Inserting Data

# Key Points

* Create all tables **before** inserting data.
* To use a sequence for populating the primary key values, the sequence needs to be created.
* We can have only one sequence with the same name in the same schema.
* Place the drop sequence statement into the script section where you drop tables and other database objects.
* Insert all data into a parent table before the child table. A child table is a table with the foreign key referencing the parent table.
* Each insert statement inserts one row into a table. To insert 10 rows into a table, you need to write 10 insert statements.
* Make sure that the spelling of the table name in the insert statement matches the spelling of the table name in the create table section.
* Make sure that the spelling of the column names in the insert statement matches the spelling of the column names in the create table statement.
* List the column names in parentheses before the values keyword.
* The number of values under the values keyword needs to match the number of columns above the values keyword.
* The order of column names needs to match the order of corresponding values. For example, the first value under the values keyword is inserted into a first column listed above values keyword.
* The data type of a value needs to match the data type of the corresponding column. For example, insert numeric data into numeric columns. Insert character data into varchar2 and char columns. Insert dates into date columns.
* Enclose character values in quotes.
* For varchar2 and char columns, the number of characters in a value cannot exceed the number of characters defined in the create table statement. For instance, we cannot insert the 21 characters long value into a column declared as varcahr2 (20).
* Use to\_date function to insert values into the date columns.
* We cannot insert more than one row with the same primary key value into the same table.
* Figure 1 shows an example of an Oracle error message in attempt to insert the value into a foreign key that does not have a matching value of the primary key in the parent table.

