corner has become a go-to destination for book lovers of all ages.

The primary goal of The Book Corner is to provide a curated selection of books that caters to the diverse interests and preferences of its customers. From classic novels to contemporary bestsellers, non-fiction to poetry, and children's books to self-help guides, The Book Corner offers a comprehensive range of titles across various genres.

The bookshop prides itself on creating a warm and inviting atmosphere where customers can browse through the shelves, discover new authors, and immerse themselves in the magical world of books. The friendly and knowledgeable staff at The Book Corner is always available to provide personalized recommendations, engage in literary discussions, and assist customers in finding the perfect book.

In addition to offering a wide selection of books, The Book Corner organizes various literary events and activities to foster a sense of community among book enthusiasts. These events include author signings, book club meetings, poetry readings, and workshops. The Book Corner firmly believes in the power of books to inspire, educate, and connect people, and these events provide opportunities for readers to engage with their favorite authors and fellow book lovers.

To support its operations and enhance the efficiency of managing inventory, customer data, and transactions, The Book Corner has implemented an Oracle database system. The Oracle database enables them to store, retrieve, and analyze critical information related to books, publishers, customer details, sales, and more. By implementing primary key and foreign key constraints, The Book Corner ensures data integrity and establishes relationships between different tables within the database.

Throughout this report, we will demonstrate the implementation of various PL/SQL queries on The Book Corner's Oracle database. These queries will cover areas such as retrieving book information, analyzing sales data, generating reports, and managing customer records. By showcasing these examples, we will highlight how the Oracle database and PL/SQL queries contribute to the smooth operations and data management at The Book Corner.

By the end of this report, we aim to provide a comprehensive understanding of The Book Corner, its business context, and the significance of an Oracle database in supporting its bookshop operations. The insights gained from this exploration will underscore the importance of technology in the book retail industry and how it enables The Book Corner to offer an exceptional reading experience to its customers.

Let us now delve into the world of The Book Corner and discover how the implementation of an Oracle database and PL/SQL queries contribute to the success and growth of this beloved bookshop.

Chapter 2: Query and Outputs

2.1 Creating Tables for the Bookshop

Question 01

Create tables with primary key and foreign key constraints having auto-increment sequence for one of the tables.

Query:

```
 -- Create Authors table
CREATE TABLE Authors (
author_id VARCHAR2(10) PRIMARY KEY,
author_name VARCHAR2(100),
author_birthdate DATE,
author_country VARCHAR2(100)
);

-- Create sequence for Authors table
CREATE SEQUENCE seq_authors
START WITH 00001
INCREMENT BY 1
NOMAXVALUE
NOCACHE
NOCYCLE;
```

Figure 1: Create Authors Table

```
-- Create Publishers table
CREATE TABLE Publishers (
  publisher_id VARCHAR2 (10) PRIMARY KEY,
  publisher_name VARCHAR2 (100),
  publisher_location VARCHAR2 (100),
  publisher_contact VARCHAR2 (100)
);
-- Create sequence for Publishers table
CREATE SEQUENCE seq_publishers
  START WITH 00001
  INCREMENT BY 1
  NOMAXVALUE
  NOCACHE
  NOCYCLE;

-- Create Books table
CREATE TABLE Books (
  book_id VARCHAR2 (10) PRIMARY KEY,
  book_title VARCHAR2 (100),
  book_genre VARCHAR2 (100),
  book_publication_date DATE,
  book_price NUMBER,
  publisher_id VARCHAR2 (10),
  author_id VARCHAR2 (10),
  CONSTRAINT fk_books_publishers FOREIGN KEY (publisher_id) REFERENCES
Publishers(publisher_id),
  CONSTRAINT fk_books_authors FOREIGN KEY (author_id) REFERENCES
Authors(author_id)
);
```

Figure 2: Create Publishers and Books Table

```
 -- Create sequence for Books table
 CREATE SEQUENCE seq_books
START WITH 00001
INCREMENT BY 1
NOMAXVALUE
NOCACHE
NOCYCLE;

 -- Create Customers table
CREATE TABLE Customers (
customer_id VARCHAR2(10) PRIMARY KEY,
customer_name VARCHAR2 (100),
customer_email VARCHAR2 (100),
customer_address VARCHAR2 (100),
customer_phone VARCHAR2 (100)
);

 -- Create sequence for Customers table
 CREATE SEQUENCE seq_customers
 START WITH 00001
 INCREMENT BY 1
 NOMAXVALUE
 NOCACHE
 NOCYCLE;
```

Figure 3: Create Customer's Table and Sequence for Customer

```
 -- Create Orders table
 CREATE TABLE Orders (
 order_id VARCHAR2(10) PRIMARY KEY,
 customer_id VARCHAR2(10),
 order_date DATE,
 total_amount NUMBER,
 CONSTRAINT fk_orders_customers FOREIGN KEY (customer_id) REFERENCES
Customers(customer_id)
 );
 -- Create sequence for Orders table
 CREATE SEQUENCE seq_orders
 START WITH 00001
 INCREMENT BY 1
 NOMAXVALUE
 NOCACHE
 NOCYCLE;
 -- Create Order_Items table
 CREATE TABLE Order_Items (
 order_item_id VARCHAR2(10) PRIMARY KEY,
 order_id VARCHAR2(10),
 book_id VARCHAR2(10),
 quantity NUMBER,
 item_price NUMBER,
 CONSTRAINT fk_order_items_orders FOREIGN KEY (order_id) REFERENCES
Orders(order_id),
 CONSTRAINT fk_order_items_books FOREIGN KEY (book_id) REFERENCES
Books(book_id)
);
```

Figure 4: Create Orders table and Sequence for Orders

```
-- Create sequence for Order_Items table
CREATE SEQUENCE seq_order_items
  START WITH 00001
  INCREMENT BY 1
  NOMAXVALUE
  NOCACHE
  NOCYCLE;

-- Create Reviews table
CREATE TABLE Reviews (
  review_id VARCHAR2(10) PRIMARY KEY,
  book_id VARCHAR2(10),
  customer_id VARCHAR2(10),
  review_text VARCHAR2(1000),
  review_date DATE,
  rating NUMBER,
  CONSTRAINT fk_reviews_books FOREIGN KEY (book_id) REFERENCES Books(book_id),
  CONSTRAINT fk_reviews_customers FOREIGN KEY (customer_id) REFERENCES
Customers(customer_id)
);

-- Create sequence for Reviews table
CREATE SEQUENCE seq_reviews
  START WITH 00001
  INCREMENT BY 1
  NOMAXVALUE
  NOCACHE
  NOCYCLE;
```

Figure 5: Create Reviews Table and Sequence for reveiws Table

Outputs:

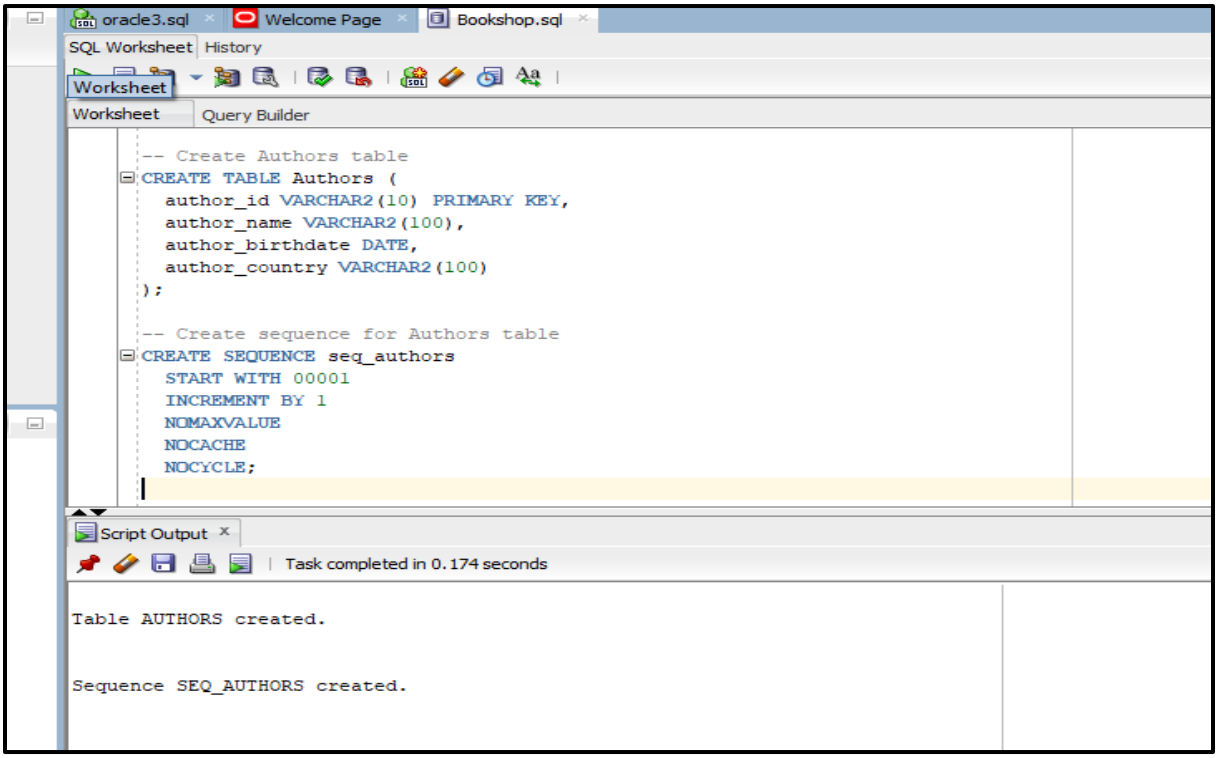


Figure 6: Output of the created Authors Table

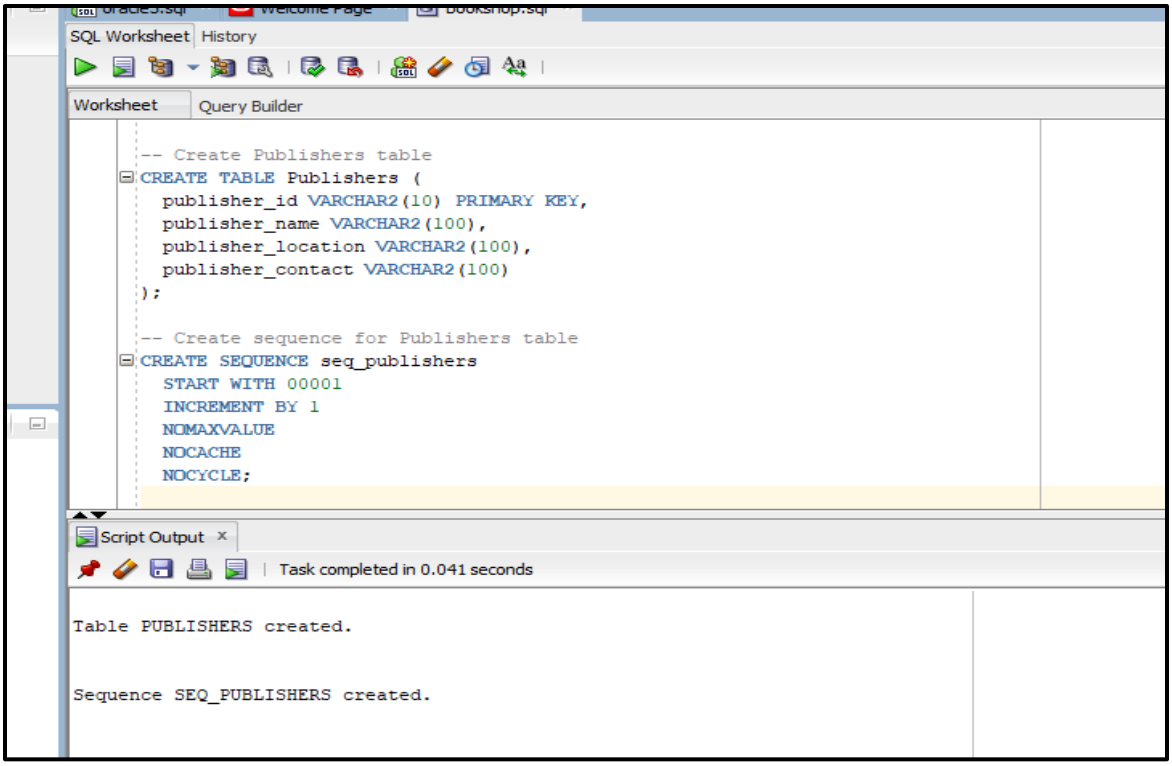


Figure 7: Output of the created Publishers Table

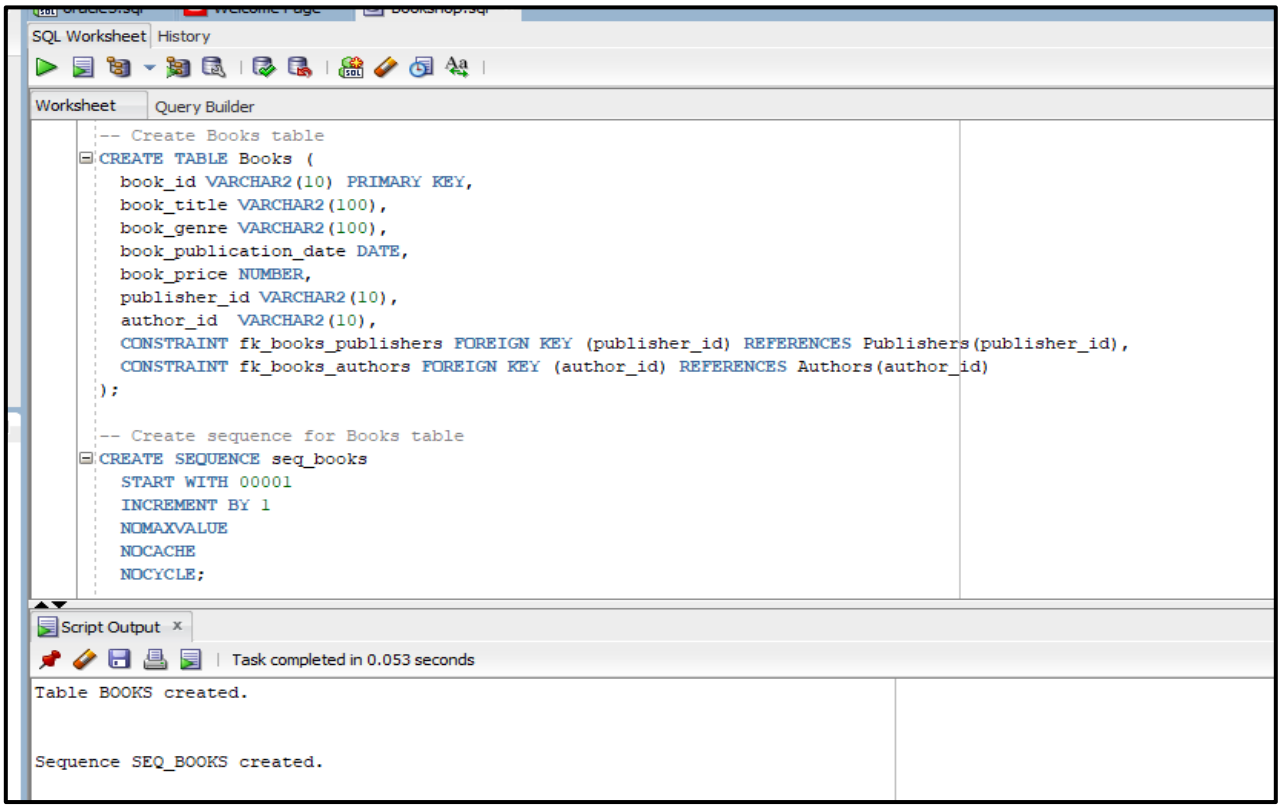


Figure 8: Output of the Created Books Table

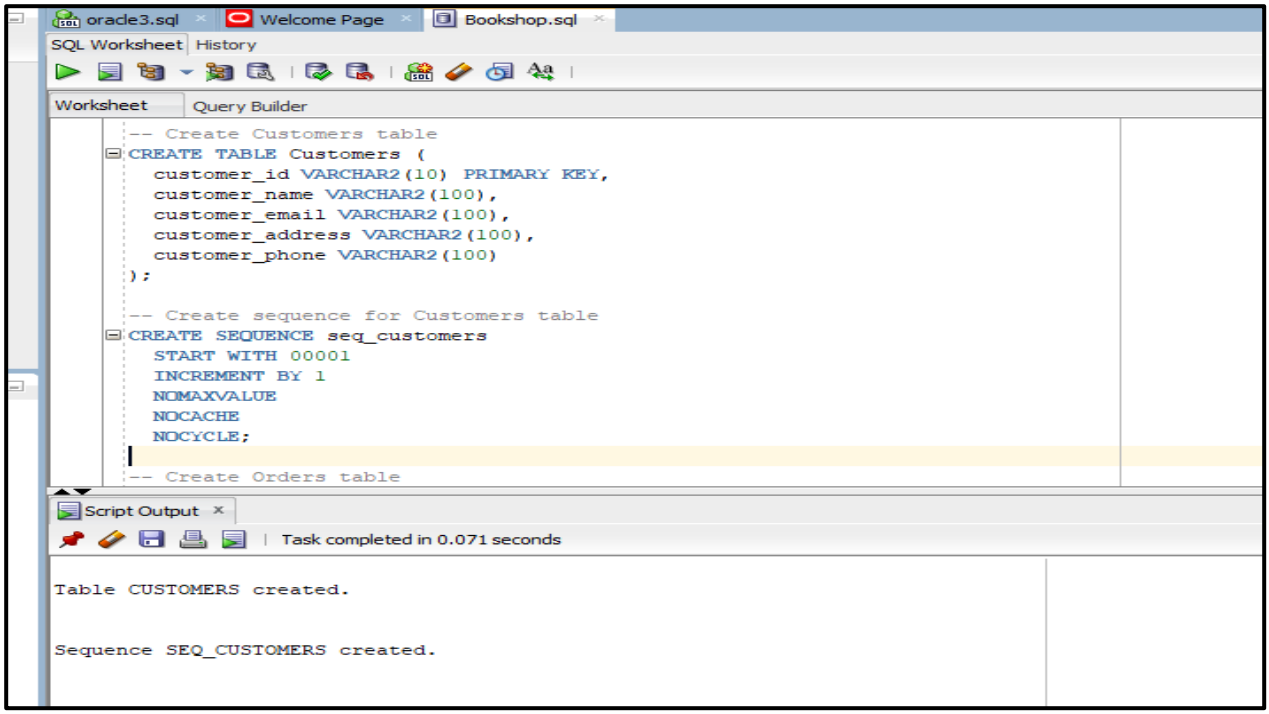


Figure 9: Output of the Created Customer's Table

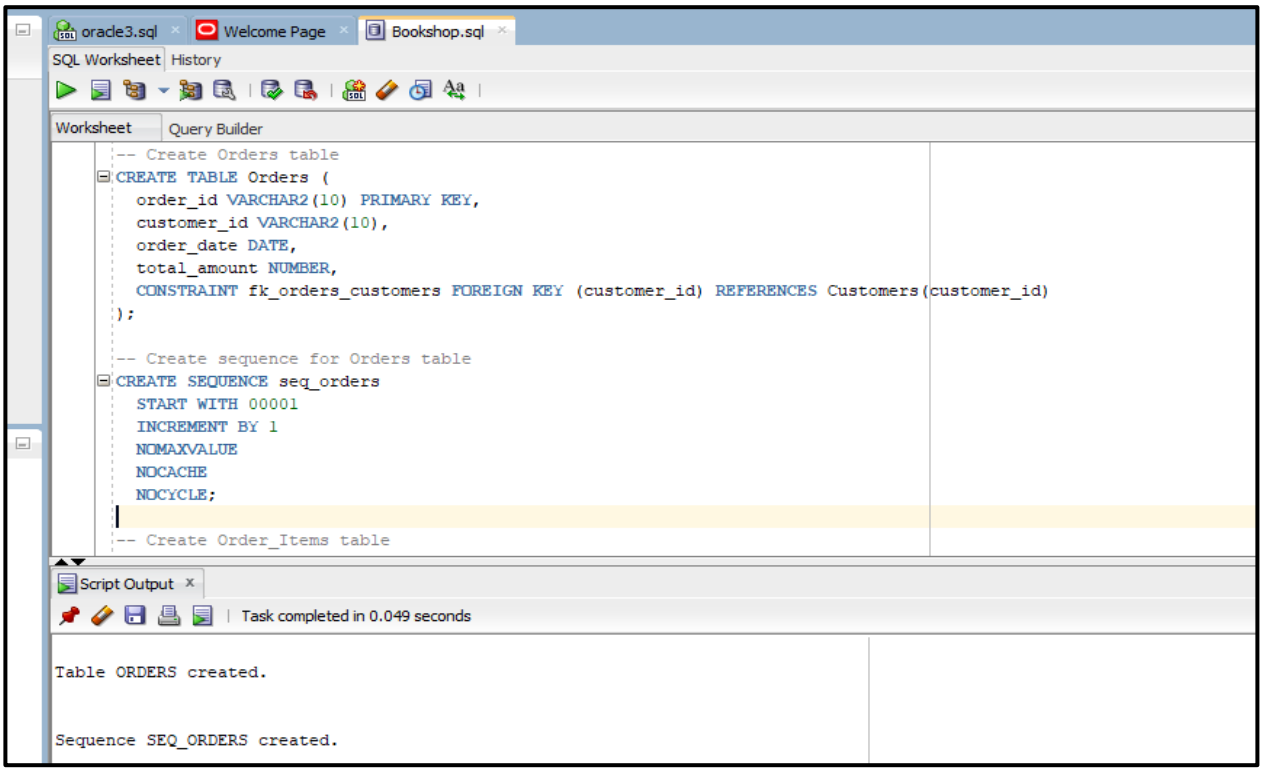


Figure 10: Output of the Created Orders Table

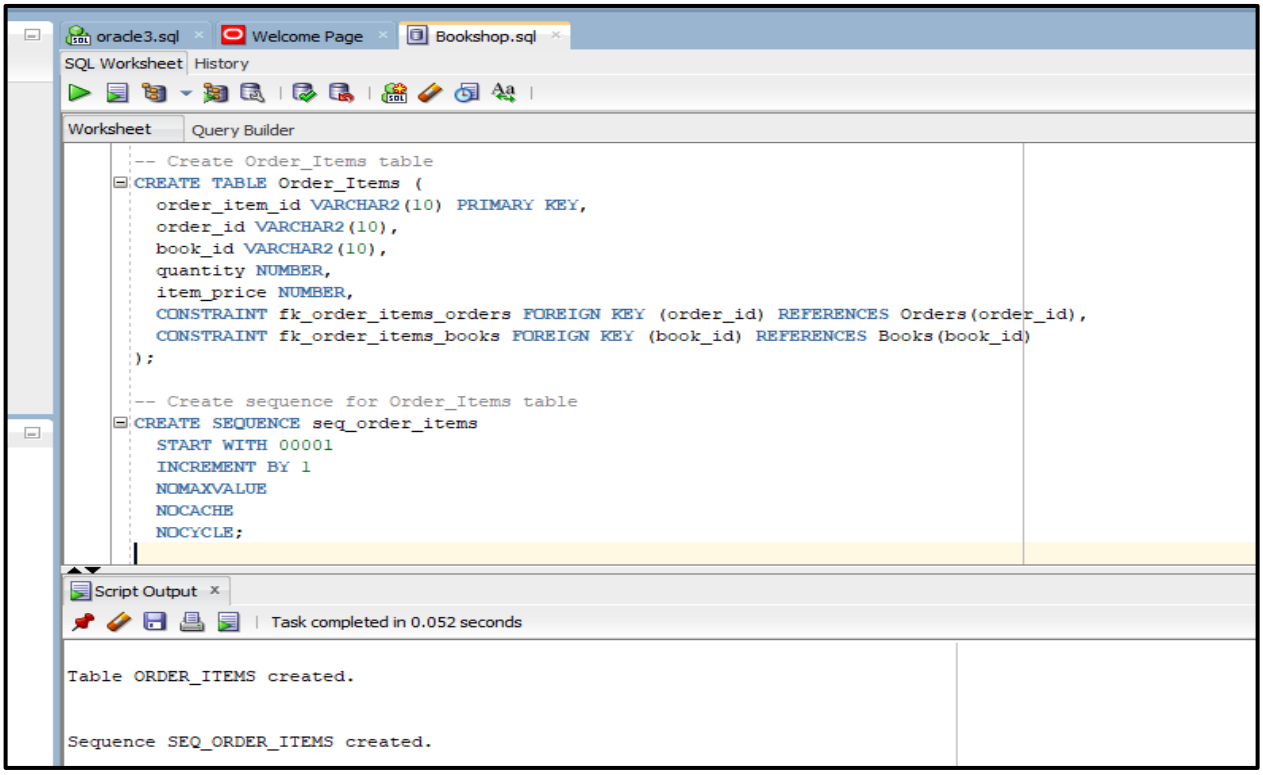


Figure 11: Output of the Created Order_Items Table

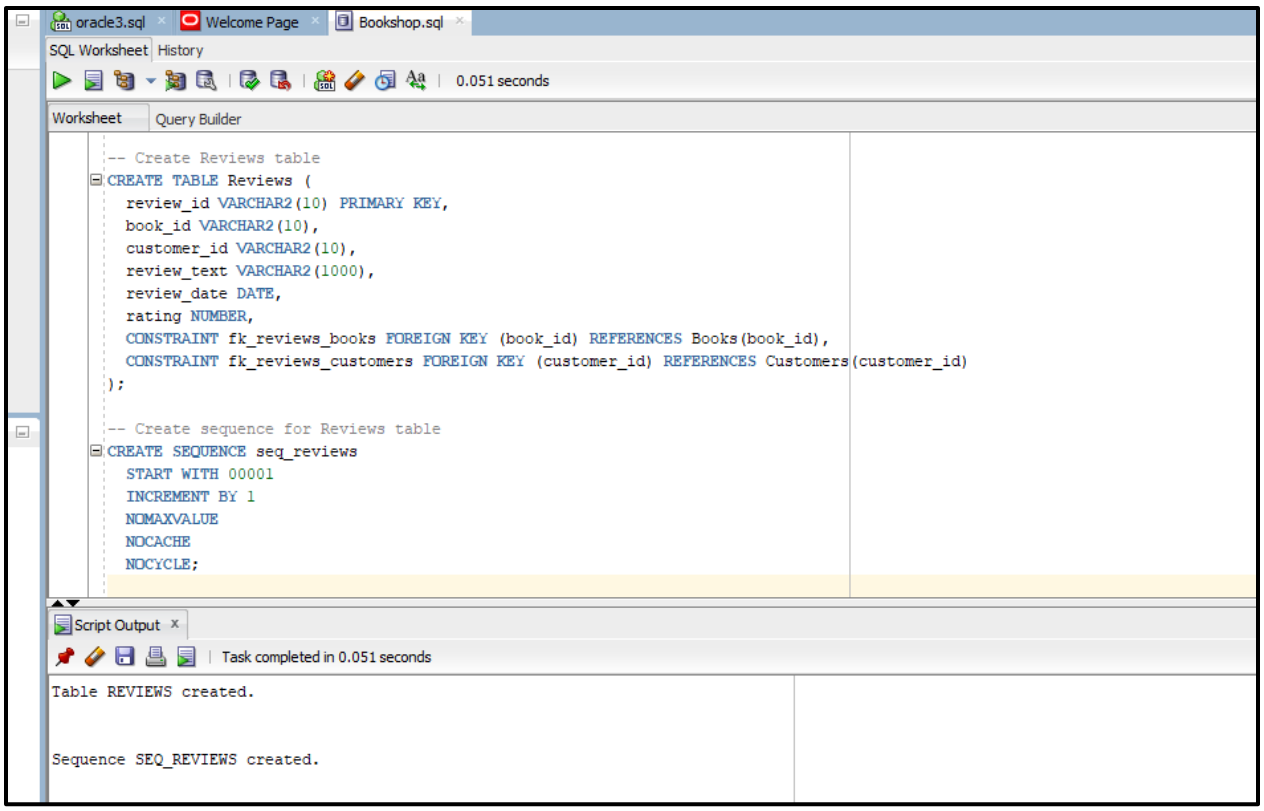


Figure 10: Output of the Created Reviews Table

2.1 Inserting values for Tables In the Bookshop Database

Question 02

Insert a set of matching records for the above tables.

Query:

```
--For Authors Table

-- Inserting data for author_id A1

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'J.K. Rowling', TO_DATE('1965-07-31', 'YYYY-MM-DD'), 'United Kingdom');

-- Inserting data for author_id A2

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'George R.R. Martin', TO_DATE('1948-09-20', 'YYYY-MM-DD'), 'United States');

-- Inserting data for author_id A3

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Stephen King', TO_DATE('1947-09-21', 'YYYY-MM-DD'), 'United States');

-- Inserting data for author_id A4

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'J.R.R. Tolkien', TO_DATE('1892-01-03', 'YYYY-MM-DD'), 'United Kingdom');

-- Inserting data for author_id A5

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Agatha Christie', TO_DATE('1890-09-15', 'YYYY-MM-DD'), 'United Kingdom');

-- Inserting data for author_id A6

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Harper Lee', TO_DATE('1926-04-28', 'YYYY-MM-DD'), 'United States');
```

Figure 11: Inserting Data for the Authors Table - 1

```
-- Inserting data for author_id A7

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'George Orwell', TO_DATE('1903-06-25', 'YYYY-MM-DD'), 'United Kingdom');

-- Inserting data for author_id A8

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'F. Scott Fitzgerald', TO_DATE('1896-09-24', 'YYYY-MM-DD'), 'United States');

-- Inserting data for author_id A9

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Jane Austen', TO_DATE('1775-12-16', 'YYYY-MM-DD'), 'United Kingdom');

-- Inserting data for author_id A10

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'J.D. Salinger', TO_DATE('1919-01-01', 'YYYY-MM-DD'), 'United States');

-- Inserting data for author_id A11

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Charlotte Brontë', TO_DATE('1816-04-21', 'YYYY-MM-DD'), 'United Kingdom');

-- Inserting data for author_id A12

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Paulo Coelho', TO_DATE('1947-08-24', 'YYYY-MM-DD'), 'Brazil');

-- Inserting data for author_id A13

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Suzanne Collins', TO_DATE('1962-08-10', 'YYYY-MM-DD'), 'United States');
```

Figure 12: Inserting Data for the Authors Table - 2

```
-- Inserting data for author_id A14

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Dan Brown', TO_DATE('1964-06-22', 'YYYY-MM-DD'),
'United States');

-- Inserting data for author_id A15

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'John Green', TO_DATE('1977-08-24', 'YYYY-MM-DD'),
'United States');

-- Inserting data for author_id A16

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Gillian Flynn', TO_DATE('1971-02-24', 'YYYY-MM-DD'),
'United States');

-- Inserting data for author_id A17

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'C.S. Lewis', TO_DATE('1898-11-29', 'YYYY-MM-DD'),
'United Kingdom');

-- Inserting data for author_id A18

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Paula Hawkins', TO_DATE('1972-08-26', 'YYYY-MM-
DD'), 'United Kingdom');

-- Inserting data for author_id A19

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Khaled Hosseini', TO_DATE('1965-03-04', 'YYYY-MM-
DD'), 'Afghanistan');

-- Inserting data for author_id A20

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'James Dashner', TO_DATE('1972-11-26', 'YYYY-MM-
DD'), 'United States');

-- Inserting data for author_id A21

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Stieg Larsson', TO_DATE('1954-08-15', 'YYYY-MM-DD'),
'Sweden');
```

Figure 13: Inserting Data for the Authors Table - 3

```
-- Inserting data for author_id A22

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Anne Frank', TO_DATE('1929-06-12', 'YYYY-MM-DD'),
'Germany');

-- Inserting data for author_id A23

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Yuval Noah Harari', TO_DATE('1976-02-24', 'YYYY-MM-DD'),
'Israel');

-- Inserting data for author_id A24

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Stephenie Meyer', TO_DATE('1973-12-24', 'YYYY-MM-DD'),
'United States');

-- Inserting data for author_id A25

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Emily Brontë', TO_DATE('1818-07-30', 'YYYY-MM-DD'), 'United
Kingdom');

-- Inserting data for author_id A26

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Bram Stoker', TO_DATE('1847-11-08', 'YYYY-MM-DD'),
'Ireland');

-- Inserting data for author_id A27

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Mary Shelley', TO_DATE('1797-08-30', 'YYYY-MM-DD'), 'United
Kingdom');

-- Inserting data for author_id A28

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Shirley Jackson', TO_DATE('1916-12-14', 'YYYY-MM-DD'),
'United States');

-- Inserting data for author_id A29

INSERT INTO Authors (author_id, author_name, author_birthdate, author_country)

VALUES ('A'||seq_authors.NEXTVAL, 'Daphne du Maurier', TO_DATE('1907-05-13', 'YYYY-MM-DD'),
'United Kingdom');
```

Figure 14: Inserting Data for the Authors Table - 4

```
--Inserting Data For Publisher Table

-- Inserting data for publisher_id P1

INSERT INTO Publishers (publisher_id, publisher_name, publisher_location, publisher_contact)

VALUES ('P'||seq_publishers.NEXTVAL, 'Penguin Random House', 'New York',
'info@penguinrandomhouse.com');

-- Inserting data for publisher_id P2

INSERT INTO Publishers (publisher_id, publisher_name, publisher_location, publisher_contact)

VALUES ('P'||seq_publishers.NEXTVAL, 'HarperCollins Publishers', 'New York',
'info@harpercollins.com');

-- Inserting data for publisher_id P3

INSERT INTO Publishers (publisher_id, publisher_name, publisher_location, publisher_contact)

VALUES ('P'||seq_publishers.NEXTVAL, 'Hachette Livre', 'Paris', 'info@hachette-livre.fr');

-- Inserting data for publisher_id P4

INSERT INTO Publishers (publisher_id, publisher_name, publisher_location, publisher_contact)

VALUES ('P'||seq_publishers.NEXTVAL, 'Simon and Schuster', 'New York',
'info@simonandschuster.com');

-- Inserting data for publisher_id P5

INSERT INTO Publishers (publisher_id, publisher_name, publisher_location, publisher_contact)

VALUES ('P'||seq_publishers.NEXTVAL, 'Macmillan Publishers', 'London', 'info@macmillan.com');

-- Inserting data for publisher_id P6

INSERT INTO Publishers (publisher_id, publisher_name, publisher_location, publisher_contact)

VALUES ('P'||seq_publishers.NEXTVAL, 'Oxford University Press', 'Oxford', 'info@oup.com');


--Inserting Data For The Book Table

-- Inserting data for book_id B1

INSERT INTO Books (book_id, book_title, book_genre, book_publication_date, book_price,
publisher_id,author_id)

VALUES ('B'||seq_books.NEXTVAL, '1984', 'Dystopian Fiction', TO_DATE('1949-06-08', 'YYYY-
MM-DD'), 9.99, 'P1','A7');
```

Figure 15: Inserting Data for the Publisher Table

```
-- Inserting data for book_id B2

INSERT INTO Books (book_id, book_title, book_genre, book_publication_date, book_price, publisher_id,author_id)

VALUES ('B'||seq_books.NEXTVAL, 'To Kill a Mockingbird', 'Fiction', TO_DATE('1960-07-11', 'YYYY-MM-DD'), 12.99, 'P2','A6');

-- Inserting data for book_id B3

INSERT INTO Books (book_id, book_title, book_genre, book_publication_date, book_price, publisher_id,author_id)

VALUES ('B'||seq_books.NEXTVAL, 'Harry Potter and the Philosopher''s Stone', 'Fantasy', TO_DATE('1997-06-26', 'YYYY-MM-DD'), 14.99, 'P1','A1');

-- Inserting data for book_id B4

INSERT INTO Books (book_id, book_title, book_genre, book_publication_date, book_price, publisher_id,author_id)

VALUES ('B'||seq_books.NEXTVAL, 'The Great Gatsby', 'Fiction', TO_DATE('1925-04-10', 'YYYY-MM-DD'), 11.99, 'P3','A8');

Similarly Insert data for the rest....

--Insert Data For Customer Table

-- Inserting data for customer_id C1

INSERT INTO Customers (customer_id, customer_name, customer_email, customer_address, customer_phone)

VALUES ('C'||seq_customers.NEXTVAL, 'Kamal Perera', 'kamalperera@example.com', '123 Main St, Colombo', '+94 77-123-4567');

-- Inserting data for customer_id C2

INSERT INTO Customers (customer_id, customer_name, customer_email, customer_address, customer_phone)

VALUES ('C'||seq_customers.NEXTVAL, 'Nimal Fernando', 'nimalfernando@example.com', '456 Galle Rd, Kandy', '+94 76-987-6543');

-- Inserting data for customer_id C3

INSERT INTO Customers (customer_id, customer_name, customer_email, customer_address, customer_phone)

VALUES ('C'||seq_customers.NEXTVAL, 'Samantha Silva', 'samanthasilva@example.com', '789 Negombo Rd, Negombo', '+94 71-555-4444');
```

Figure 16: Inserting Data for the Customers Table - 1

```
-- Inserting data for customer_id C4

INSERT INTO Customers (customer_id, customer_name, customer_email, customer_address,
customer_phone)

VALUES ('C'||seq_customers.NEXTVAL, 'Priyanthi Gunaratne', 'priyanthigunaratne@example.com', '321
Kandy Rd, Gampaha', '+94 76-999-8888');

-- Inserting data for customer_id C5

INSERT INTO Customers (customer_id, customer_name, customer_email, customer_address,
customer_phone)

VALUES ('C'||seq_customers.NEXTVAL, 'Ravi Bandara', 'ravibandara@example.com', '567 Main St,
Matara', '+94 77-111-2222');

-- Inserting data for customer_id C6

INSERT INTO Customers (customer_id, customer_name, customer_email, customer_address,
customer_phone)

VALUES ('C'||seq_customers.NEXTVAL, 'Mala Perumal', 'malaperumal@example.com', '987 Galle Rd,
Jaffna', '+94 76-444-5555');

-- Inserting data for customer_id C7

INSERT INTO Customers (customer_id, customer_name, customer_email, customer_address,
customer_phone)

VALUES ('C'||seq_customers.NEXTVAL, 'Chaminda Jayawardena',
'chamindajayawardena@example.com', '654 Colombo Rd, Kurunegala', '+94 71-777-8888');

-- Inserting data for customer_id C8

INSERT INTO Customers (customer_id, customer_name, customer_email, customer_address,
customer_phone)

VALUES ('C'||seq_customers.NEXTVAL, 'Sarala Rajapakse', 'saralarajapakse@example.com', '852
Negombo Rd, Anuradhapura', '+94 77-222-3333');

-- Inserting data for customer_id C9

INSERT INTO Customers (customer_id, customer_name, customer_email, customer_address,
customer_phone)

VALUES ('C'||seq_customers.NEXTVAL, 'Dilshan Peiris', 'dilshanpeiris@example.com', '456 Kandy Rd,
Galle', '+94 76-666-7777');

-- Inserting data for customer_id C10

INSERT INTO Customers (customer_id, customer_name, customer_email, customer_address,
customer_phone)

VALUES ('C'||seq_customers.NEXTVAL, 'Malini Wijesuriya', 'maliniwijesuriya@example.com', '123
Main St, Ratnapura', '+94 71-999-8888');
```

Figure 17: Inserting Data for the Customers Table - 2

```
 -- Inserting data for customer_id C11

 INSERT INTO Customers (customer_id, customer_name, customer_email, customer_address,
customer_phone)

 VALUES ('C'||seq_customers.NEXTVAL, 'Thilak Kumara', 'thilakkumara@example.com', '789 Galle Rd,
Badulla', '+94 76-333-4444');

 -- Inserting data for customer_id C12

 INSERT INTO Customers (customer_id, customer_name, customer_email, customer_address,
customer_phone)

 VALUES ('C'||seq_customers.NEXTVAL, 'Kumari Jayasinghe', 'kumarijayasinghe@example.com', '987
Colombo Rd, Kurunegala', '+94 71-777-6666');


 --  Insert Data For Orders Table

 -- Inserting data for order_id O1

  INSERT INTO Orders (order_id, customer_id, order_date, total_amount)

  VALUES ('O'||seq_orders.NEXTVAL, 'C1', TO_DATE('2023-06-01', 'YYYY-MM-DD'), 150.00);

  -- Inserting data for order_id O2

  INSERT INTO Orders (order_id, customer_id, order_date, total_amount)

  VALUES ('O'||seq_orders.NEXTVAL, 'C2', TO_DATE('2023-06-02', 'YYYY-MM-DD'), 120.50);

  -- Inserting data for order_id O3

 INSERT INTO Orders (order_id, customer_id, order_date, total_amount)

 VALUES ('O'||seq_orders.NEXTVAL, 'C3', TO_DATE('2023-06-02', 'YYYY-MM-DD'), 75.80);

  -- Inserting data for order_id O4

 INSERT INTO Orders (order_id, customer_id, order_date, total_amount)

VALUES ('O'||seq_orders.NEXTVAL, 'C4', TO_DATE('2023-06-03', 'YYYY-MM-DD'), 200.00);

-- Inserting data for order_id O5

INSERT INTO Orders (order_id, customer_id, order_date, total_amount)

VALUES ('O'||seq_orders.NEXTVAL, 'C5', TO_DATE('2023-06-04', 'YYYY-MM-DD'), 85.20);
```

Figure 18: Inserting Data for the Orders Table - 1

```
-- Inserting data for order_id O6
INSERT INTO Orders (order_id, customer_id, order_date, total_amount)
VALUES ('O'||seq_orders.NEXTVAL, 'C6', TO_DATE('2023-06-05', 'YYYY-MM-DD'), 175.50);
-- Inserting data for order_id O7
INSERT INTO Orders (order_id, customer_id, order_date, total_amount)
VALUES ('O'||seq_orders.NEXTVAL, 'C7', TO_DATE('2023-06-06', 'YYYY-MM-DD'), 110.00);
-- Inserting data for order_id O8
INSERT INTO Orders (order_id, customer_id, order_date, total_amount)
VALUES ('O'||seq_orders.NEXTVAL, 'C8', TO_DATE('2023-06-06', 'YYYY-MM-DD'), 95.75);
-- Inserting data for order_id O9
INSERT INTO Orders (order_id, customer_id, order_date, total_amount)
VALUES ('O'||seq_orders.NEXTVAL, 'C9', TO_DATE('2023-06-07', 'YYYY-MM-DD'), 120.25);
-- Inserting data for order_id O10
INSERT INTO Orders (order_id, customer_id, order_date, total_amount)
VALUES ('O'||seq_orders.NEXTVAL, 'C10', TO_DATE('2023-06-08', 'YYYY-MM-DD'), 140.00);
--Insert Data For Order_Item Table
-- Inserting data for order_item_id OI1
INSERT INTO Order_Items (order_item_id, order_id, book_id, quantity, item_price)
VALUES ('OI'||seq_order_items.NEXTVAL, 'O1', 'B1', 2, 20.00);
-- Inserting data for order_item_id OI2
INSERT INTO Order_Items (order_item_id, order_id, book_id, quantity, item_price)
VALUES ('OI'||seq_order_items.NEXTVAL, 'O2', 'B2', 1, 15.50);
-- Inserting data for order_item_id OI3
INSERT INTO Order_Items (order_item_id, order_id, book_id, quantity, item_price)
VALUES ('OI'||seq_order_items.NEXTVAL, 'O2', 'B4', 3, 8.60);
```

Figure 19: Inserting Data for the Orders Table - 2

```
 -- Inserting data for order_item_id OI4
 INSERT INTO Order_Items (order_item_id, order_id, book_id, quantity, item_price)
VALUES ('OI'||seq_order_items.NEXTVAL, 'O3', 'B1', 1, 20.00);
-- Inserting data for order_item_id OI5
INSERT INTO Order_Items (order_item_id, order_id, book_id, quantity, item_price)
VALUES ('OI'||seq_order_items.NEXTVAL, 'O4', 'B3', 2, 25.00);
-- Inserting data for order_item_id OI6
INSERT INTO Order_Items (order_item_id, order_id, book_id, quantity, item_price)
VALUES ('OI'||seq_order_items.NEXTVAL, 'O5', 'B5', 1, 10.50);
-- Inserting data for order_item_id OI7
INSERT INTO Order_Items (order_item_id, order_id, book_id, quantity, item_price)
VALUES ('OI'||seq_order_items.NEXTVAL, 'O6', 'B1', 2, 20.00);
-- Inserting data for order_item_id OI8
INSERT INTO Order_Items (order_item_id, order_id, book_id, quantity, item_price)
VALUES ('OI'||seq_order_items.NEXTVAL, 'O6', 'B3', 1, 25.00);
-- Inserting data for order_item_id OI9
 INSERT INTO Order_Items (order_item_id, order_id, book_id, quantity, item_price)
VALUES ('OI'||seq_order_items.NEXTVAL, 'O7', 'B2', 3, 15.50);
-- Inserting data for order_item_id OI10
INSERT INTO Order_Items (order_item_id, order_id, book_id, quantity, item_price)
 VALUES ('OI'||seq_order_items.NEXTVAL, 'O8', 'B4', 2, 8.60);
 --Insert Data Into Review Table
 -- Inserting data for review_id R1
INSERT INTO Reviews (review_id, book_id, customer_id, review_text, review_date, rating)
VALUES ('R'||seq_reviews.NEXTVAL, 'B1', 'C1', 'Great book! I highly recommend it.',
TO_DATE('2023-06-01', 'YYYY-MM-DD'), 5);
```

Figure 20: Inserting Data for the Orders and Reviews Table

```
-- Inserting data for review_id R2

INSERT INTO Reviews (review_id, book_id, customer_id, review_text, review_date, rating)

VALUES ('R'||seq_reviews.NEXTVAL, 'B2', 'C2', 'An intriguing story with unexpected twists.',
TO_DATE('2023-06-02', 'YYYY-MM-DD'), 4);

-- Inserting data for review_id R3

INSERT INTO Reviews (review_id, book_id, customer_id, review_text, review_date, rating)

VALUES ('R'||seq_reviews.NEXTVAL, 'B3', 'C3', 'A must-read for mystery lovers.',
TO_DATE('2023-06-02', 'YYYY-MM-DD'), 4);

-- Inserting data for review_id R4

INSERT INTO Reviews (review_id, book_id, customer_id, review_text, review_date, rating)

VALUES ('R'||seq_reviews.NEXTVAL, 'B4', 'C4', 'Captivating characters and beautiful prose.',
TO_DATE('2023-06-03', 'YYYY-MM-DD'), 5);

-- Inserting data for review_id R5

INSERT INTO Reviews (review_id, book_id, customer_id, review_text, review_date, rating)

VALUES ('R'||seq_reviews.NEXTVAL, 'B5', 'C5', 'A heartwarming story of love and friendship.',
TO_DATE('2023-06-04', 'YYYY-MM-DD'), 4);

-- Inserting data for review_id R6

INSERT INTO Reviews (review_id, book_id, customer_id, review_text, review_date, rating)

VALUES ('R'||seq_reviews.NEXTVAL, 'B1', 'C6', 'Couldnt put it down! Absolutely loved it.',
TO_DATE('2023-06-05', 'YYYY-MM-DD'), 5);

-- Inserting data for review_id R7

INSERT INTO Reviews (review_id, book_id, customer_id, review_text, review_date, rating)

VALUES ('R'||seq_reviews.NEXTVAL, 'B2', 'C7', 'An emotional rollercoaster that touched my
heart.', TO_DATE('2023-06-06', 'YYYY-MM-DD'), 5);

-- Inserting data for review_id R8

INSERT INTO Reviews (review_id, book_id, customer_id, review_text, review_date, rating)

VALUES ('R'||seq_reviews.NEXTVAL, 'B3', 'C8', 'A gripping page-turner that kept me hooked.',
TO_DATE('2023-06-06', 'YYYY-MM-DD'), 4);
```