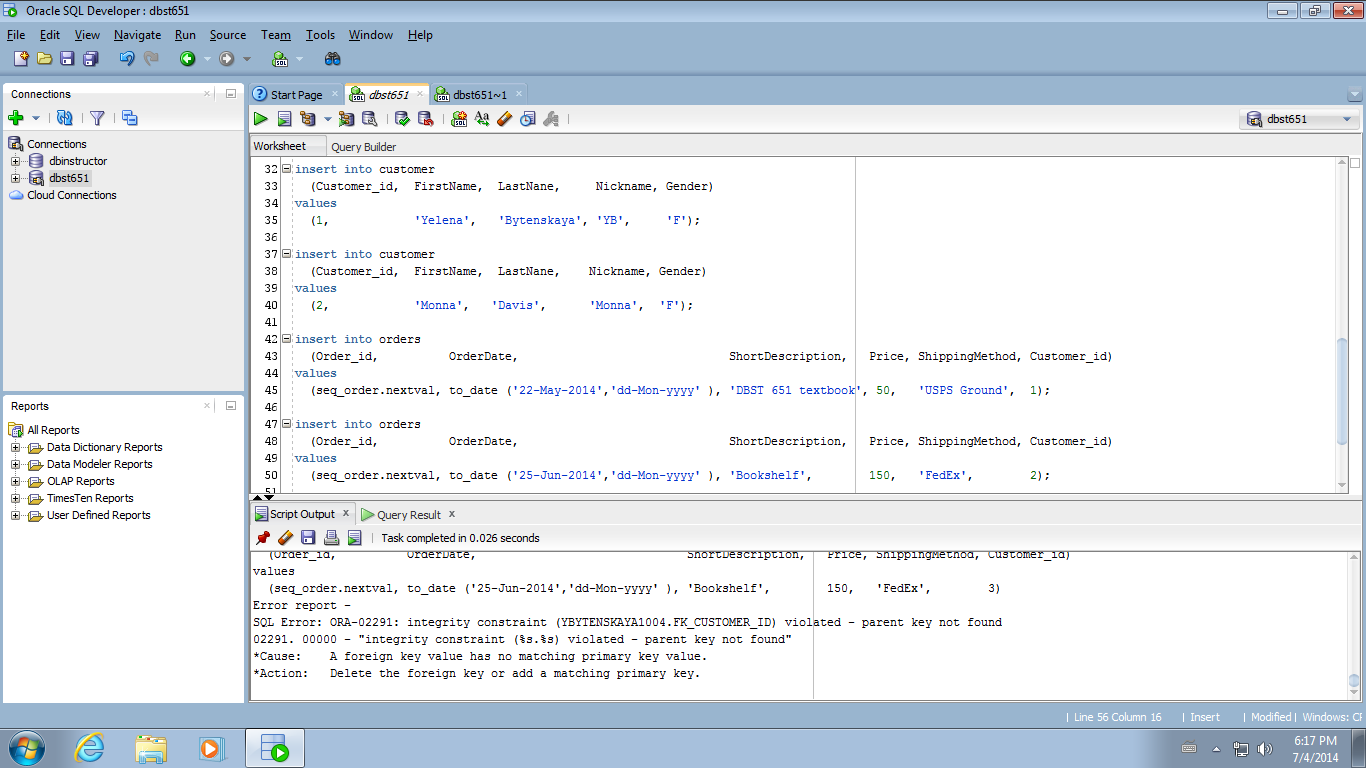


Figure 1: Integrity constraint error message

# Example

Figure 2 is the script to drop and create tables and sequence. The drop sequence statement on line 3 follows the drop table statements on line 1 and 2.

The parent table customer (line 5-13) is created before the child table orders (line 15-25). The orders table has a foreign key defined on customer\_id column (line 24).

The create sequence statement (line 27-29) follows the create table statement.

The sequence name in the drop sequence statement on line 3 matches the sequence name in the create sequence statement on line 27.

Start with 1 on line 28 means that the generated values start from 1. Increment by 1 on line 29 means that we add 1 to the current value to generate the next value. (1, 2, 3, …)