Figure 21: Inserting Data for the Reviews Table

```
-- Inserting data for review_id R9

INSERT INTO Reviews (review_id, book_id, customer_id, review_text, review_date, rating)

VALUES ('R'||seq_reviews.NEXTVAL, 'B4', 'C9', 'A thought-provoking and beautifully written book.', TO_DATE('2023-06-07', 'YYYY-MM-DD'), 5);

-- Inserting data for review_id R10

INSERT INTO Reviews (review_id, book_id, customer_id, review_text, review_date, rating)

VALUES ('R'||seq_reviews.NEXTVAL, 'B5', 'C10', 'Highly recommended for fans of the genre.', TO_DATE('2023-06-08', 'YYYY-MM-DD'), 4);


--To check whether data is inserted

select * from authors;

select * from publishers;

select * from customers;

select * from orders;

select * from books;

select * from reviews;

select * from order_items;
```

Figure 22: Inserting Data for the Reviews Table and Selecting Data

Outputs:

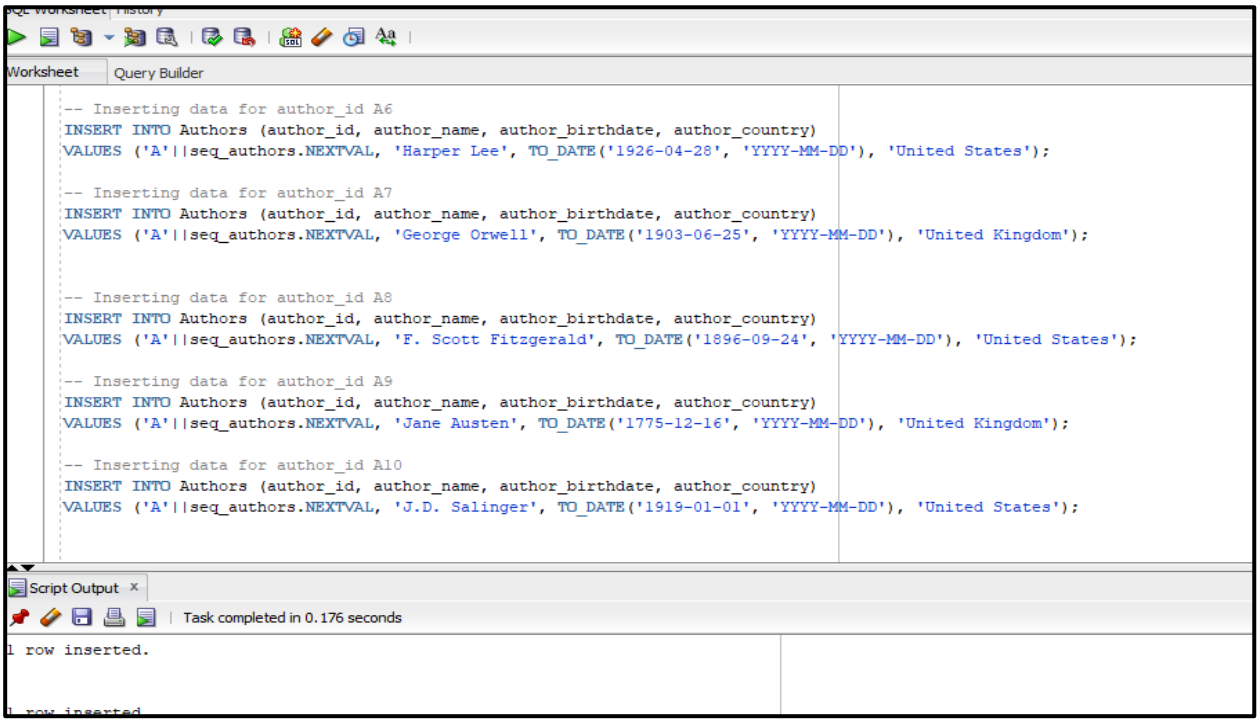


Figure 23: Inserting Data for the Authors Table (Output) - 1

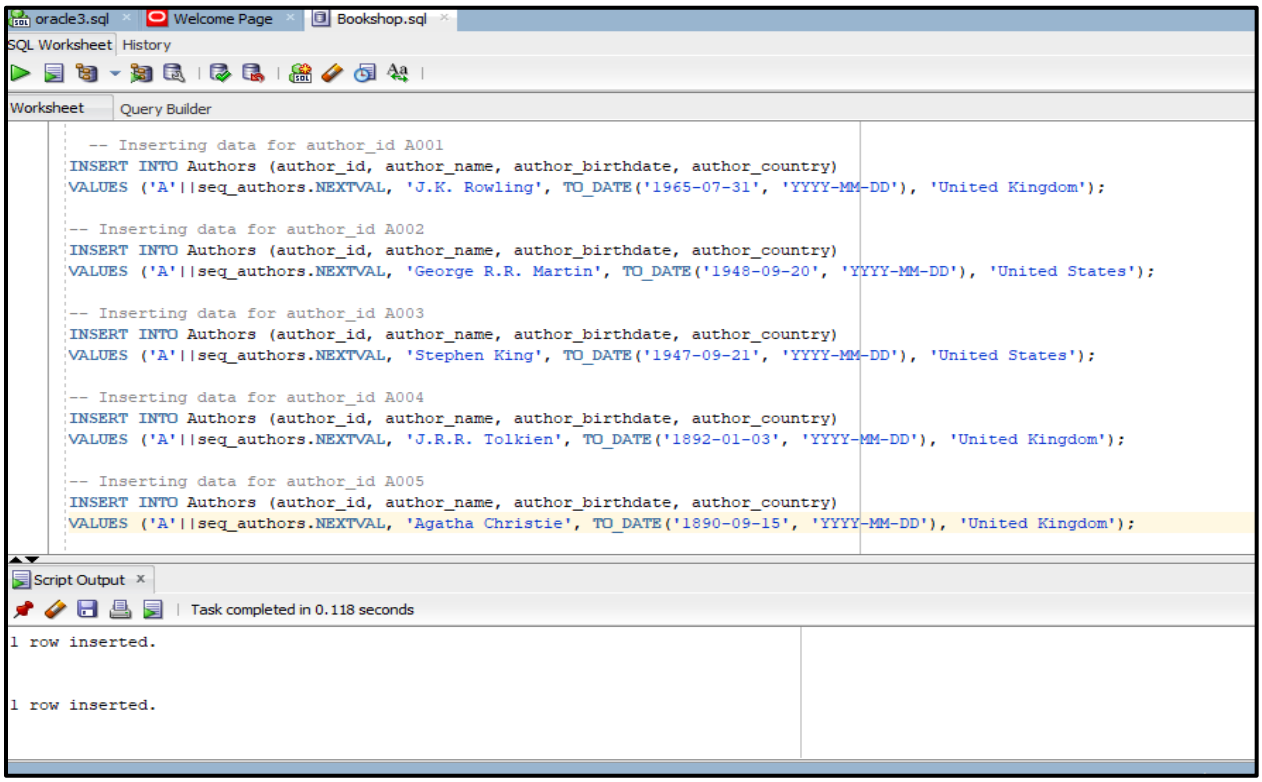


Figure 24: Inserting Data for the Authors Table (Output) - 2

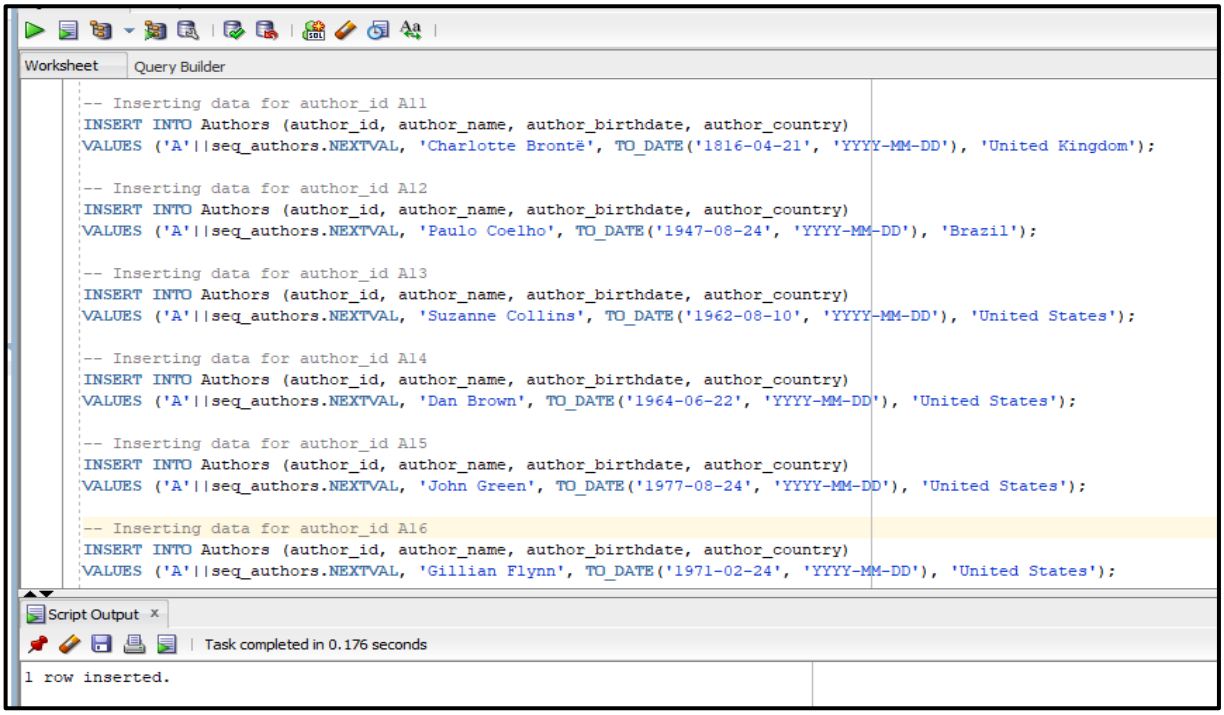


Figure 25: Inserting Data for the Authors Table (Output) - 3

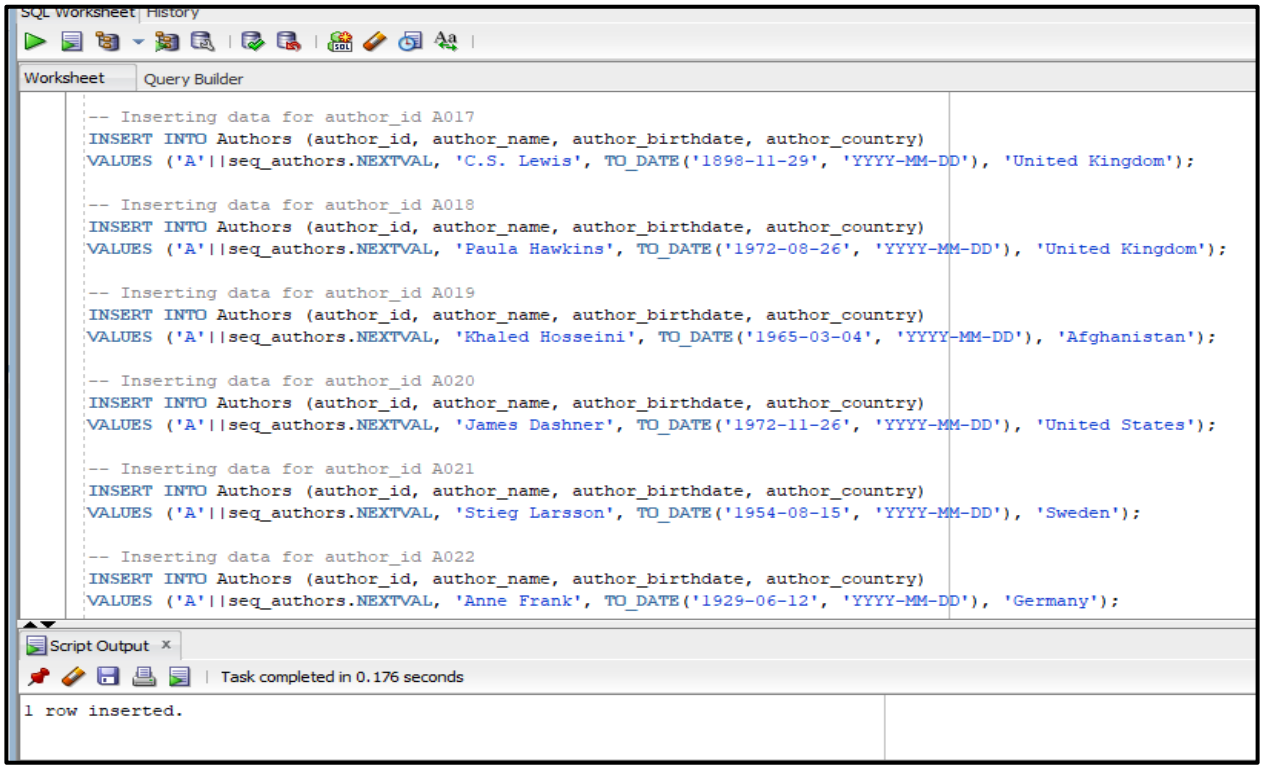


Figure 26: Inserting Data for the Authors Table (Output) - 4

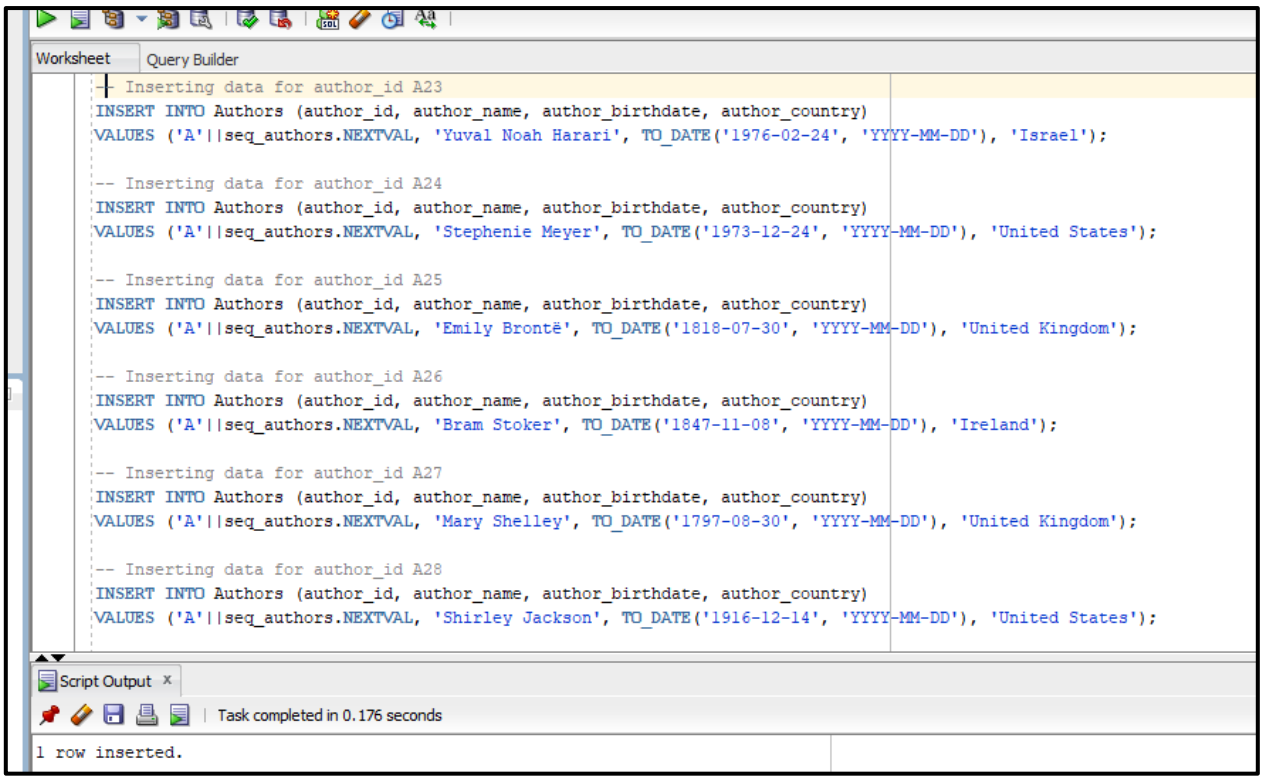


Figure 27: Inserting Data for the Authors Table (Output) - 5

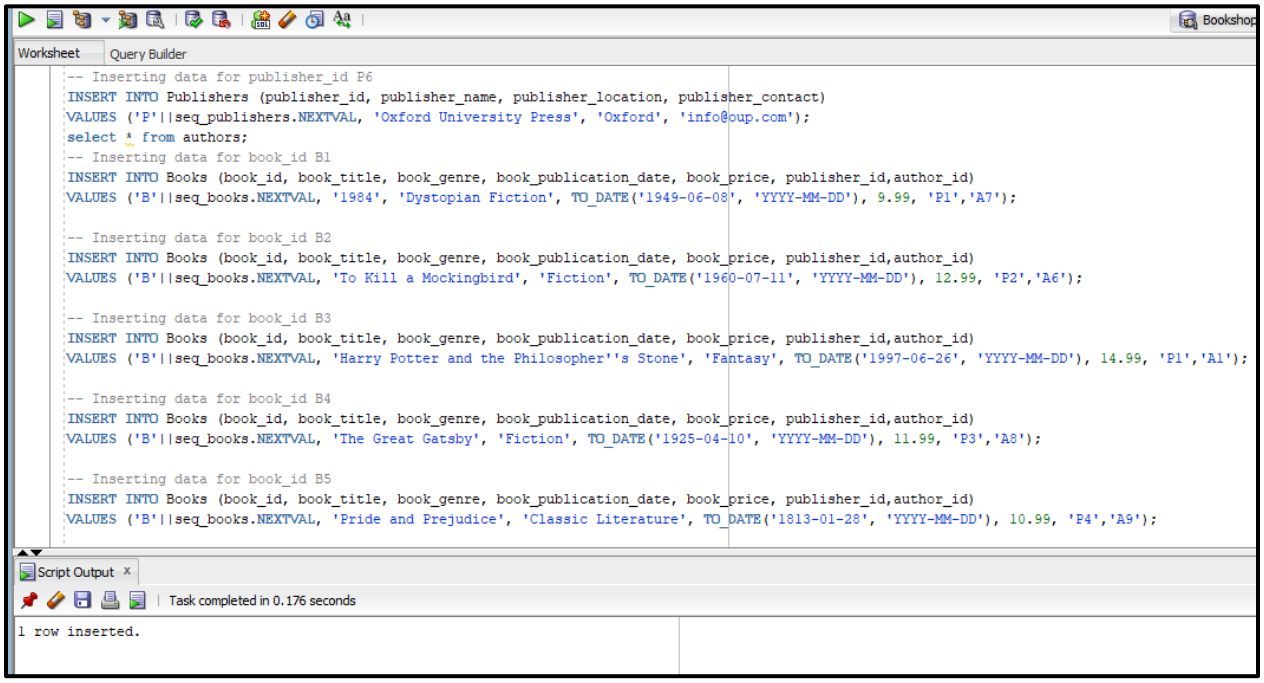


Figure 28: Inserting Data for the Publishers Table (Output) - 1

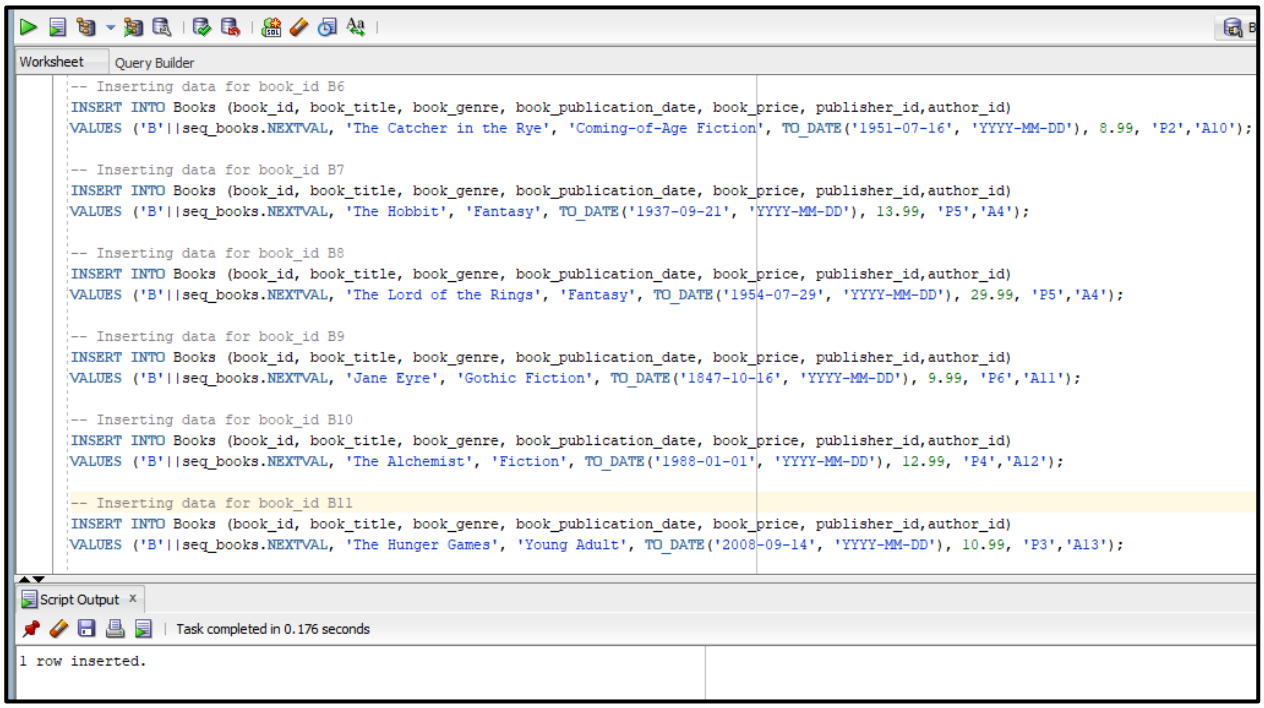


Figure 29: Inserting Data for the Books Table (Output) - 1

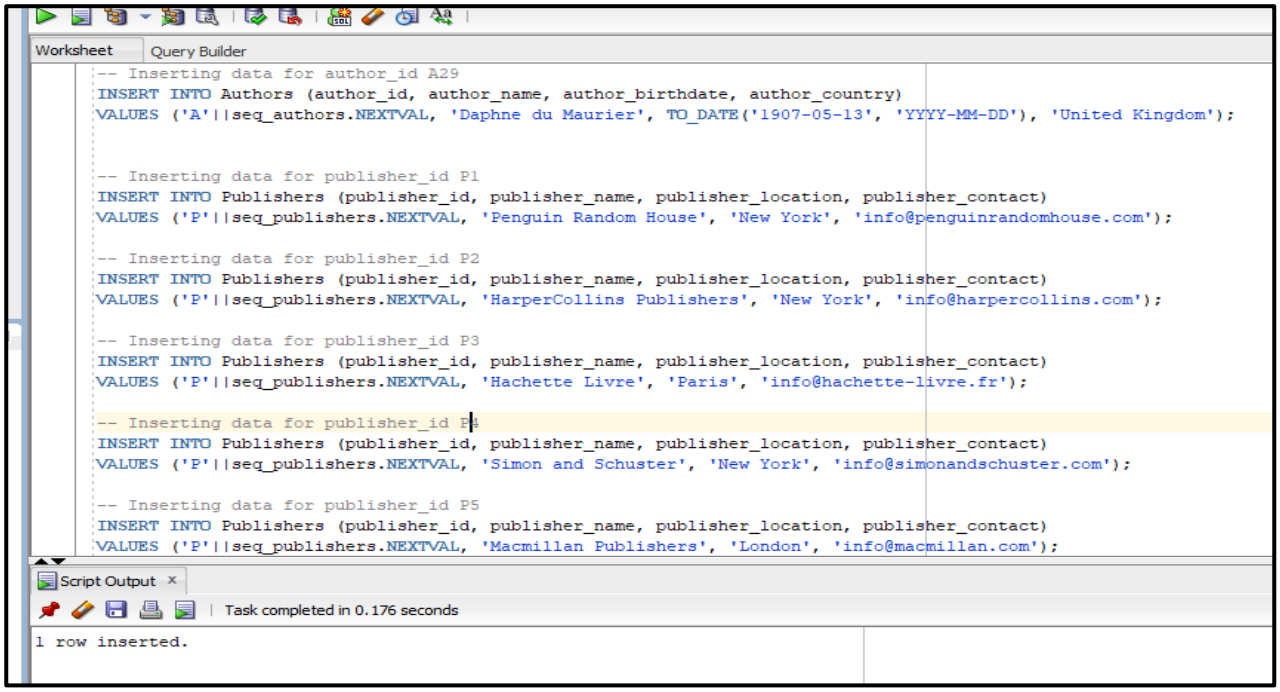


Figure 30: Inserting Data for the Publishers Table (Output) - 2

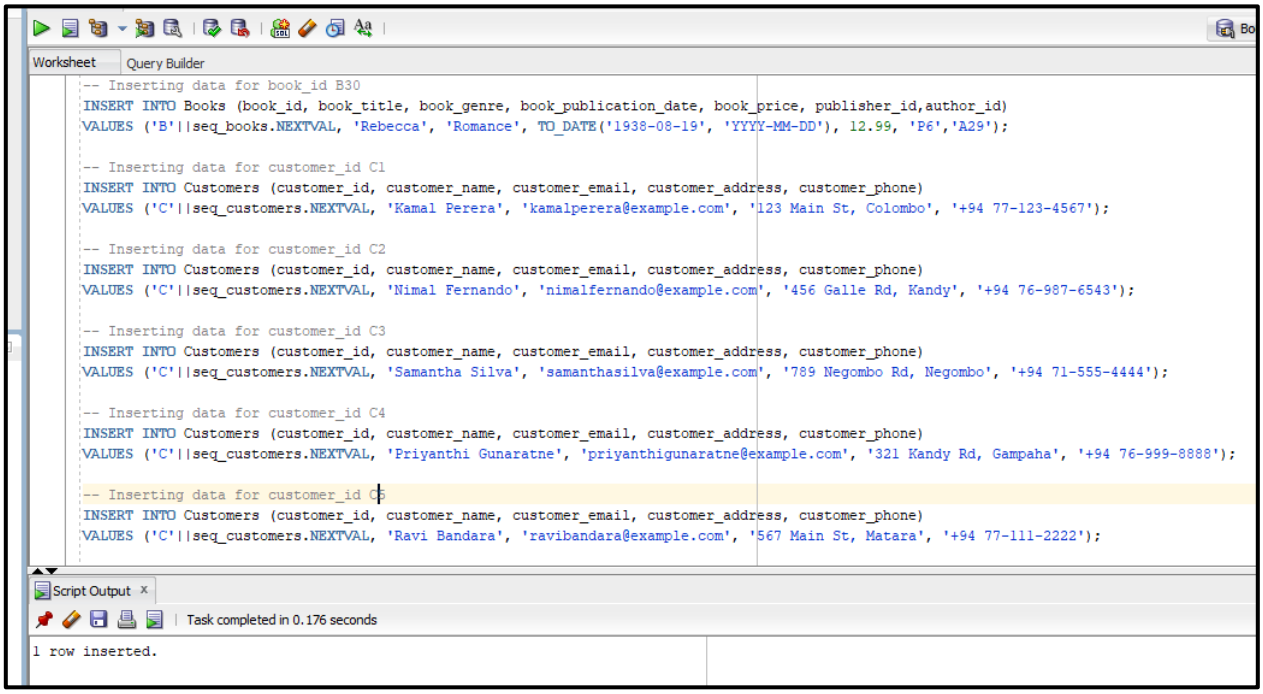


Figure 32: Inserting Data for the Books Table (Output) - 2

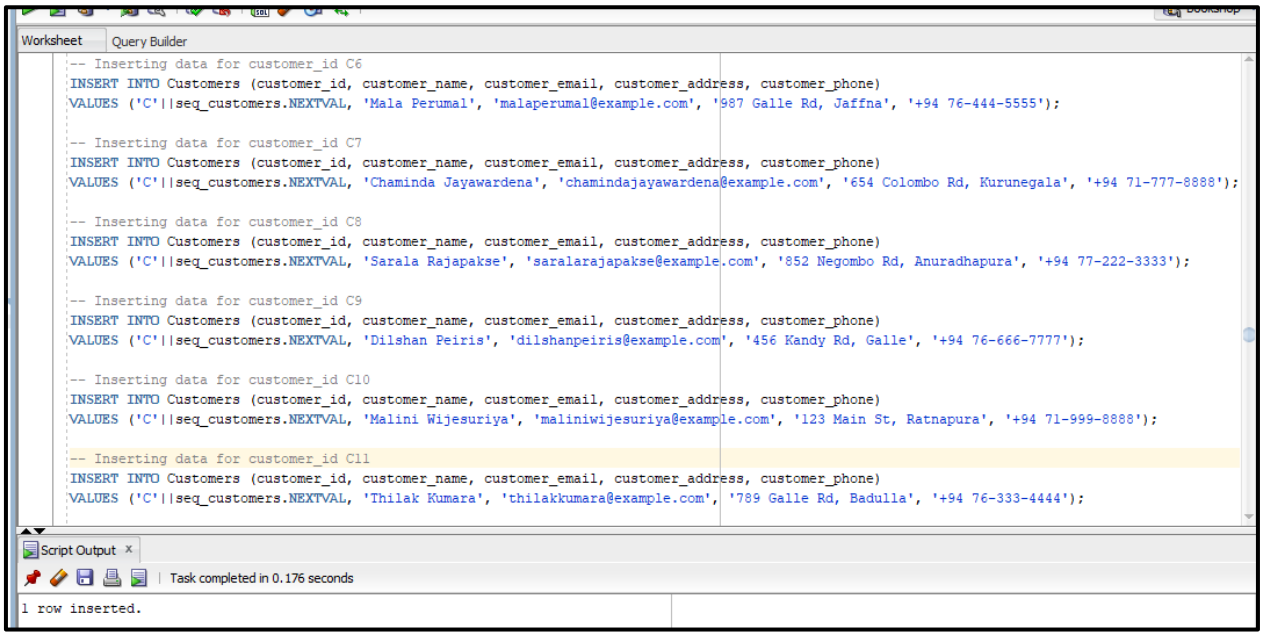


Figure 31: Inserting Data for the Customers Table (Output) - 1

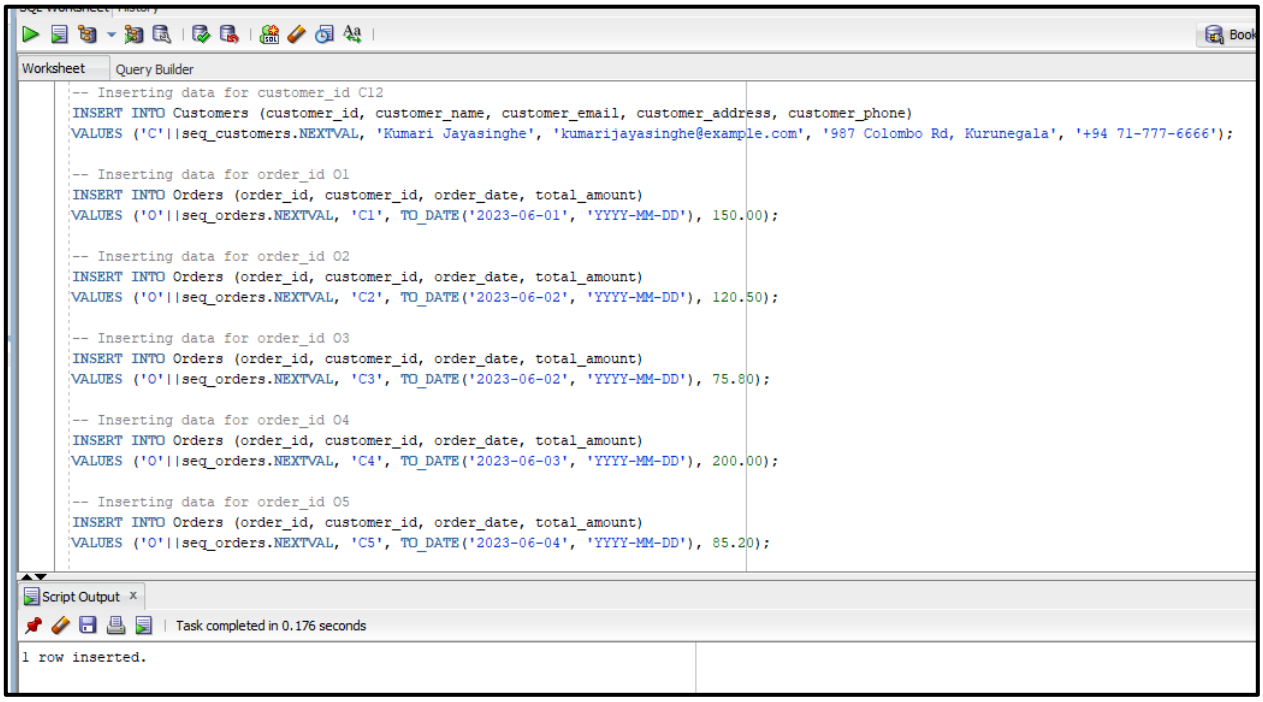


Figure 34: Inserting Data for the Orders Table (Output) - 1

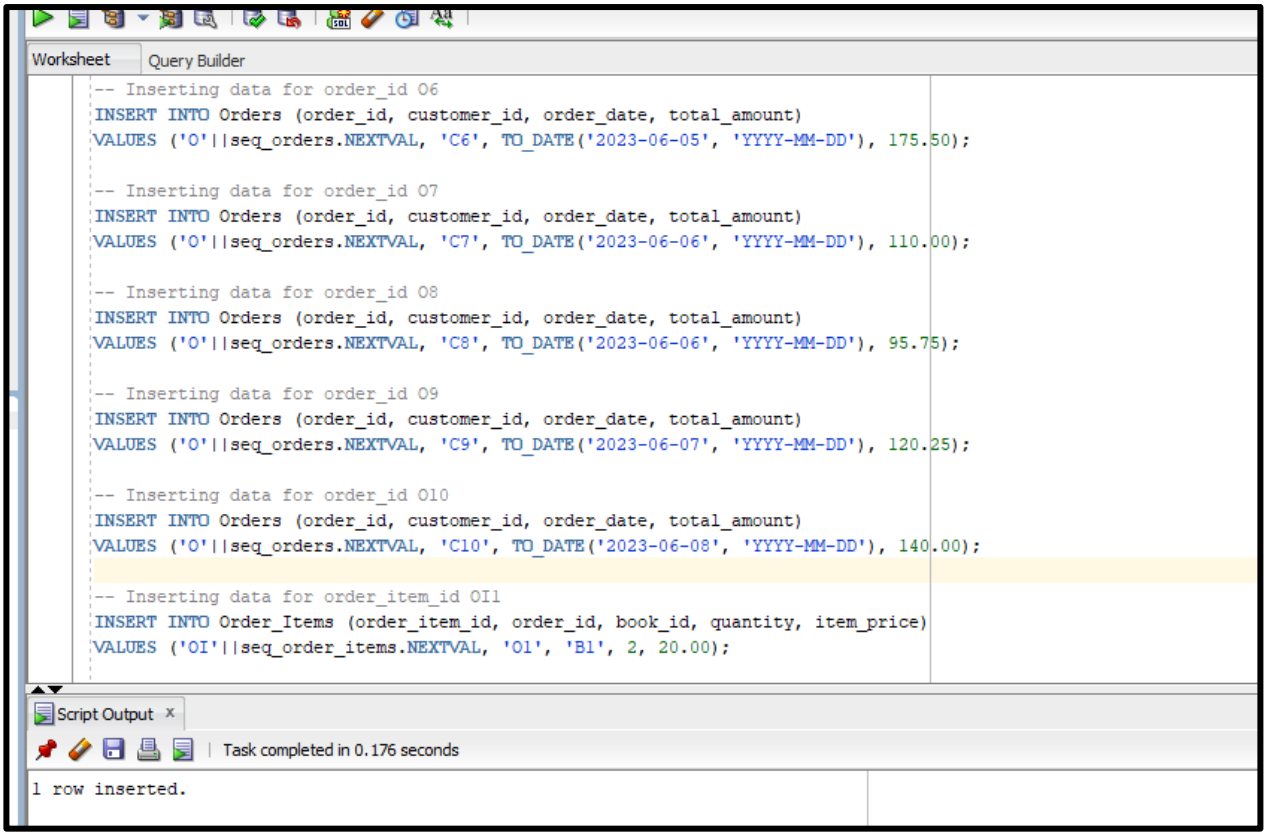


Figure 33: Inserting Data for the Orders Table (Output) - 2

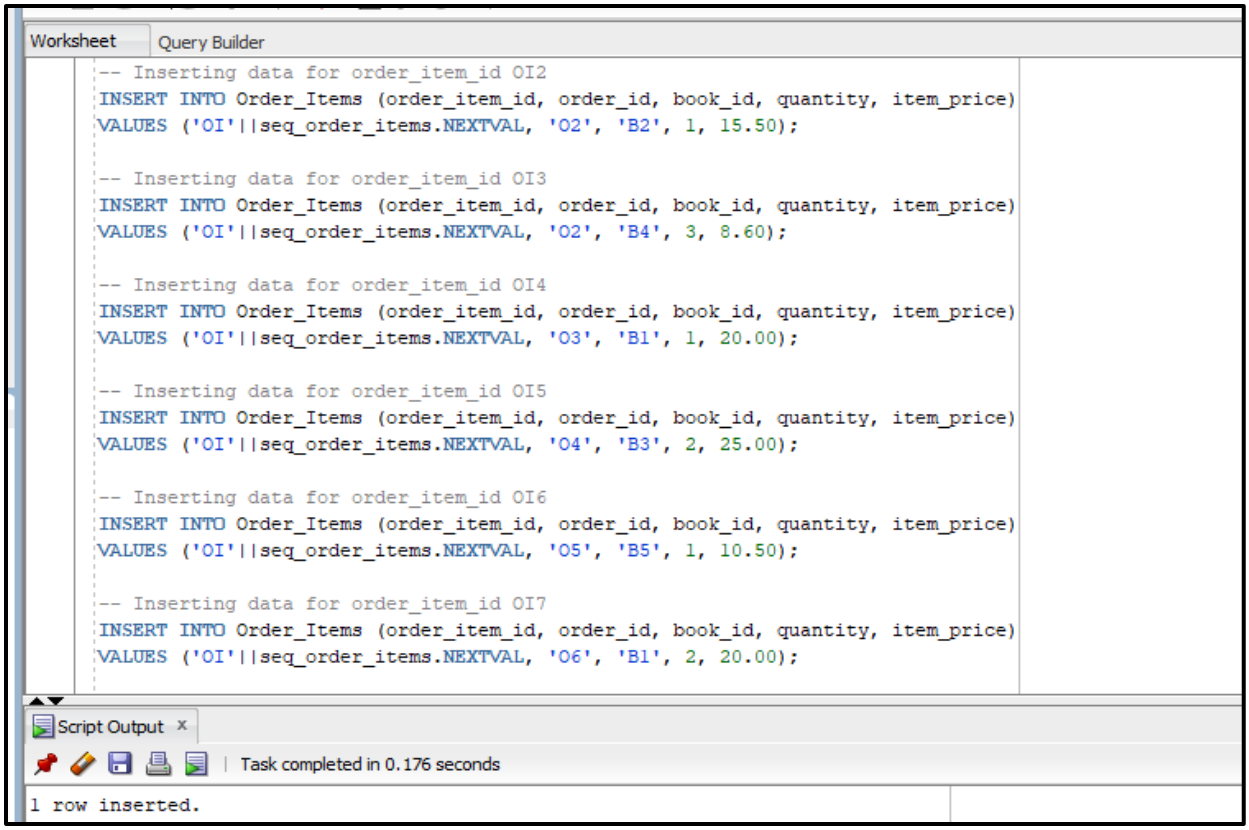


Figure 36: Inserting Data for the Order_Items Table (Output) - 1

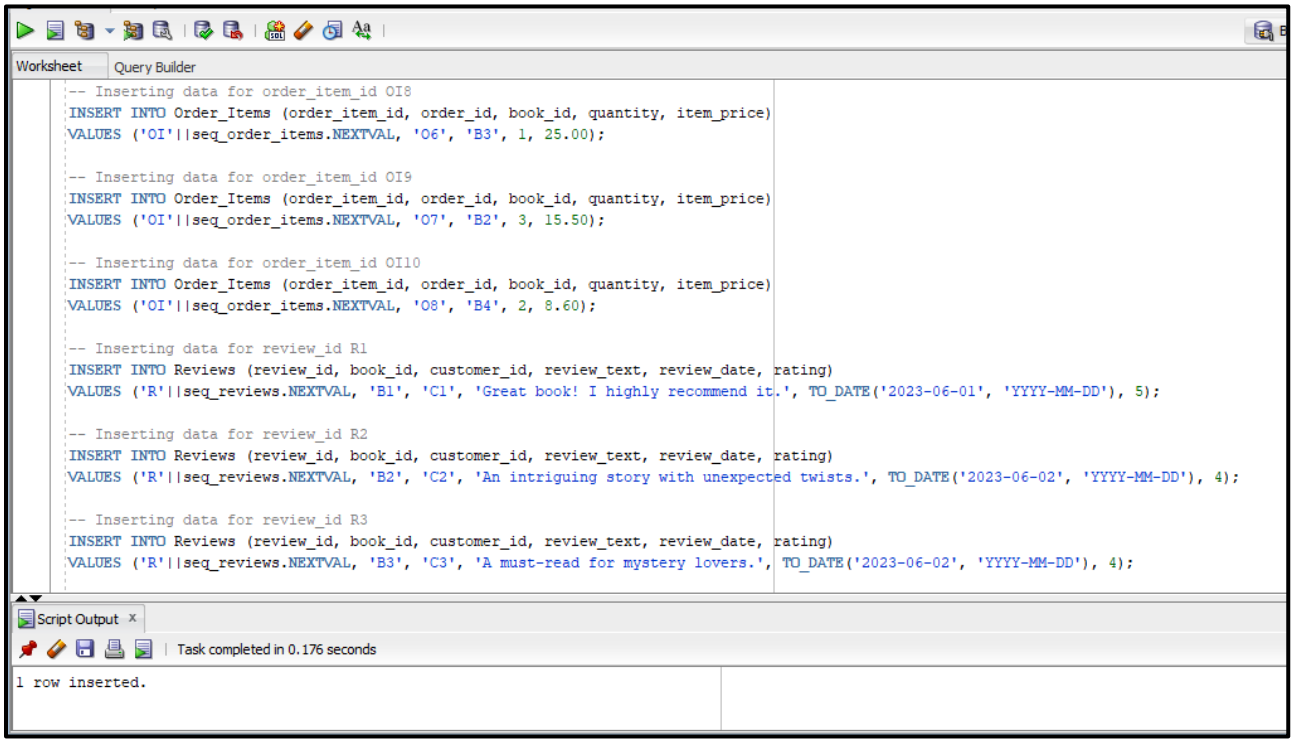


Figure 35: Inserting Data for the Order_Items Table (Output) - 2

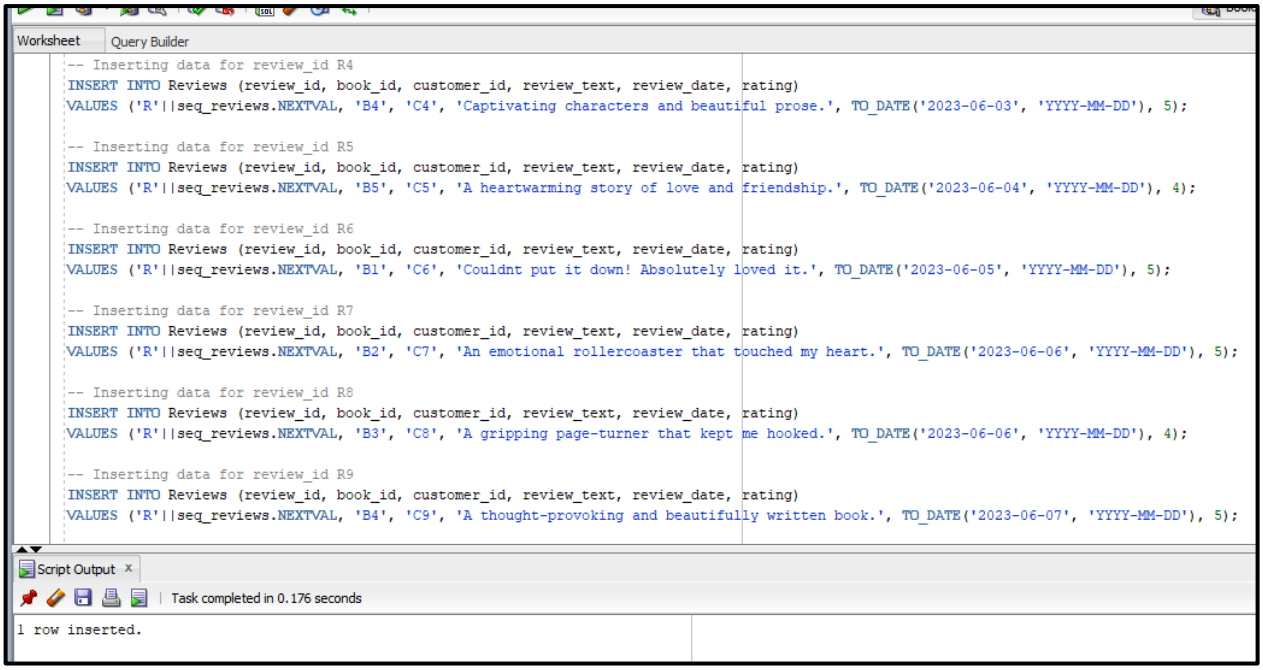
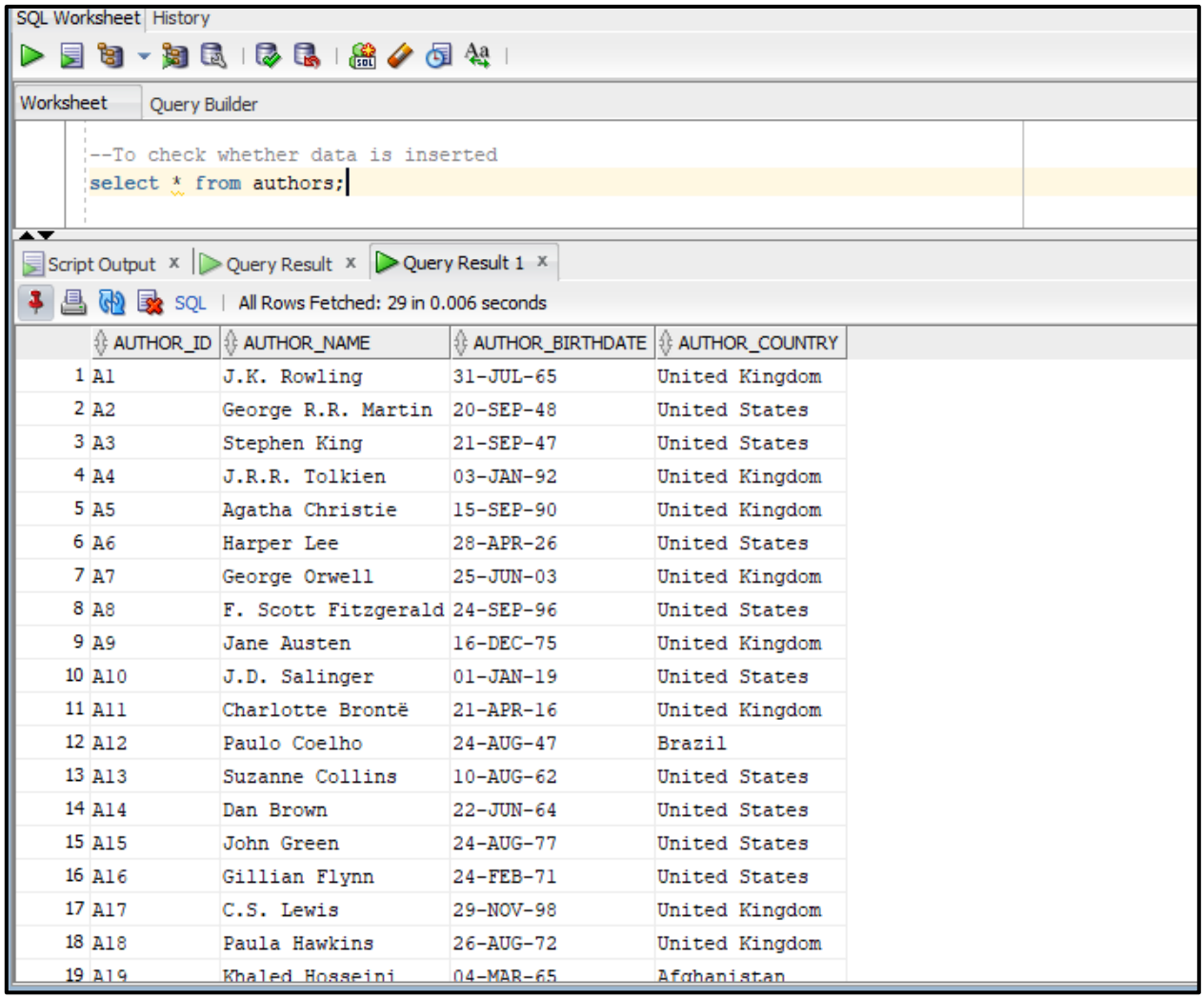