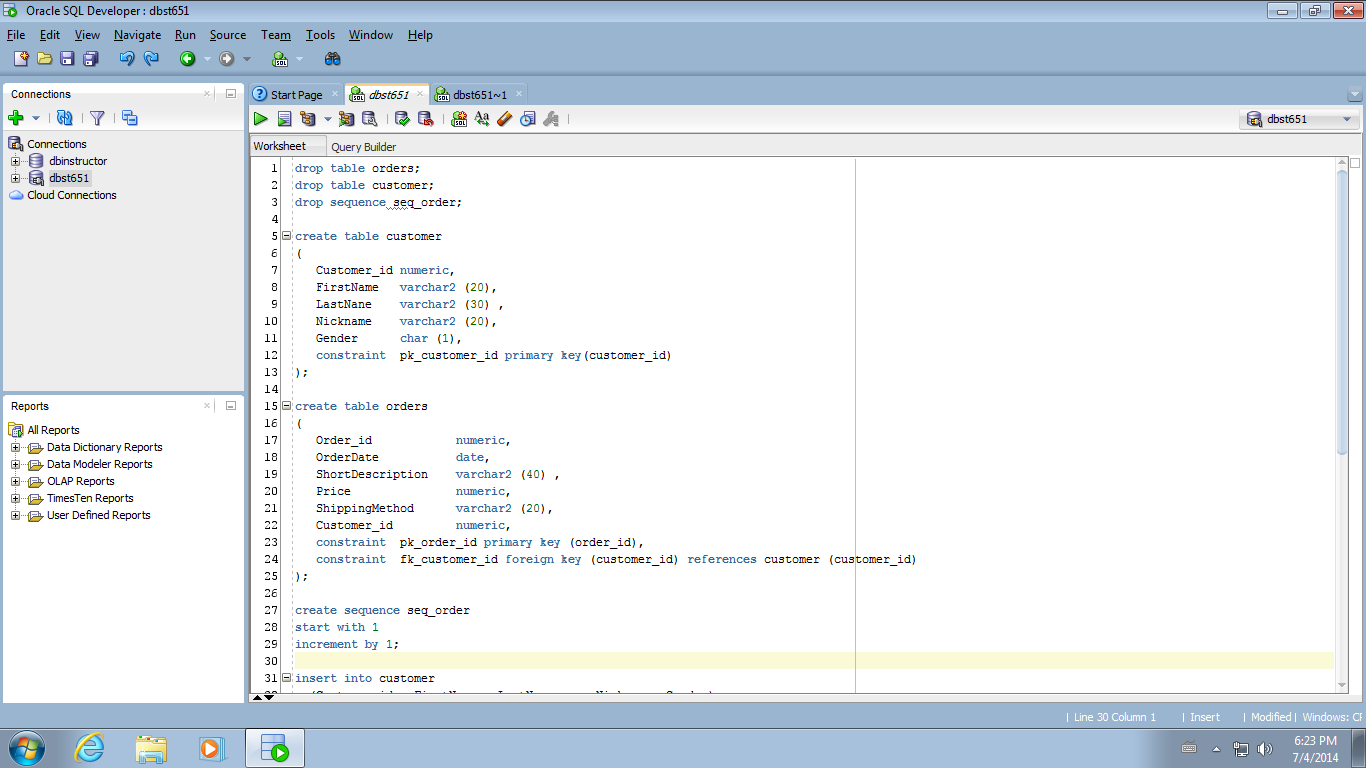


Figure 2: DDL to drop and create tables and sequence

The insert statements on Figure 3 follow the create table statements above. We insert two rows into a customer table and two rows into an order table.

Lines 32-35 are the statement to insert the first row into a customer table. Line 33 is the list of column names. Line 35 is the list of corresponding values.

Lines 37-40 are the statement to insert the second row into the customer table. Since customer\_id is a primary key, we cannot insert the second row with the same customer\_id value.

The insert statements for the child table orders (lines 42-50) follow the insert statements for the parent table customer.

On line 45 we use a sequence to populate the value of a primary key.

We use to\_date function to insert the value into a date column. For example,

to\_date ('22-May-2014','dd-Mon-yyyy' )

The first value in parentheses is the date that we insert. The second value in parentheses specifies the format. For instance, dd means that the first 2 characters in '22-May-2014' is the day of the month. Mon stands for the month abbreviation, and yyyy stands for 4 digits year.

The last value on line 45 is the value of the foreign key column customer\_id. If an insert statement into a customer table on lines 32-35 ran successfully, we have a customer with customer\_id=1. Hence, 1 is the valid customer\_id value of a in orders table.

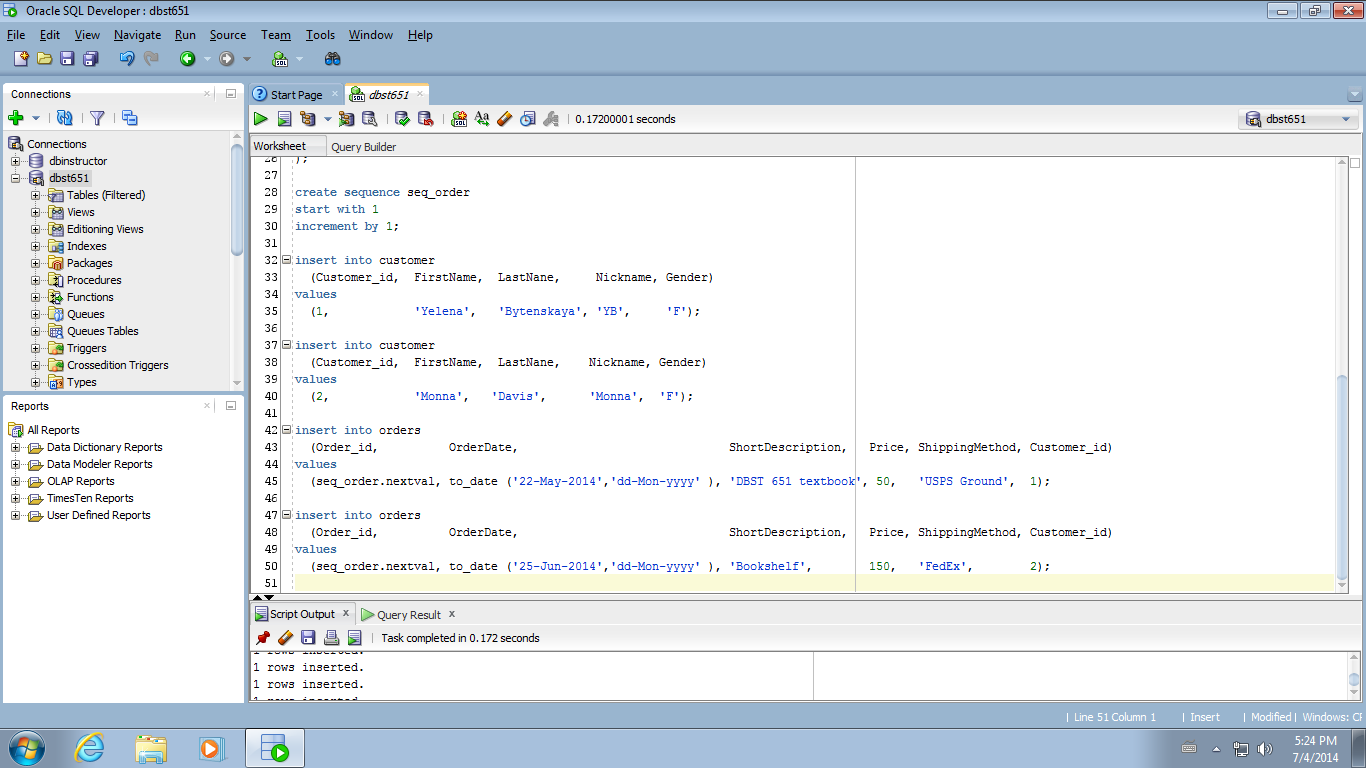


Figure 3: Insert statements

To verify that the insert statements work, we run the select statement on **Figure 4** to display all data from a customer table. We run the insert statement on **Figure 5** to display all data from orders table.