Figure 37: Inserting Data for the Reviews Table (Output) - 1

OUTPUT RESULTS:

- Show data in Authors Table



	AUTHOR_ID	AUTHOR_NAME	AUTHOR_BIRTHDATE	AUTHOR_COUNTRY
1	A1	J.K. Rowling	31-JUL-65	United Kingdom
2	A2	George R.R. Martin	20-SEP-48	United States
3	A3	Stephen King	21-SEP-47	United States
4	A4	J.R.R. Tolkien	03-JAN-92	United Kingdom
5	A5	Agatha Christie	15-SEP-90	United Kingdom
6	A6	Harper Lee	28-APR-26	United States
7	A7	George Orwell	25-JUN-03	United Kingdom
8	A8	F. Scott Fitzgerald	24-SEP-96	United States
9	A9	Jane Austen	16-DEC-75	United Kingdom
10	A10	J.D. Salinger	01-JAN-19	United States
11	A11	Charlotte Brontë	21-APR-16	United Kingdom
12	A12	Paulo Coelho	24-AUG-47	Brazil
13	A13	Suzanne Collins	10-AUG-62	United States
14	A14	Dan Brown	22-JUN-64	United States
15	A15	John Green	24-AUG-77	United States
16	A16	Gillian Flynn	24-FEB-71	United States
17	A17	C.S. Lewis	29-NOV-98	United Kingdom
18	A18	Paula Hawkins	26-AUG-72	United Kingdom
19	A19	Khaled Hosseini	04-MAR-65	Afghanistan

Figure 38: Displays all the Data from Authors Table - 1

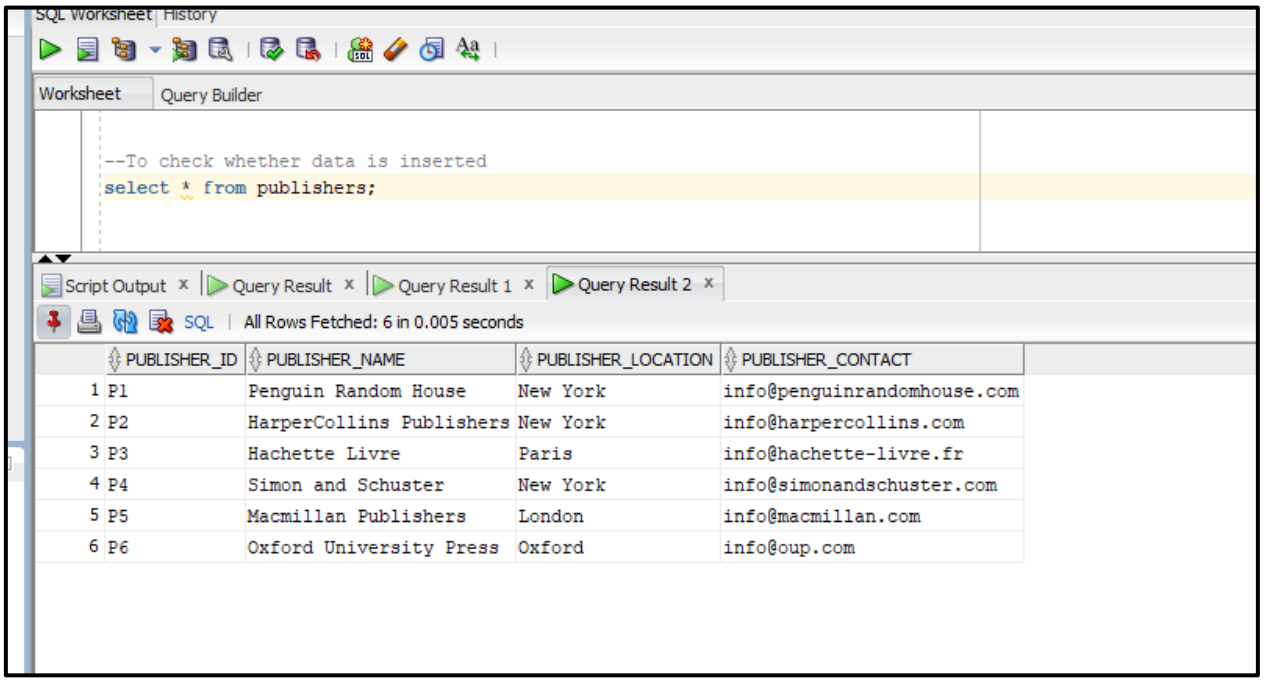


Figure 40: Displays all the Data from Publishers Table

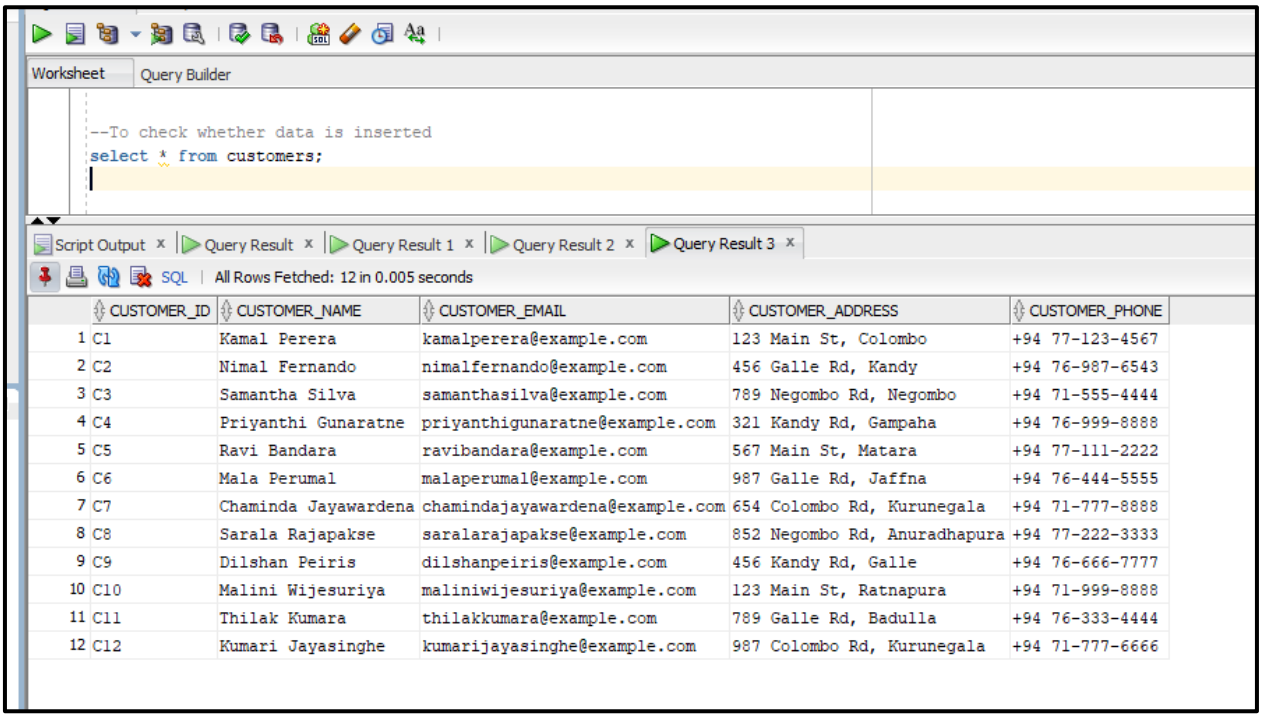
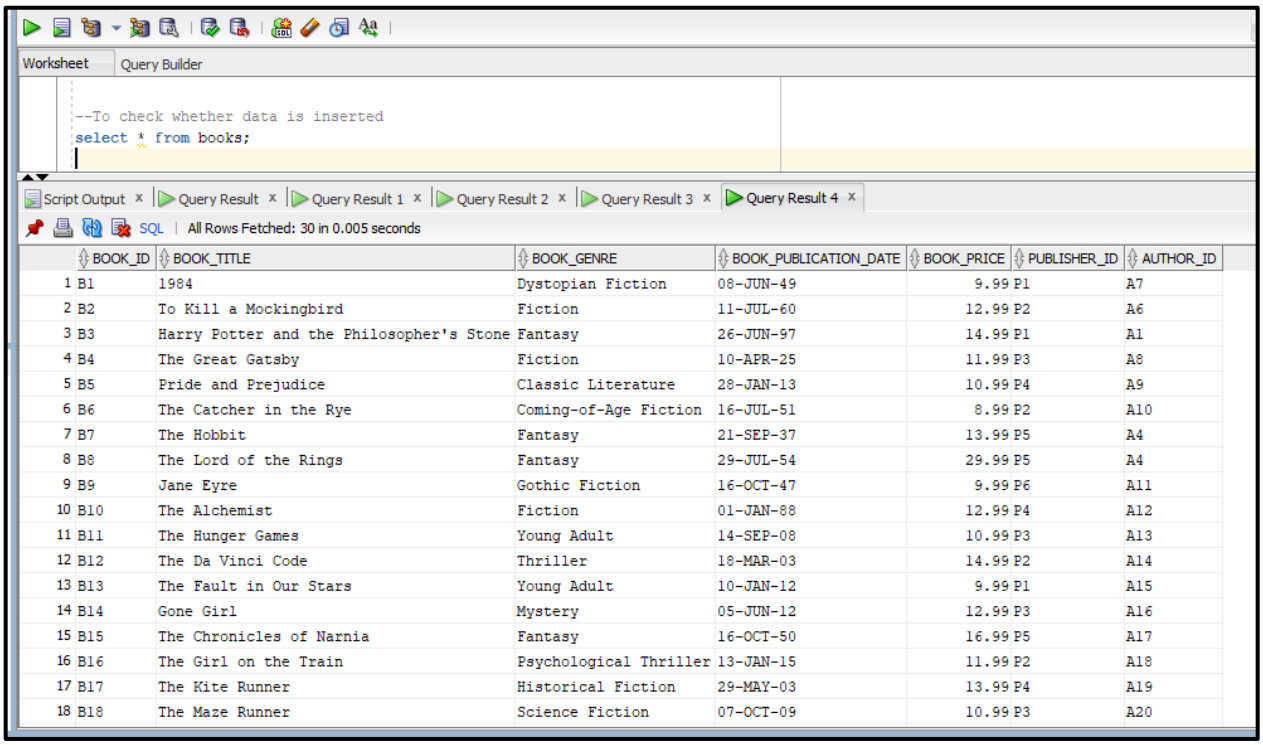


Figure 39: Displays all the Data from Customers Table

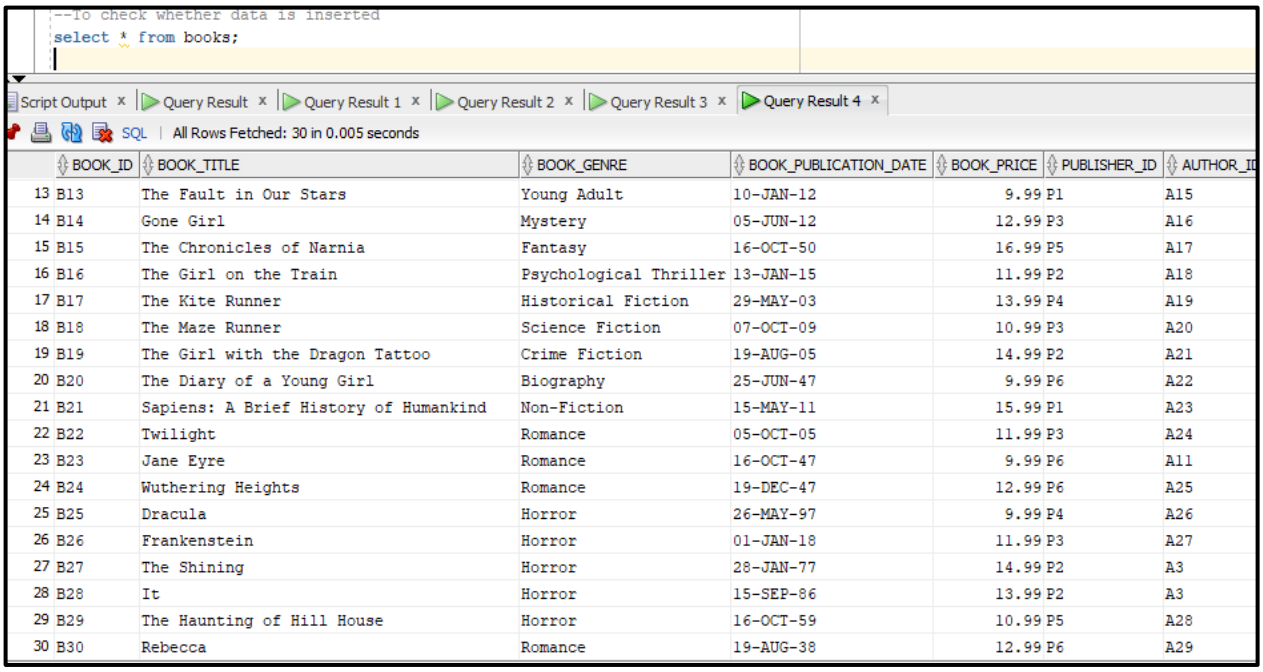
18	A18	Paula Hawkins	26-AUG-72	United Kingdom
19	A19	Khaled Hosseini	04-MAR-65	Afghanistan
20	A20	James Dashner	26-NOV-72	United States
21	A21	Stieg Larsson	15-AUG-54	Sweden
22	A22	Anne Frank	12-JUN-29	Germany
23	A23	Yuval Noah Harari	24-FEB-76	Israel
24	A24	Stephenie Meyer	24-DEC-73	United States
25	A25	Emily Brontë	30-JUL-18	United Kingdom
26	A26	Bram Stoker	08-NOV-47	Ireland
27	A27	Mary Shelley	30-AUG-97	United Kingdom
28	A28	Shirley Jackson	14-DEC-16	United States
29	A29	Daphne du Maurier	13-MAY-07	United Kingdom

Figure 41: Displays all the Data from Authors Table - 2



	BOOK_ID	BOOK_TITLE	BOOK_GENRE	BOOK_PUBLICATION_DATE	BOOK_PRICE	PUBLISHER_ID	AUTHOR_ID
1	B1	1984	Dystopian Fiction	08-JUN-49	9.99	P1	A7
2	B2	To Kill a Mockingbird	Fiction	11-JUL-60	12.99	P2	A6
3	B3	Harry Potter and the Philosopher's Stone	Fantasy	26-JUN-97	14.99	P1	A1
4	B4	The Great Gatsby	Fiction	10-APR-25	11.99	P3	A8
5	B5	Pride and Prejudice	Classic Literature	28-JAN-13	10.99	P4	A9
6	B6	The Catcher in the Rye	Coming-of-Age Fiction	16-JUL-51	8.99	P2	A10
7	B7	The Hobbit	Fantasy	21-SEP-37	13.99	P5	A4
8	B8	The Lord of the Rings	Fantasy	29-JUL-54	29.99	P5	A4
9	B9	Jane Eyre	Gothic Fiction	16-OCT-47	9.99	P6	A11
10	B10	The Alchemist	Fiction	01-JAN-88	12.99	P4	A12
11	B11	The Hunger Games	Young Adult	14-SEP-08	10.99	P3	A13
12	B12	The Da Vinci Code	Thriller	18-MAR-03	14.99	P2	A14
13	B13	The Fault in Our Stars	Young Adult	10-JAN-12	9.99	P1	A15
14	B14	Gone Girl	Mystery	05-JUN-12	12.99	P3	A16
15	B15	The Chronicles of Narnia	Fantasy	16-OCT-50	16.99	P5	A17
16	B16	The Girl on the Train	Psychological Thriller	13-JAN-15	11.99	P2	A18
17	B17	The Kite Runner	Historical Fiction	29-MAY-03	13.99	P4	A19
18	B18	The Maze Runner	Science Fiction	07-OCT-09	10.99	P3	A20

Figure 42: Displays all the Data from Books Table - 1



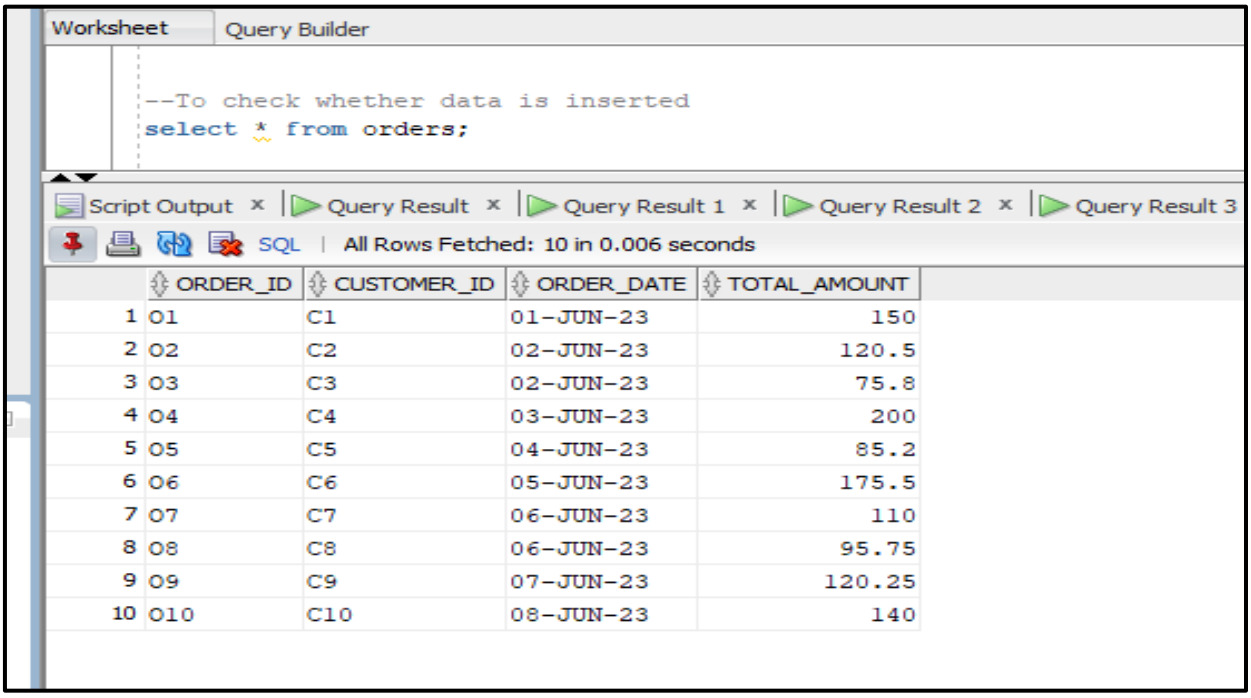
```
--To check whether data is inserted
select * from books;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3 | Query Result 4

SQL | All Rows Fetched: 30 in 0.005 seconds

	BOOK_ID	BOOK_TITLE	BOOK_GENRE	BOOK_PUBLICATION_DATE	BOOK_PRICE	PUBLISHER_ID	AUTHOR_ID
13	B13	The Fault in Our Stars	Young Adult	10-JAN-12	9.99	P1	A15
14	B14	Gone Girl	Mystery	05-JUN-12	12.99	P3	A16
15	B15	The Chronicles of Narnia	Fantasy	16-OCT-50	16.99	P5	A17
16	B16	The Girl on the Train	Psychological Thriller	13-JAN-15	11.99	P2	A18
17	B17	The Kite Runner	Historical Fiction	29-MAY-03	13.99	P4	A19
18	B18	The Maze Runner	Science Fiction	07-OCT-09	10.99	P3	A20
19	B19	The Girl with the Dragon Tattoo	Crime Fiction	19-AUG-05	14.99	P2	A21
20	B20	The Diary of a Young Girl	Biography	25-JUN-47	9.99	P6	A22
21	B21	Sapiens: A Brief History of Humankind	Non-Fiction	15-MAY-11	15.99	P1	A23
22	B22	Twilight	Romance	05-OCT-05	11.99	P3	A24
23	B23	Jane Eyre	Romance	16-OCT-47	9.99	P6	A11
24	B24	Wuthering Heights	Romance	19-DEC-47	12.99	P6	A25
25	B25	Dracula	Horror	26-MAY-97	9.99	P4	A26
26	B26	Frankenstein	Horror	01-JAN-18	11.99	P3	A27
27	B27	The Shining	Horror	28-JAN-77	14.99	P2	A3
28	B28	It	Horror	15-SEP-86	13.99	P2	A3
29	B29	The Haunting of Hill House	Horror	16-OCT-59	10.99	P5	A28
30	B30	Rebecca	Romance	19-AUG-38	12.99	P6	A29

Figure 43: Displays all the Data from Books Table - 1



Worksheet | Query Builder

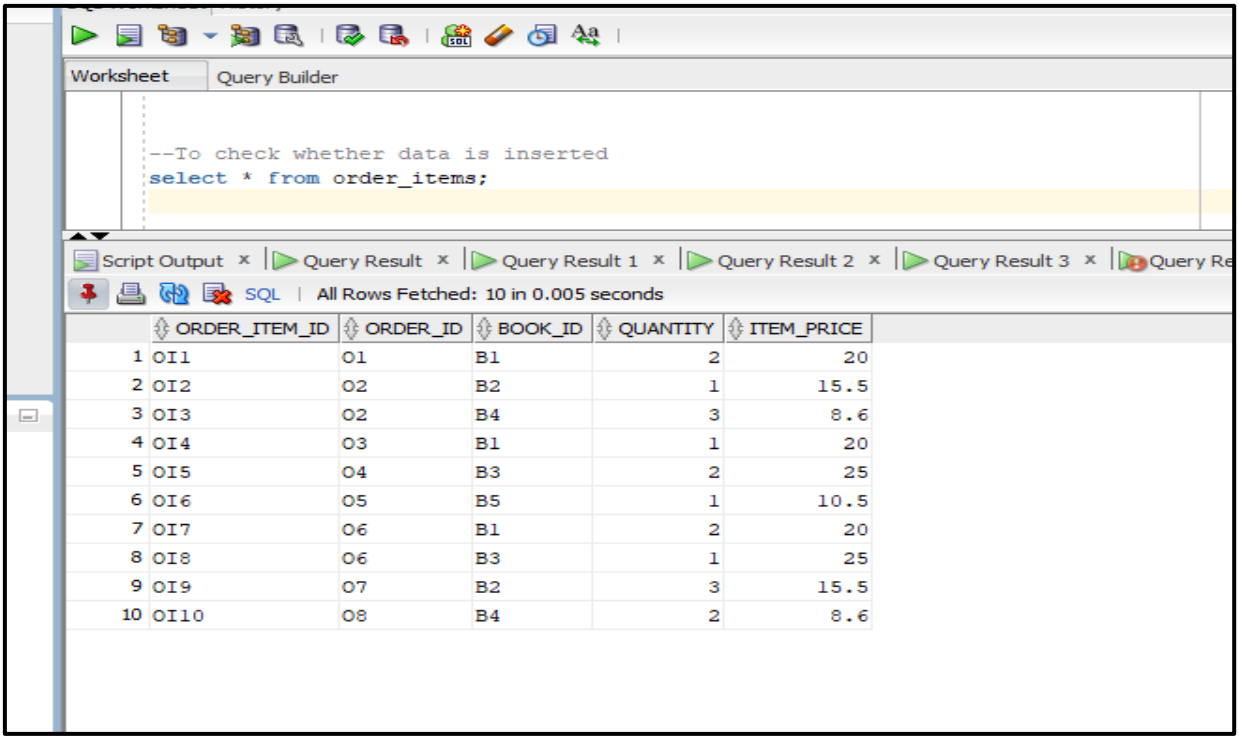
```
--To check whether data is inserted
select * from orders;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3