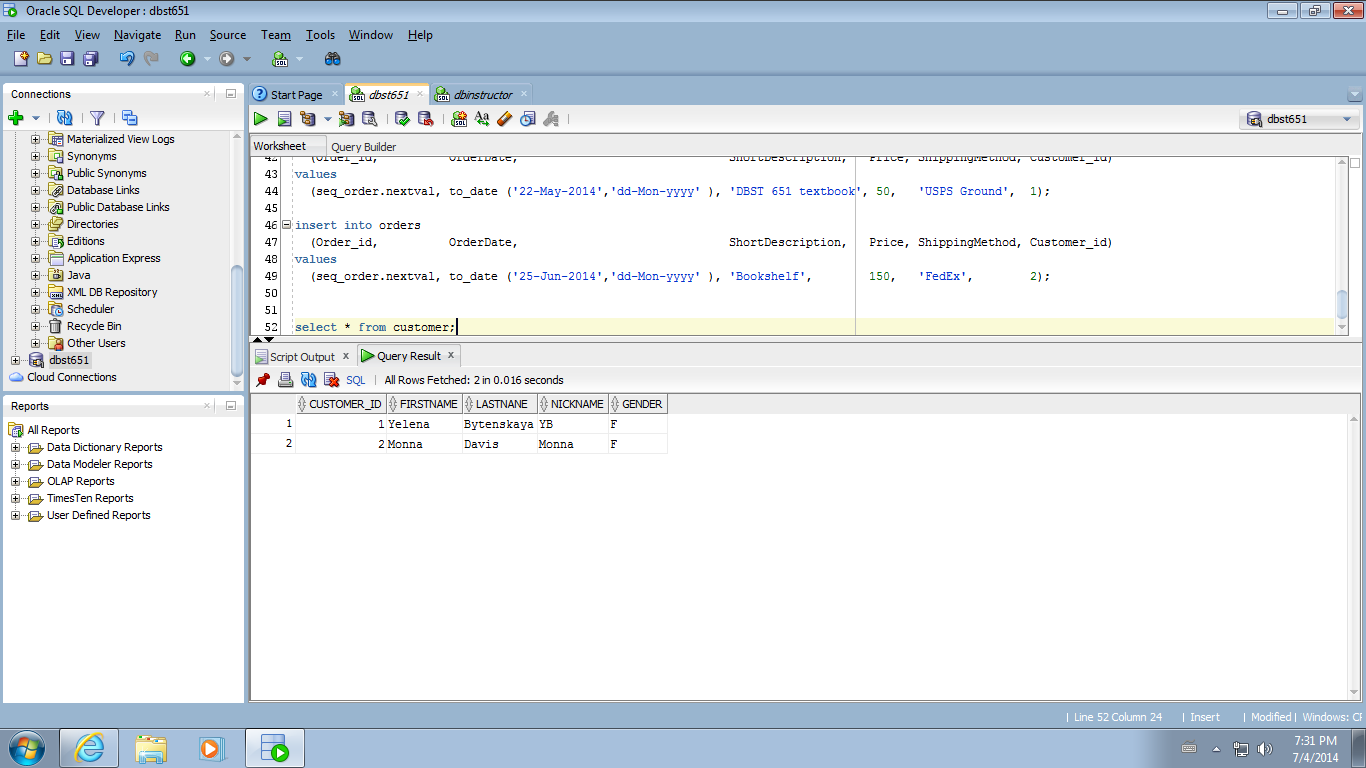


Figure 4: Data inserted into customer

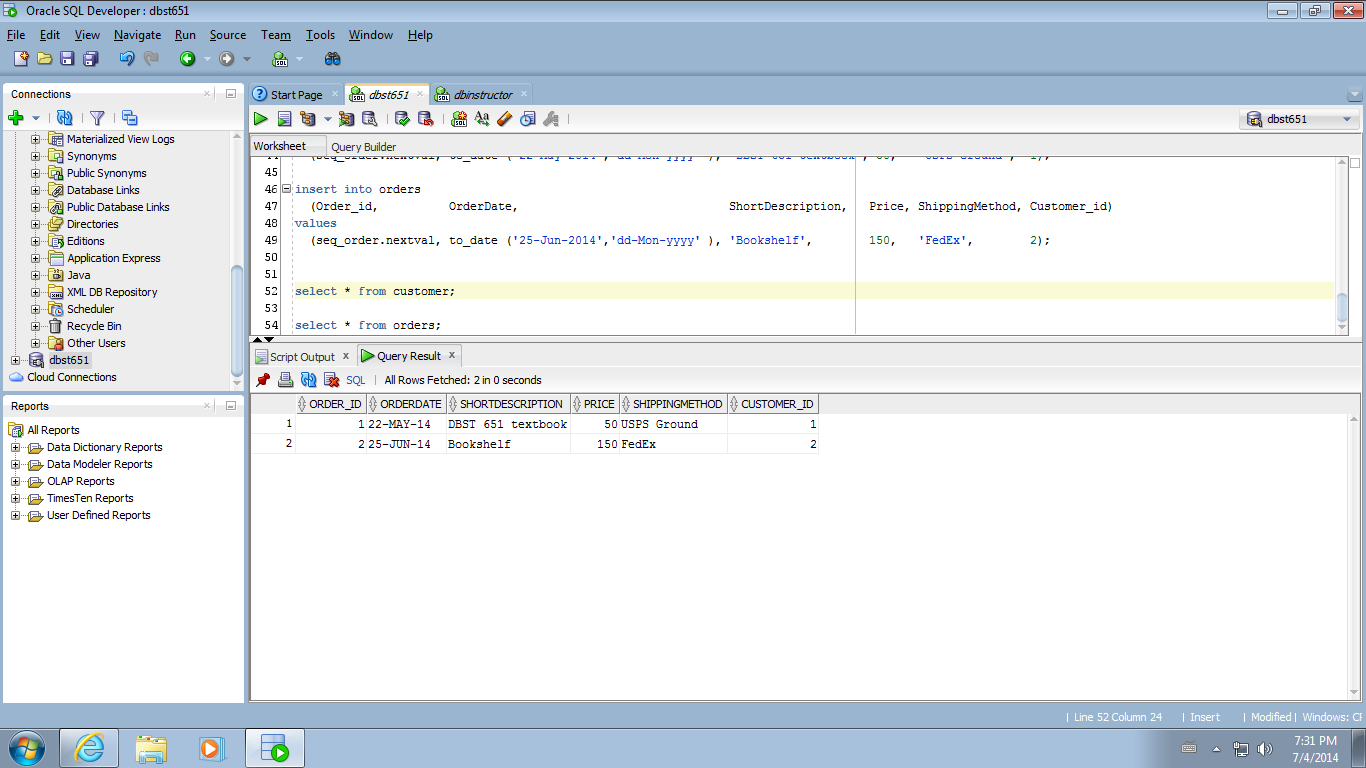


Figure 5: Data inserted into orders