SQL | All Rows Fetched: 10 in 0.006 seconds

	ORDER_ID	CUSTOMER_ID	ORDER_DATE	TOTAL_AMOUNT
1	O1	C1	01-JUN-23	150
2	O2	C2	02-JUN-23	120.5
3	O3	C3	02-JUN-23	75.8
4	O4	C4	03-JUN-23	200
5	O5	C5	04-JUN-23	85.2
6	O6	C6	05-JUN-23	175.5
7	O7	C7	06-JUN-23	110
8	O8	C8	06-JUN-23	95.75
9	O9	C9	07-JUN-23	120.25
10	O10	C10	08-JUN-23	140

Figure 44: Displays all the Data from Orders Table - 2



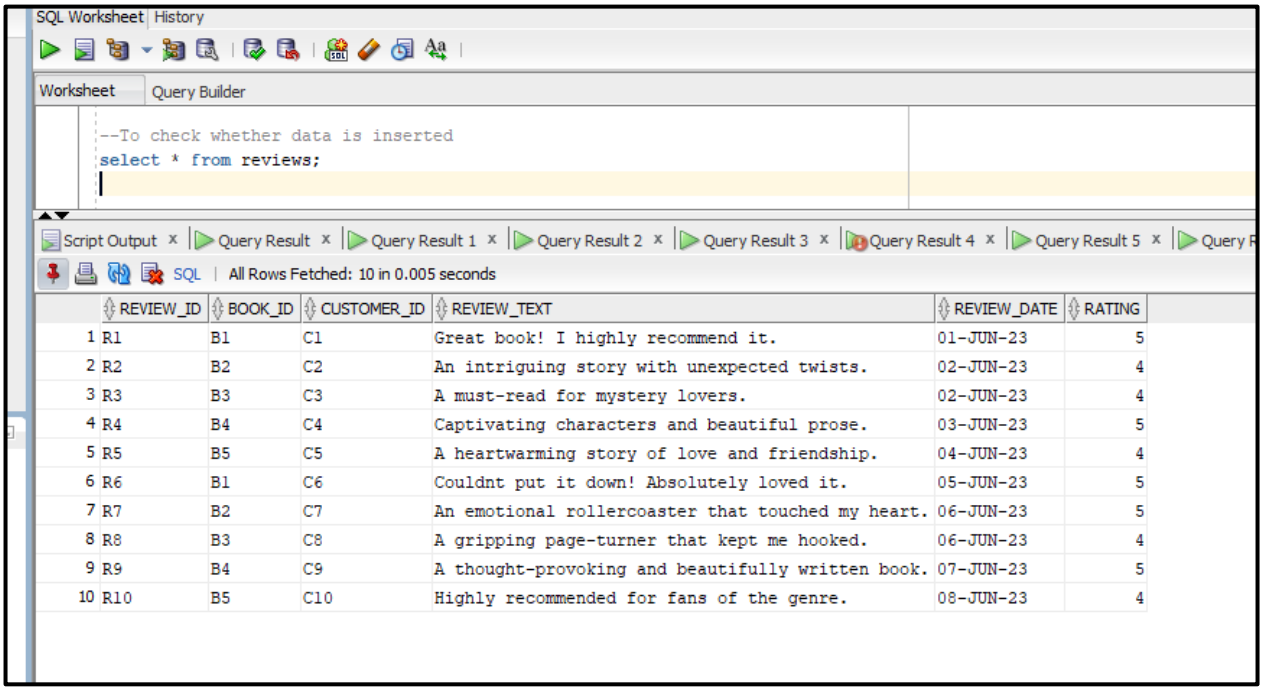
```
--To check whether data is inserted
select * from order_items;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3

SQL | All Rows Fetched: 10 in 0.005 seconds

	ORDER_ITEM_ID	ORDER_ID	BOOK_ID	QUANTITY	ITEM_PRICE
1	OI1	O1	B1	2	20
2	OI2	O2	B2	1	15.5
3	OI3	O2	B4	3	8.6
4	OI4	O3	B1	1	20
5	OI5	O4	B3	2	25
6	OI6	O5	B5	1	10.5
7	OI7	O6	B1	2	20
8	OI8	O6	B3	1	25
9	OI9	O7	B2	3	15.5
10	OI10	O8	B4	2	8.6

Figure 45: Displays all the Data from Order_Items Table - 1



```
--To check whether data is inserted
select * from reviews;
```

Script Output | Query Result | Query Result 1 | Query Result 2 | Query Result 3 | Query Result 4 | Query Result 5

SQL | All Rows Fetched: 10 in 0.005 seconds

	REVIEW_ID	BOOK_ID	CUSTOMER_ID	REVIEW_TEXT	REVIEW_DATE	RATING
1	R1	B1	C1	Great book! I highly recommend it.	01-JUN-23	5
2	R2	B2	C2	An intriguing story with unexpected twists.	02-JUN-23	4
3	R3	B3	C3	A must-read for mystery lovers.	02-JUN-23	4
4	R4	B4	C4	Captivating characters and beautiful prose.	03-JUN-23	5
5	R5	B5	C5	A heartwarming story of love and friendship.	04-JUN-23	4
6	R6	B1	C6	Couldnt put it down! Absolutely loved it.	05-JUN-23	5
7	R7	B2	C7	An emotional rollercoaster that touched my heart.	06-JUN-23	5
8	R8	B3	C8	A gripping page-turner that kept me hooked.	06-JUN-23	4
9	R9	B4	C9	A thought-provoking and beautifully written book.	07-JUN-23	5
10	R10	B5	C10	Highly recommended for fans of the genre.	08-JUN-23	4

Figure 46: Displays all the Data from Reviews Table - 1

2.3 Writing Select Queries

Question 03

Write any select queries each using where, group by, having, and order by

Query:

```
--03.
-- Select query using WHERE clause
SELECT * FROM Books WHERE book_genre = 'Fiction';


-- Select query using GROUP BY clause
SELECT book_genre, COUNT(*) AS total_books FROM Books GROUP BY book_genre;


-- Select query using HAVING clause
SELECT book_genre, COUNT(*) AS total_books FROM Books GROUP BY book_genre
HAVING COUNT(*) > 3;


-- Select query using ORDER BY clause
SELECT * FROM Books ORDER BY book_title ASC;
```

Figure 47: Query Using WHERE, GROUP BY, ORDER BY Clauses

OUTPUT RESULTS:

- Select query using WHERE clause

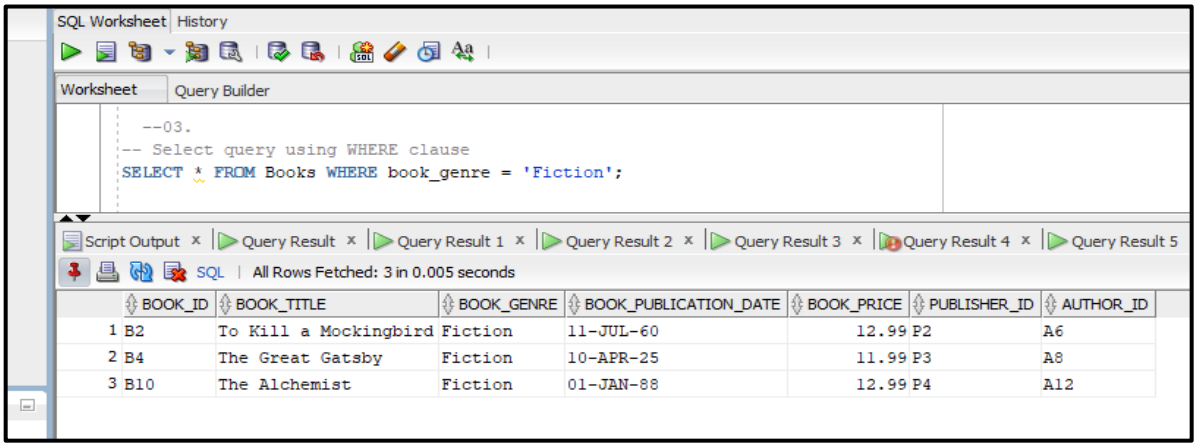


Figure 48: Displays Data for Select query using WHERE clause

- Select query using GROUP BY clause

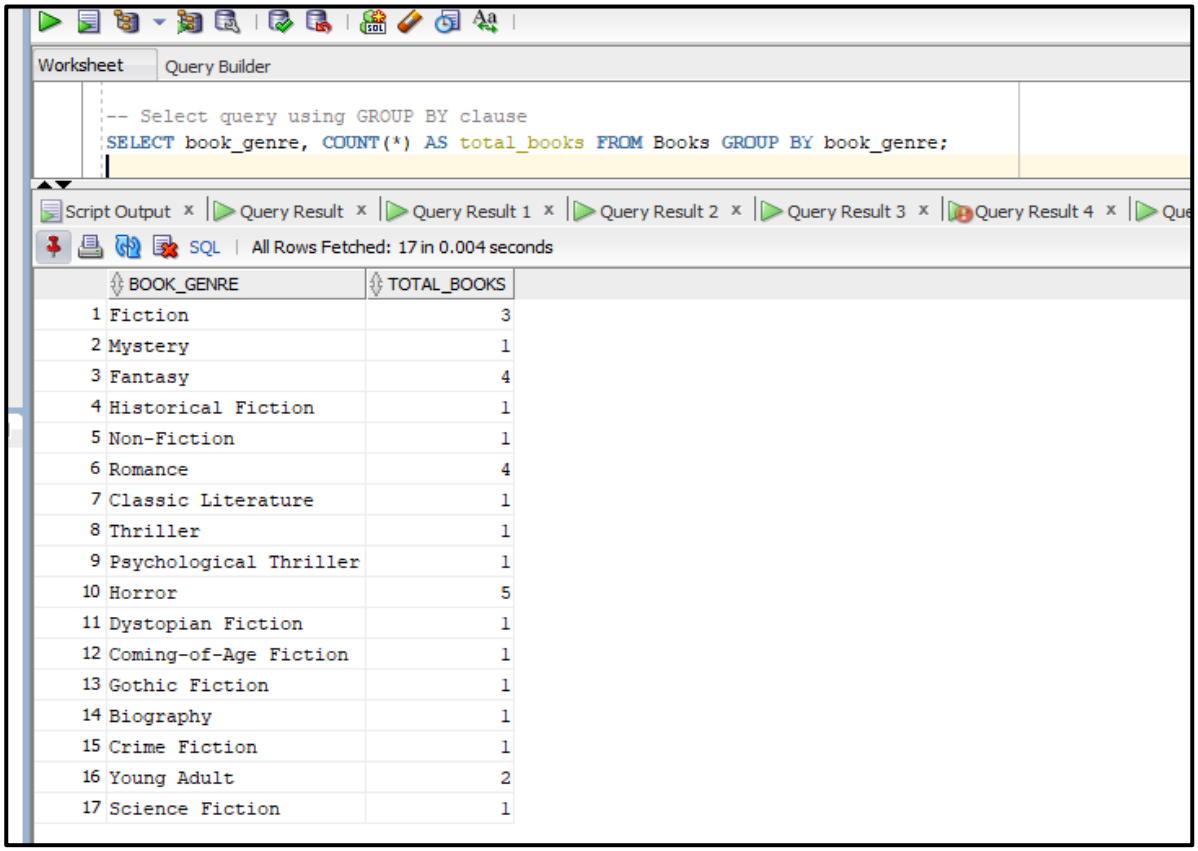
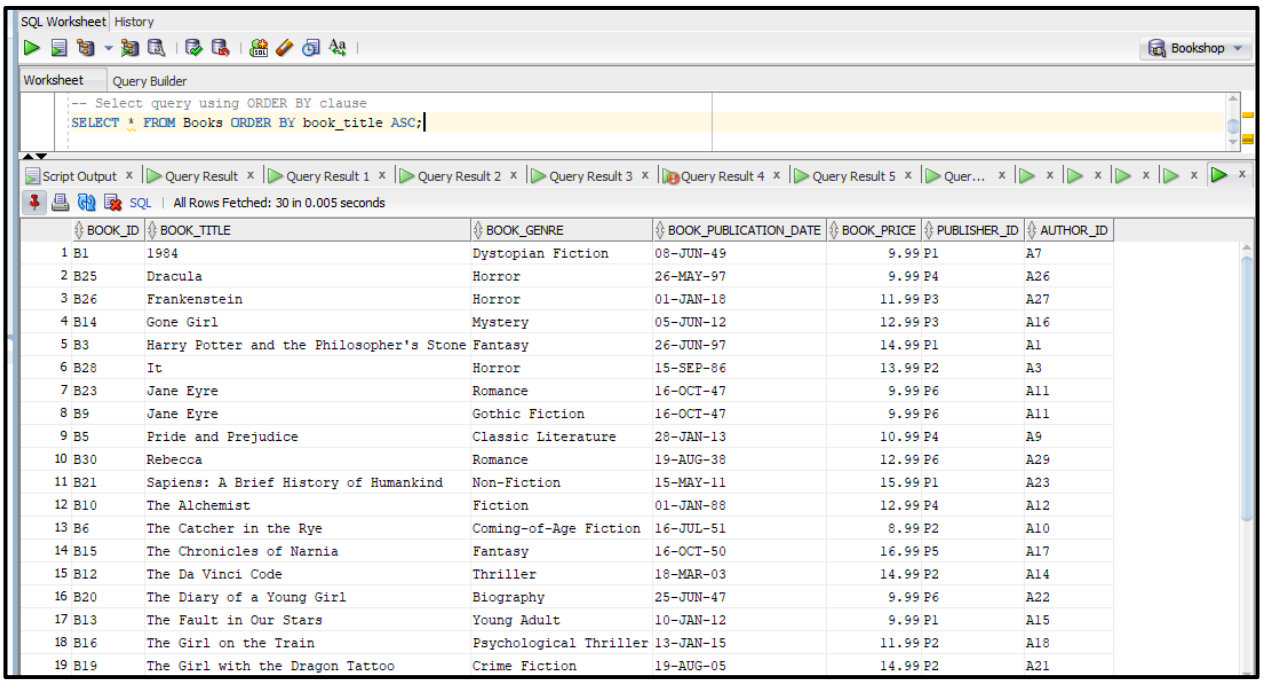


Figure 49: Displays Data for Select query using GROUP BY clause

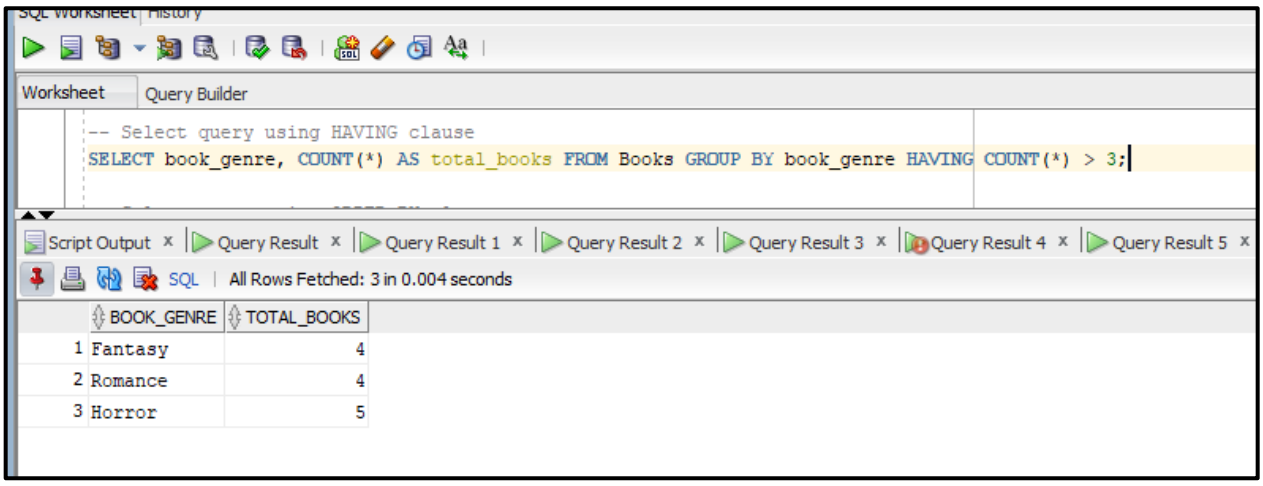
- Select query using ORDER BY clause



	BOOK_ID	BOOK_TITLE	BOOK_GENRE	BOOK_PUBLICATION_DATE	BOOK_PRICE	PUBLISHER_ID	AUTHOR_ID
1	B1	1984	Dystopian Fiction	08-JUN-49	9.99	P1	A7
2	B25	Dracula	Horror	26-MAY-97	9.99	P4	A26
3	B26	Frankenstein	Horror	01-JAN-18	11.99	P3	A27
4	B14	Gone Girl	Mystery	05-JUN-12	12.99	P3	A16
5	B3	Harry Potter and the Philosopher's Stone	Fantasy	26-JUN-97	14.99	P1	A1
6	B28	It	Horror	15-SEP-86	13.99	P2	A3
7	B23	Jane Eyre	Romance	16-OCT-47	9.99	P6	A11
8	B9	Jane Eyre	Gothic Fiction	16-OCT-47	9.99	P6	A11
9	B5	Pride and Prejudice	Classic Literature	28-JAN-13	10.99	P4	A9
10	B30	Rebecca	Romance	19-AUG-38	12.99	P6	A29
11	B21	Sapiens: A Brief History of Humankind	Non-Fiction	15-MAY-11	15.99	P1	A23
12	B10	The Alchemist	Fiction	01-JAN-88	12.99	P4	A12
13	B6	The Catcher in the Rye	Coming-of-Age Fiction	16-JUL-51	8.99	P2	A10
14	B15	The Chronicles of Narnia	Fantasy	16-OCT-50	16.99	P5	A17
15	B12	The Da Vinci Code	Thriller	18-MAR-03	14.99	P2	A14
16	B20	The Diary of a Young Girl	Biography	25-JUN-47	9.99	P6	A22
17	B13	The Fault in Our Stars	Young Adult	10-JAN-12	9.99	P1	A15
18	B16	The Girl on the Train	Psychological Thriller	13-JAN-15	11.99	P2	A18
19	B19	The Girl with the Dragon Tattoo	Crime Fiction	19-AUG-05	14.99	P2	A21

Figure 50: Displays Data for Select query using ORDER BY clause

- Select query using HAVING clause



	BOOK_GENRE	TOTAL_BOOKS
1	Fantasy	4
2	Romance	4
3	Horror	5

Figure 51: Displays Data for Select query using HAVING COUNT clauses

2.4 Writing Single and Multiple Row Queries

Question 04

Write a single-row and multiple-row subquery using the above tables.

Query:

```
--04.

-- Single-row subquery example

SELECT book_title, (SELECT author_name FROM Authors WHERE author_id =
Books.author_id) AS author_name

FROM Books;


-- Multiple-row subquery example

SELECT book_title, book_genre

FROM Books

WHERE publisher_id IN (SELECT publisher_id FROM Publishers WHERE
publisher_location = 'New York');
```

Figure 52: Query for a sub-query

OUTPUT RESULTS:

- Single-row subquery example

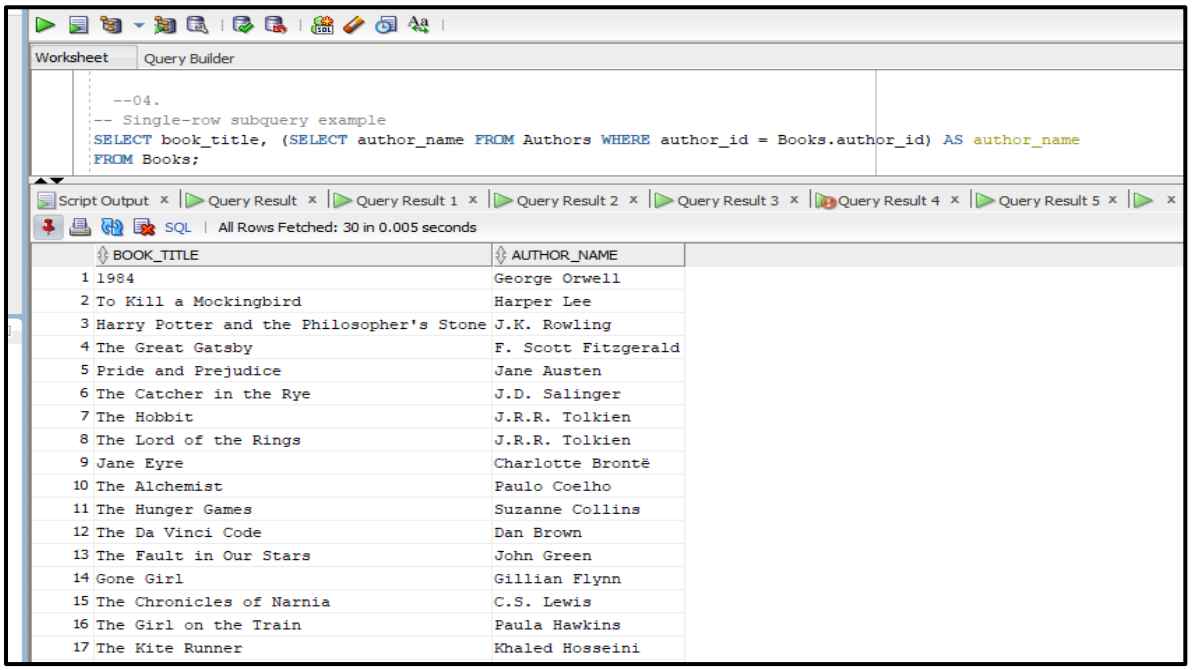


Figure 53: Single - row Subquery Example

- Multiple-row subquery example

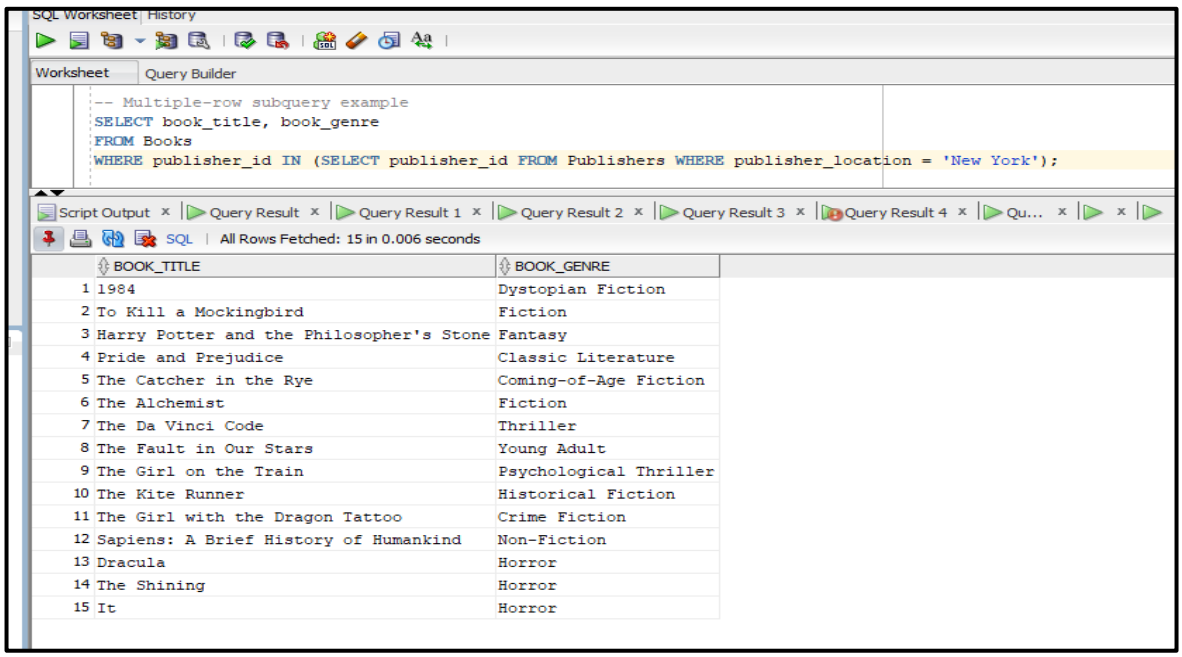


Figure 54: Multiple - row Subquery example

2.5 Writing Queries On Joins

Question 05

Write queries using left, right, and full outer joins. (There should be suitable records inserted to get the required outputs).

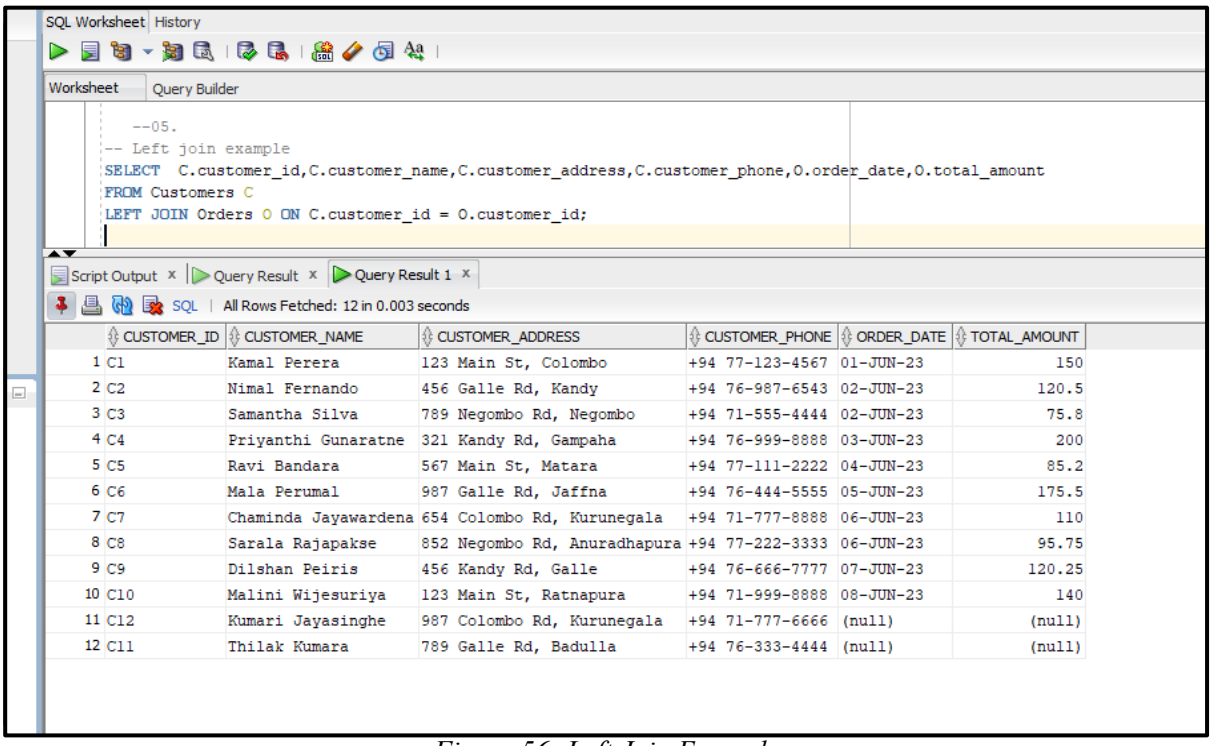
Query:

```
--05.

-- Left join example

SELECT
C.customer_id,C.customer_name,C.customer_address,C.customer_phone,O.order_date,O.total_amount

FROM Customers C

LEFT JOIN Orders O ON C.customer_id = O.customer_id;

-- Right join example

SELECT
C.customer_id,C.customer_name,C.customer_address,C.customer_phone,O.order_date,O.total_amount

FROM Customers C

RIGHT JOIN Orders O ON C.customer_id = O.customer_id;

-- Full outer join example

SELECT
C.customer_id,C.customer_name,C.customer_address,C.customer_phone,O.order_date,O.total_amount

FROM Customers C

FULL OUTER JOIN Orders O ON C.customer_id = O.customer_id;
```

Figure 55: Left, Right and Full Outer Join Examples

OUTPUT RESULTS

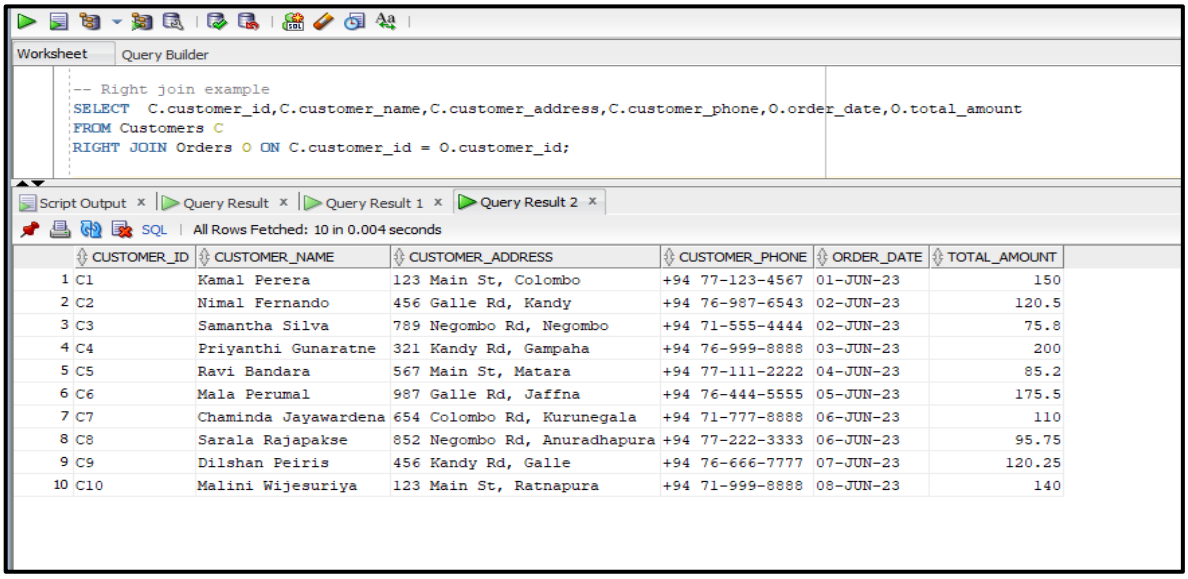
- Left join example



	CUSTOMER_ID	CUSTOMER_NAME	CUSTOMER_ADDRESS	CUSTOMER_PHONE	ORDER_DATE	TOTAL_AMOUNT
1	C1	Kamal Perera	123 Main St, Colombo	+94 77-123-4567	01-JUN-23	150
2	C2	Nimal Fernando	456 Galle Rd, Kandy	+94 76-987-6543	02-JUN-23	120.5
3	C3	Samantha Silva	789 Negombo Rd, Negombo	+94 71-555-4444	02-JUN-23	75.8
4	C4	Priyanthi Gunaratne	321 Kandy Rd, Gampaha	+94 76-999-8888	03-JUN-23	200
5	C5	Ravi Bandara	567 Main St, Matara	+94 77-111-2222	04-JUN-23	85.2
6	C6	Mala Perumal	987 Galle Rd, Jaffna	+94 76-444-5555	05-JUN-23	175.5
7	C7	Chaminda Jayawardena	654 Colombo Rd, Kurunegala	+94 71-777-8888	06-JUN-23	110
8	C8	Sarala Rajapakse	852 Negombo Rd, Anuradhapura	+94 77-222-3333	06-JUN-23	95.75
9	C9	Dilshan Peiris	456 Kandy Rd, Galle	+94 76-666-7777	07-JUN-23	120.25
10	C10	Malini Wijesuriya	123 Main St, Ratnapura	+94 71-999-8888	08-JUN-23	140
11	C12	Kumari Jayasinghe	987 Colombo Rd, Kurunegala	+94 71-777-6666	(null)	(null)
12	C11	Thilak Kumara	789 Galle Rd, Badulla	+94 76-333-4444	(null)	(null)

Figure 56: Left Join Example

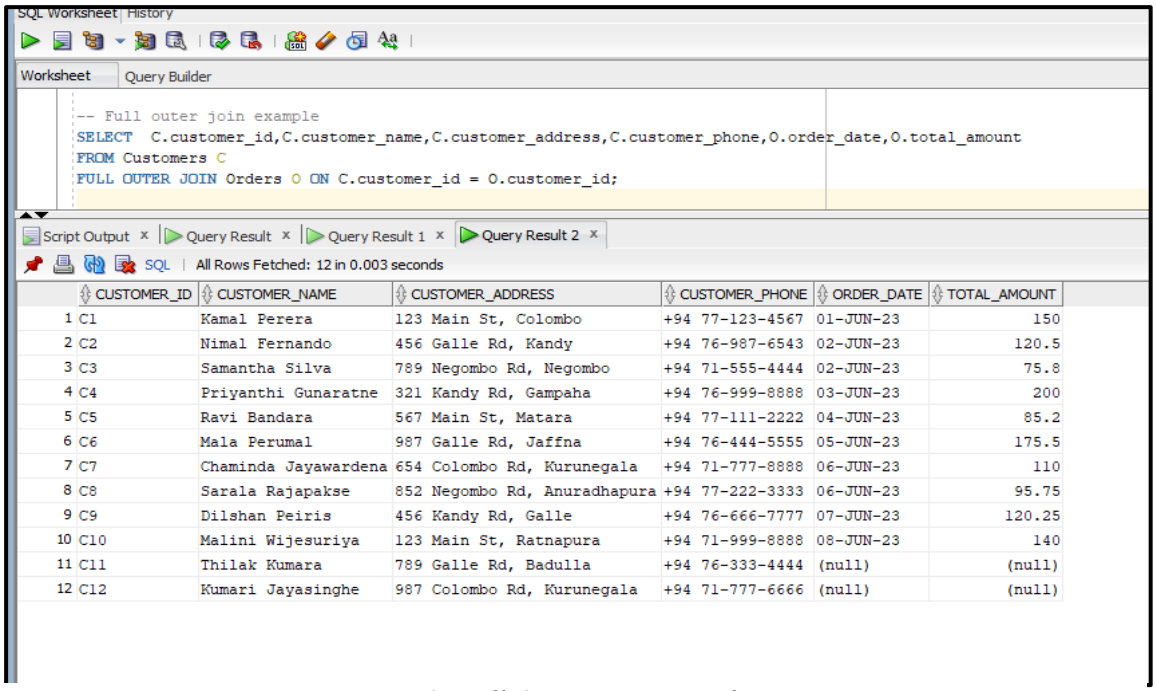
- Right join example



	CUSTOMER_ID	CUSTOMER_NAME	CUSTOMER_ADDRESS	CUSTOMER_PHONE	ORDER_DATE	TOTAL_AMOUNT
1	C1	Kamal Perera	123 Main St, Colombo	+94 77-123-4567	01-JUN-23	150
2	C2	Nimal Fernando	456 Galle Rd, Kandy	+94 76-987-6543	02-JUN-23	120.5
3	C3	Samantha Silva	789 Negombo Rd, Negombo	+94 71-555-4444	02-JUN-23	75.8
4	C4	Priyanthi Gunaratne	321 Kandy Rd, Gampaha	+94 76-999-8888	03-JUN-23	200
5	C5	Ravi Bandara	567 Main St, Matara	+94 77-111-2222	04-JUN-23	85.2
6	C6	Mala Perumal	987 Galle Rd, Jaffna	+94 76-444-5555	05-JUN-23	175.5
7	C7	Chaminda Jayawardena	654 Colombo Rd, Kurunegala	+94 71-777-8888	06-JUN-23	110
8	C8	Sarala Rajapakse	852 Negombo Rd, Anuradhapura	+94 77-222-3333	06-JUN-23	95.75
9	C9	Dilshan Peiris	456 Kandy Rd, Galle	+94 76-666-7777	07-JUN-23	120.25
10	C10	Malini Wijesuriya	123 Main St, Ratnapura	+94 71-999-8888	08-JUN-23	140

Figure 57: Right Join Example

- Full outer join example



	CUSTOMER_ID	CUSTOMER_NAME	CUSTOMER_ADDRESS	CUSTOMER_PHONE	ORDER_DATE	TOTAL_AMOUNT
1	C1	Kamal Perera	123 Main St, Colombo	+94 77-123-4567	01-JUN-23	150
2	C2	Nimal Fernando	456 Galle Rd, Kandy	+94 76-987-6543	02-JUN-23	120.5
3	C3	Samantha Silva	789 Negombo Rd, Negombo	+94 71-555-4444	02-JUN-23	75.8
4	C4	Priyanthi Gunaratne	321 Kandy Rd, Gampaha	+94 76-999-8888	03-JUN-23	200
5	C5	Ravi Bandara	567 Main St, Matara	+94 77-111-2222	04-JUN-23	85.2
6	C6	Mala Perumal	987 Galle Rd, Jaffna	+94 76-444-5555	05-JUN-23	175.5
7	C7	Chaminda Jayawardena	654 Colombo Rd, Kurunegala	+94 71-777-8888	06-JUN-23	110
8	C8	Sarala Rajapakse	852 Negombo Rd, Anuradhapura	+94 77-222-3333	06-JUN-23	95.75
9	C9	Dilshan Peiris	456 Kandy Rd, Galle	+94 76-666-7777	07-JUN-23	120.25
10	C10	Malini Wijesuriya	123 Main St, Ratnapura	+94 71-999-8888	08-JUN-23	140
11	C11	Thilak Kumara	789 Galle Rd, Badulla	+94 76-333-4444	(null)	(null)
12	C12	Kumari Jayasinghe	987 Colombo Rd, Kurunegala	+94 71-777-6666	(null)	(null)

Figure 58: Full Outer Join Example

2.6 Creating Views

Question 06

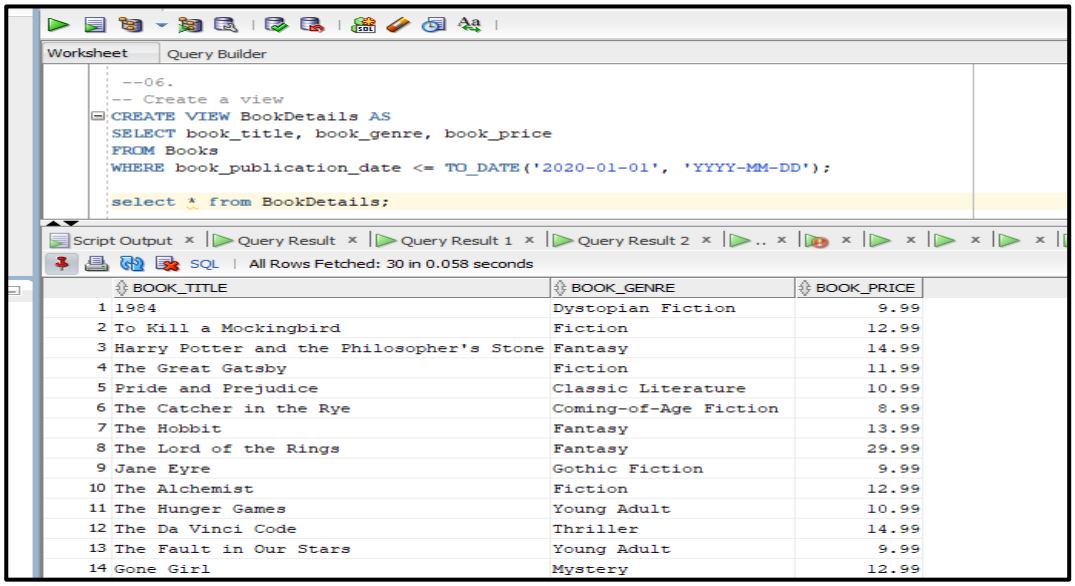
Create a view using one of the tables created.

Query:

```
--06.

-- Create a view

CREATE VIEW BookDetails AS

SELECT book_title, book_genre, book_price

FROM Books

WHERE book_publication_date <= TO_DATE('2020-01-01', 'YYYY-MM-DD');


select * from BookDetails;
```

Figure 59: Query for Creating a View

OUTPUT RESULTS:



	BOOK_TITLE	BOOK_GENRE	BOOK_PRICE
1	1984	Dystopian Fiction	9.99
2	To Kill a Mockingbird	Fiction	12.99
3	Harry Potter and the Philosopher's Stone	Fantasy	14.99
4	The Great Gatsby	Fiction	11.99
5	Pride and Prejudice	Classic Literature	10.99
6	The Catcher in the Rye	Coming-of-Age Fiction	8.99
7	The Hobbit	Fantasy	13.99
8	The Lord of the Rings	Fantasy	29.99
9	Jane Eyre	Gothic Fiction	9.99
10	The Alchemist	Fiction	12.99
11	The Hunger Games	Young Adult	10.99
12	The Da Vinci Code	Thriller	14.99
13	The Fault in Our Stars	Young Adult	9.99
14	Gone Girl	Mystery	12.99

Figure 60: Output of the VIEW Created

2.7 PL/ SQL Block To Retrieve A Record For Specific Input.

Question 07

Write a PL/ SQL block to retrieve a record for specific input..

Query:

```
SET SERVEROUTPUT ON;
DECLARE
  v_book_title Books.book_title%TYPE;
  v_book_genre Books.book_genre%TYPE;
  v_book_date Books.book_publication_date%TYPE;
  v_book_price Books.book_price%TYPE;
  v_book_publisher Books.publisher_id%TYPE;
  v_book_author Books.author_id%TYPE;
  v_book_id Books.book_id%TYPE:='&p_book_id';
BEGIN
  SELECT book_title,book_genre,book_publication_date,book_price,publisher_id,author_id
  INTO v_book_title,v_book_genre,v_book_date,v_book_price,v_book_publisher,v_book_author
  FROM Books WHERE book_id =v_book_id ;
  DBMS_OUTPUT.PUT_LINE('BOOK DETAILS:.....');
  DBMS_OUTPUT.PUT_LINE('Book Title: ' || v_book_title);
  DBMS_OUTPUT.PUT_LINE('Book Genre: ' || v_book_genre);
  DBMS_OUTPUT.PUT_LINE('Book Published Date: ' || v_book_date);
  DBMS_OUTPUT.PUT_LINE('Book Price: ' || v_book_price);
  DBMS_OUTPUT.PUT_LINE('Book Publisher ID: ' || v_book_publisher);
  DBMS_OUTPUT.PUT_LINE('Book Author ID: ' || v_book_author);
END;
/
```

Figure 61: Retrieve a code for a specific input code

- Here we are taking user input as the search ID.

OUTPUT RESULTS:

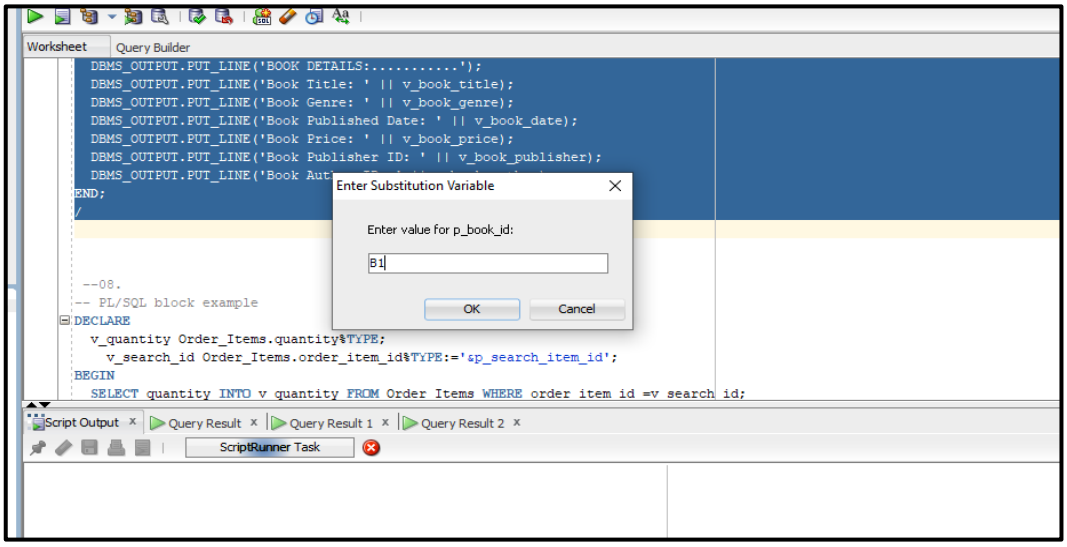


Figure 62: Output to retrieve data from input Example

Finally Book Title is given:

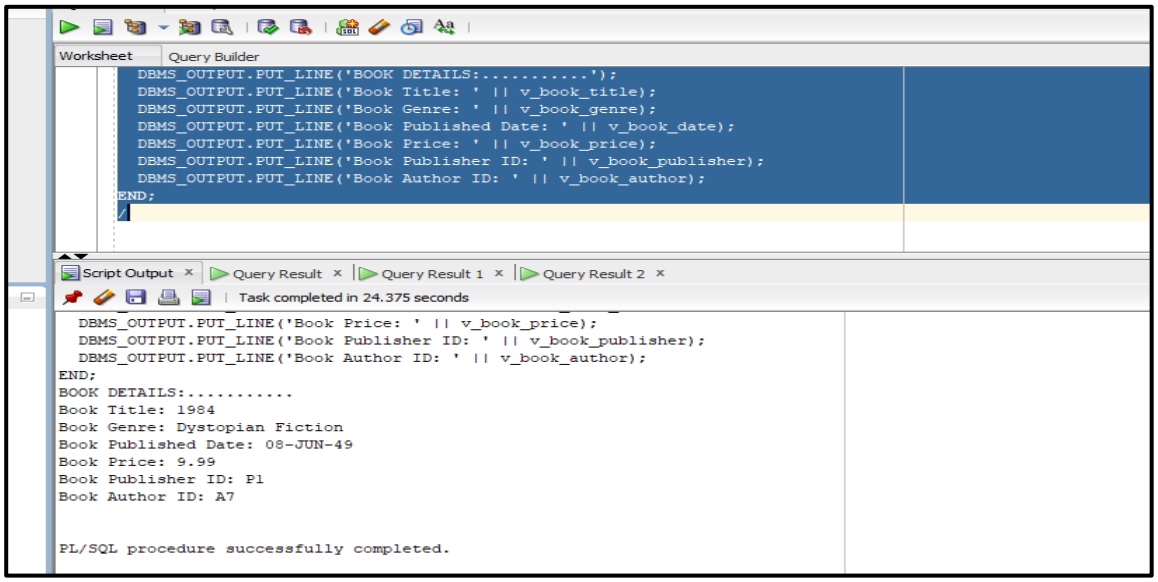


Figure 63: Display Output for the Retrieved Data Example

2.8 Write A PL/ SQL Block To Update A Record For Specific Input.

Question 08

Write a PL/ SQL block to update a record for specific input

Query:

```
--08.
-- PL/SQL block example
DECLARE
  v_quantity Order_Items.quantity%TYPE;
    v_search_id Order_Items.order_item_id%TYPE:='&p_search_item_id';
BEGIN
  SELECT quantity INTO v_quantity FROM Order_Items WHERE order_item_id
=v_search_id;
  IF v_quantity < 10 THEN
    UPDATE Order_Items SET quantity = v_quantity + 1 WHERE order_item_id =
v_search_id;
    DBMS_OUTPUT.PUT_LINE('Quantity updated.');
  ELSE
    DBMS_OUTPUT.PUT_LINE('Maximum quantity reached.');
  END IF;
END;
/
```

- Here, we take user input through substitution variables

OUTPUT RESULTS:

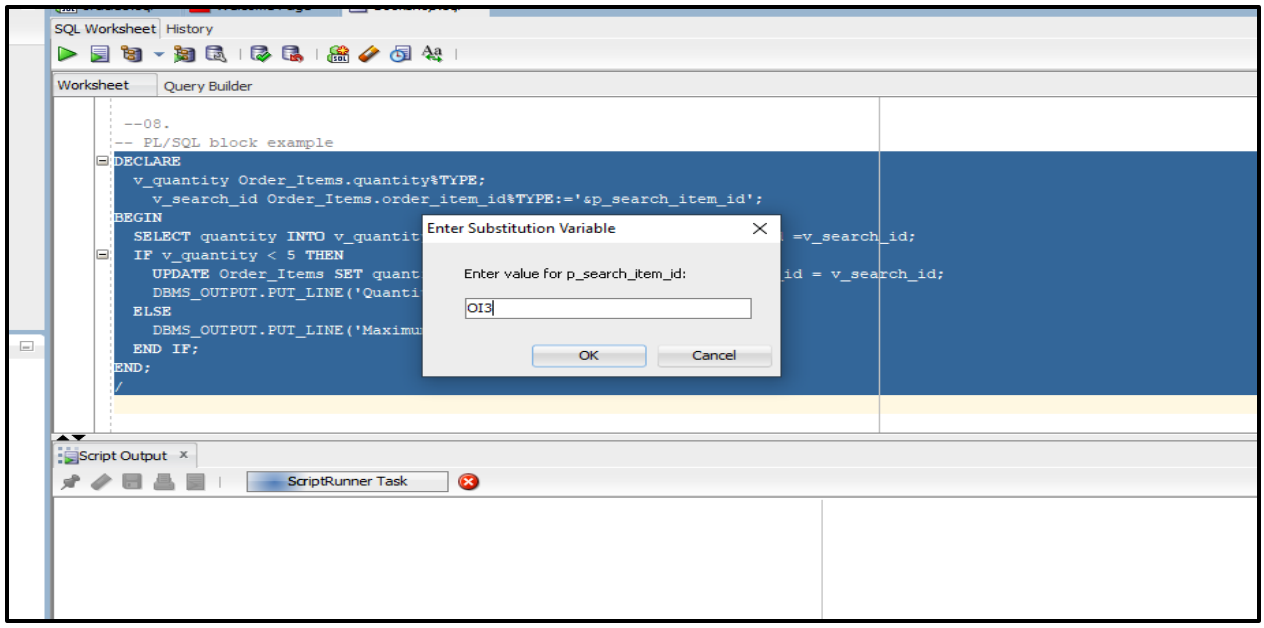


Figure 64: Substitution Variables Example

Finally It is updated:

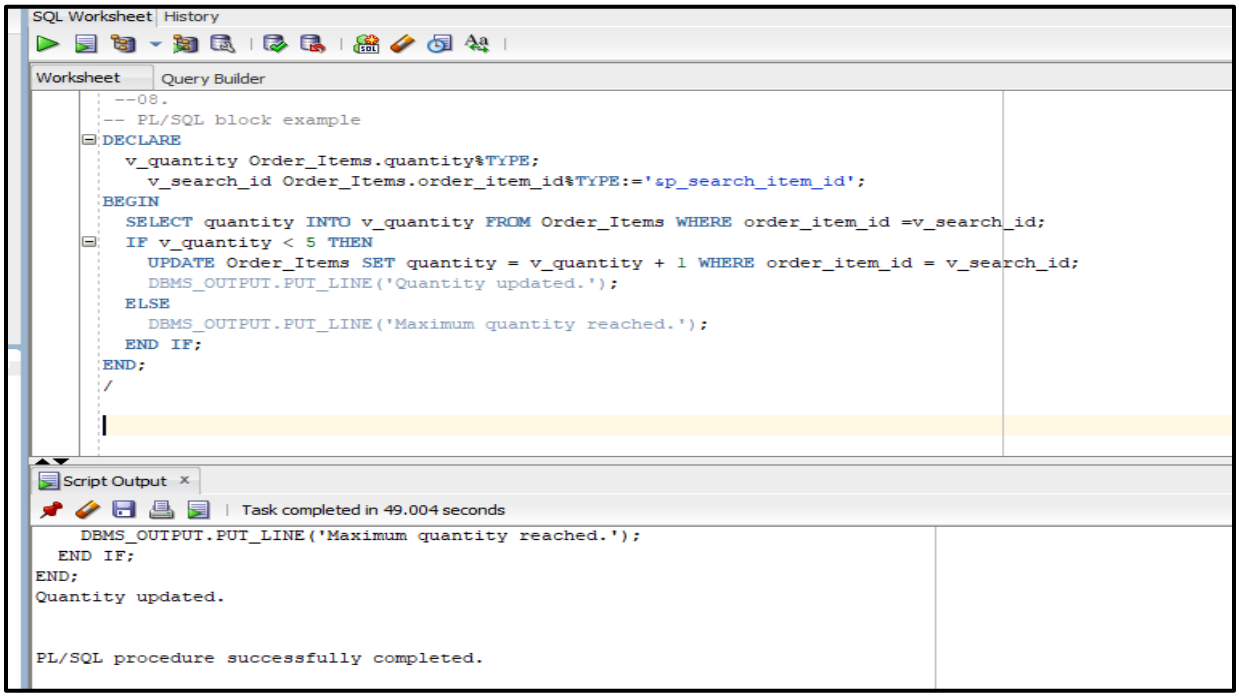


Figure 65: Updated Substitution Example

2.9 Write A PL/ SQL Block To Delete A Record For Specific Input

Question 09

Write a PL/ SQL block to delete a record for specific input

Query:

```
--09.
-- PL/SQL block example
DECLARE
  v_order_id Orders.order_id%TYPE;
   v_search_id Orders.order_id%TYPE:='&p_search_id';
BEGIN
  SELECT order_id INTO v_order_id FROM Orders WHERE order_id =v_search_id;
  DELETE FROM Order_Items WHERE order_id = v_order_id;
  DELETE FROM Orders WHERE order_id = v_order_id;
  DBMS_OUTPUT.PUT_LINE('Order deleted.');
EXCEPTION
  WHEN NO_DATA_FOUND THEN
    DBMS_OUTPUT.PUT_LINE('Order not found.');
END;
/
```

Figure 66: Delete a specific Input Code

- Here we have taken user inputs to get the order id.

OUTPUT RESULTS:

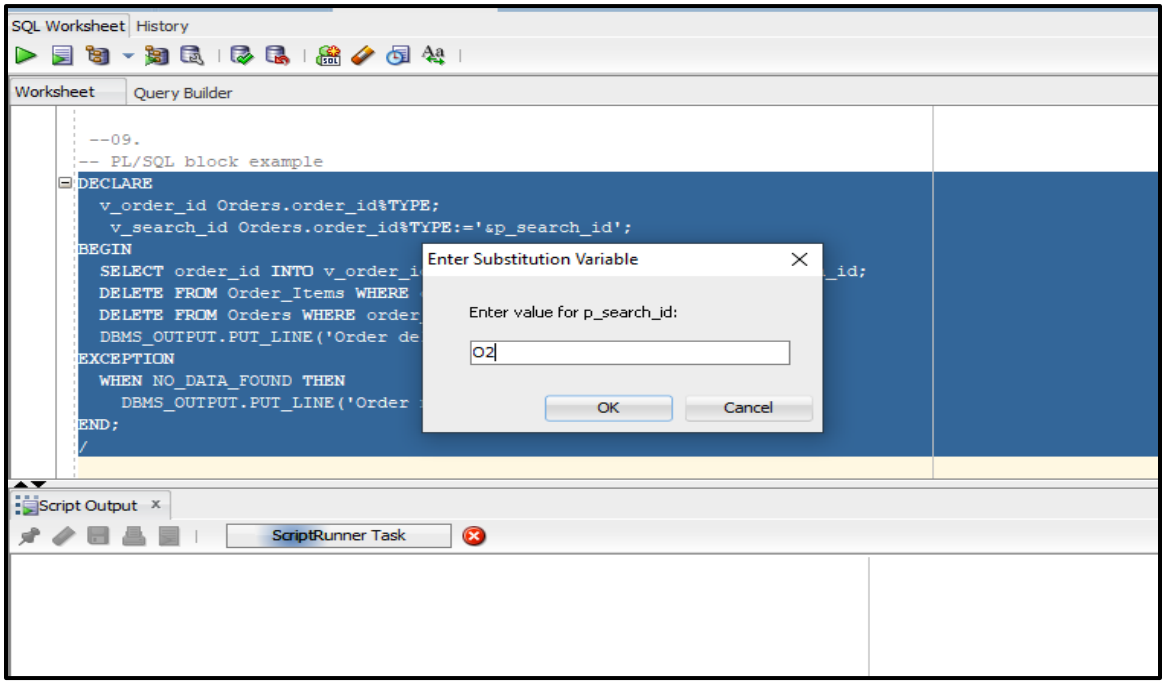


Figure 67: Delete Specific Input Output

- Finally Order deleted successfully:

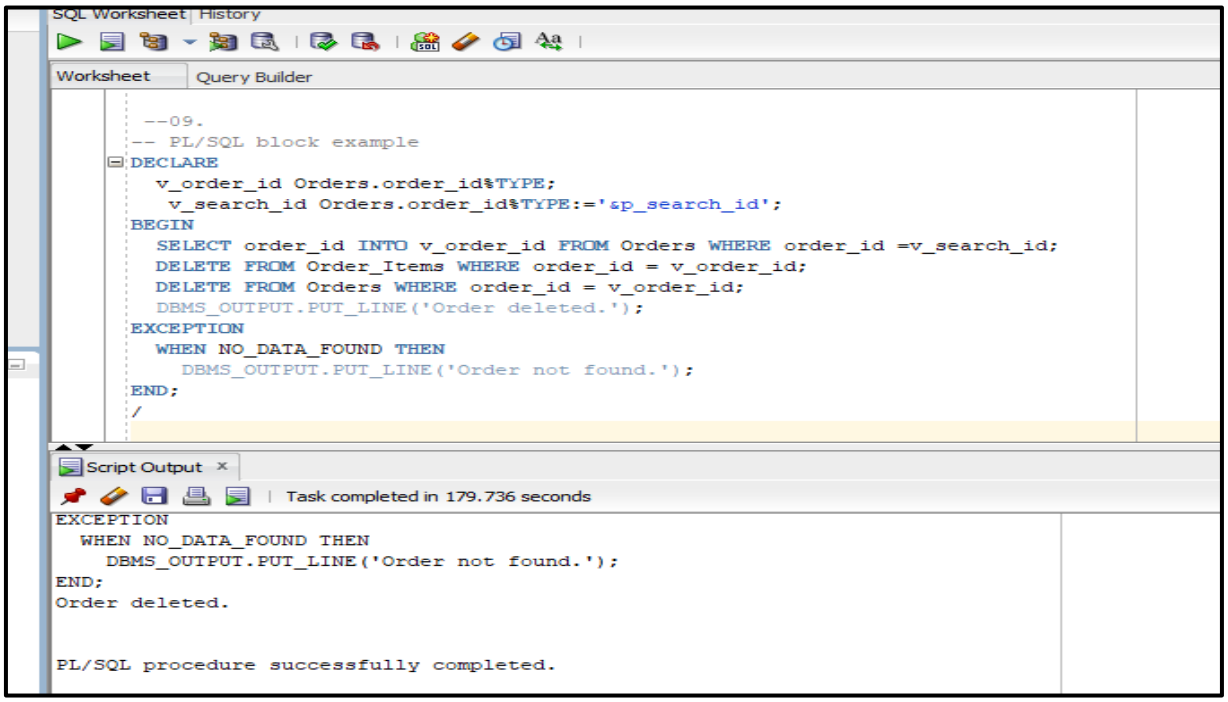


Figure 68: Delete Code Output Example

2.10 Modifying Query To Display The Number Of Rows Deleted

Question 10

Modify the above query to display the number of rows deleted

Query:

```
--10.
-- PL/SQL block example
DECLARE
  v_num_rows NUMBER;
  v_order_id Order_Items.order_id%TYPE:='&p_order_id';
BEGIN
  DELETE FROM Order_Items WHERE order_id = v_order_id;
  v_num_rows := SQL%ROWCOUNT;
  DBMS_OUTPUT.PUT_LINE(v_num_rows || ' rows deleted.');
END;
/
```

Figure 69: Display No. of rows deleted Query

OUTPUT RESULTS:

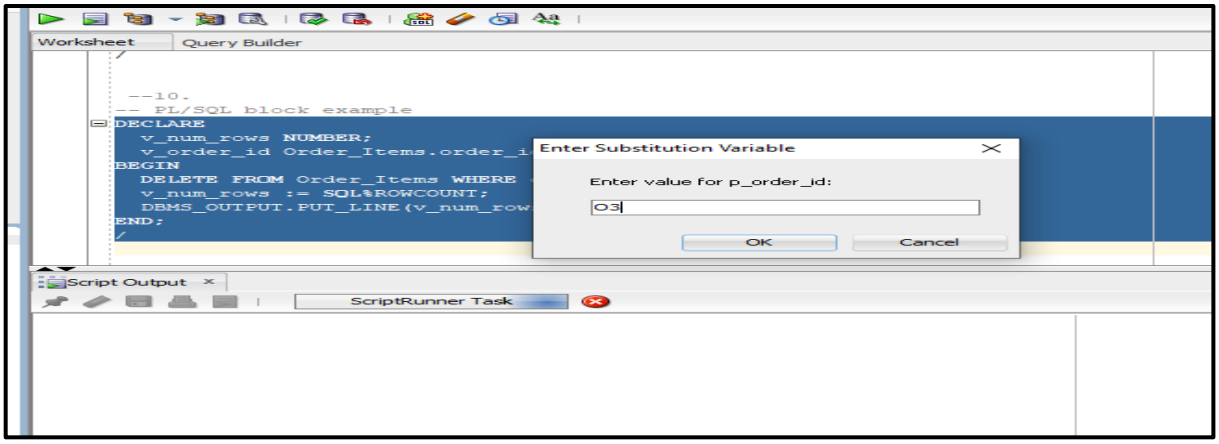


Figure 70: No. of Rows Deleted Output

- Finally after deleting it displays the number of rows deleted.

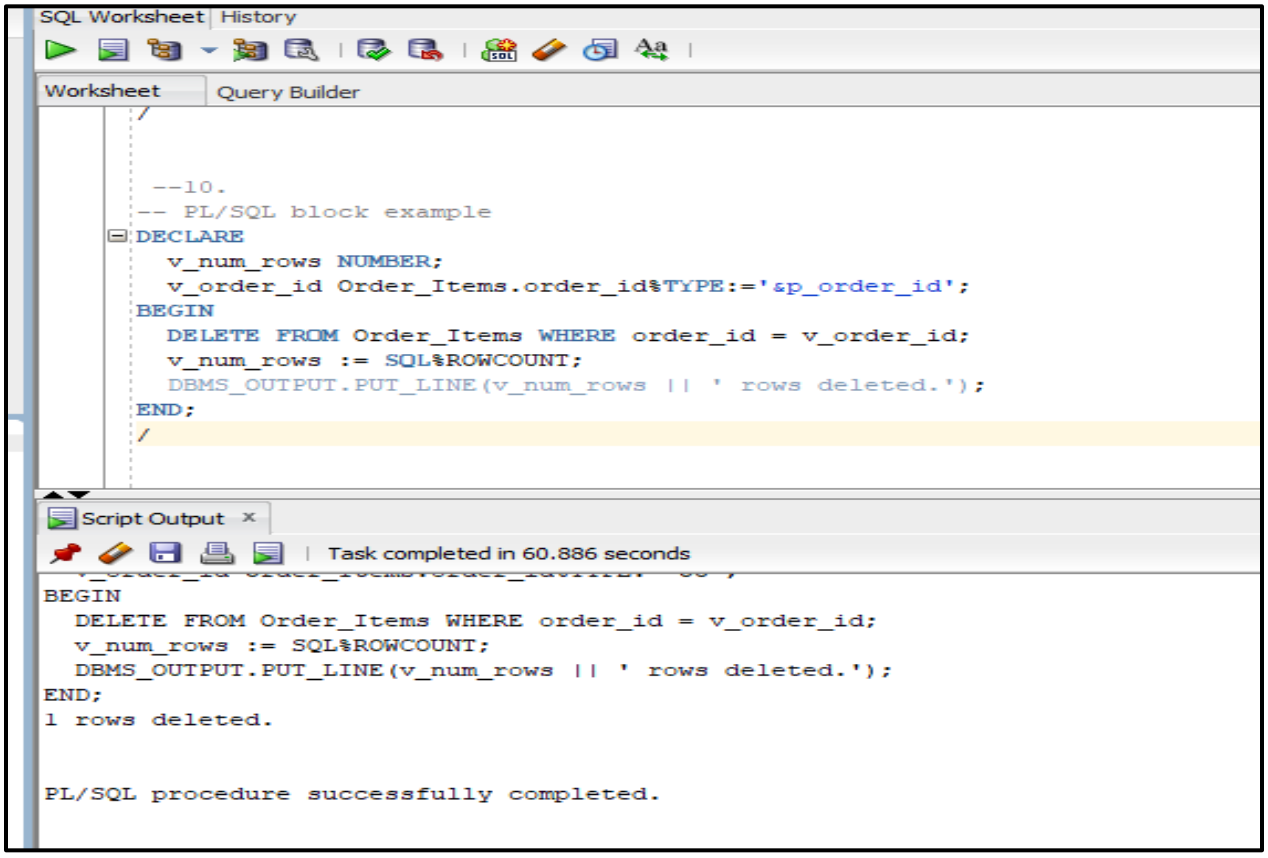


Figure 71: Output for the Rows Deleted

CHAPTER 3: CONCLUSION

In conclusion, the implementation of an Oracle database for The Book Corner, an independent bookstore, has proven to be a significant milestone in modernizing their operations and enhancing overall efficiency. By leveraging the power of Oracle's robust features and capabilities, the bookstore has successfully transitioned from manual record-keeping to a streamlined, automated system. The Oracle database has provided The Book Corner with numerous benefits, including improved data management, enhanced data security, and increased scalability. The ability to store and retrieve vast amounts of book-related information with ease has significantly improved the bookstore's inventory management processes, allowing for better tracking of book availability, sales, and customer preferences.

Additionally, the Oracle database has facilitated seamless integration with other systems and applications used by The Book Corner, such as their online store and customer relationship management (CRM) software. This integration has enabled real-time data synchronization, eliminating data discrepancies and ensuring consistent and accurate information across various channels. Furthermore, the implementation of an Oracle database has enhanced data security measures for The Book Corner.

Lastly, the scalability of the Oracle database has future-proofed The Book Corner's operations, allowing for seamless expansion as the bookstore grows. The ability to handle increasing volumes of data and accommodate the evolving needs of the business ensures that the database will continue to support the bookstore's operations for years to come.

Overall, the implementation of an Oracle database has transformed The Book Corner's operations, optimizing their processes, improving data management, and enhancing customer experiences. By embracing modern technology and leveraging the capabilities of Oracle's robust database solution, the bookstore has positioned itself for continued success in an increasingly competitive market.

Chapter 4: References

- Bibliography: Oracle Tutorial. (2023). Learn to program. Retrieved from https://www.oracletutorial.com/plsql-tutorial/ (Accessed June 17 2023) In-Line Citation :(Oracle Tutorial 2022)
- Bibliography: Tutorialspoint. (2023). Learn to code. Retrieved from https://www.tutorialspoint.com/plsql/index.htm (Accessed June 17 2023) In-Line Citation : (Tutorialspoint 2022)
- Bibliography: Code Academy. (2023). Learn to code with free online courses. Retrieved from https://www.codecademy.com/ (Accessed March 30 2023) In-Line Citation :(Code Academy 2023)
- Bibliography: javatpoint. (2023). Learn to program. Retrieved from https://www.javatpoint.com/pl-sql-tutorial (Accessed June 18 2023) In-Line Citation :(Java point 2023)
- Bibliography: Techonthenet. (2023). Learn to program. Retrieved from https://www.techonthenet.com/oracle/index.php (Accessed June 18 2023) In-Line Citation :(Tech on the net 2022)