



# Broward Community College

## Course Outline

STATUS:   A  

COMMON COURSE NUMBER:           COP 2740C          

COURSE TITLE:           Introduction to Oracle: SQL and PL/SQL          

CREDIT HOURS:                   4                  

**CONTACT HOURS BREAKDOWN:**

Lecture/Discussion                   56                  

Lab                                   8                                  

Other                                   0                                  

Contact Hours/Week                   4                  

**CATALOG COURSE DESCRIPTION:**

Prerequisite: CIS2342 and COP1334C

Corequisite: None

This course offers students an extensive introduction to data server technology. This class covers the concepts of relational databases and the powerful SQL and PL/SQL programming languages. Students are taught to create and maintain database objects and to store, retrieve and manipulate data. In addition, students learn to create PL/SQL blocks of application code that can be shared by multiple forms, reports, and data management applications. Demonstrations and hands-on practice reinforce the fundamental concepts. This class is designed to prepare students to successfully complete the Oracle Application Developer and Database Administrator certification exams.

**UNIT TITLES:**

1. Introduction to Oracle SQL and PL/SQL
2. Writing a Basic SQL Statement
3. Restricting and Sorting Data
4. Single Row Functions
5. Displaying Data from Multiple Tables (Relational Database)
6. Aggregating Data using Group Functions
7. Writing Sub-queries
8. Multi-Column Sub-queries
9. Producing Readable Output with SQL \*Plus
10. Manipulating Data
11. Creating & Managing Tables
12. Including Constraints

13. Creating View
14. Other Database Objects
15. Controlling User Access
16. SQL Workshop
17. Declaring Variables
18. Writing Executable Statements
19. Interacting with the Oracle Server
20. Writing Control Structures

## **I. Course Overview:**

Upon successful completion of this course, the students should be able to create and maintain database objects, control data/user access, use basic SQL \*Plus commands, describe relational database concepts, retrieve, insert update, and delete data, describe the features and syntax of PL/SQL, control transactions, conditionally control code flow (loops, control structures and explicit cursors) and handle runtime errors.

## **II. Units:**

### **Unit 1. Introduction to Oacle SQL and PL/SQL**

#### General Outcome:

- 1.0 The students should be able to use oracle functions as a database and describe the basic components. They will also know the basic purpose of SQL and PL/SQL.

#### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 1.1 Describe the basic components of Oracle
- 1.2 Discuss how Oracle serves as a database
- 1.3 Discuss client/server database
- 1.4 Describe a data server
- 1.5 Explain the purposes of SQL and PL/SQL
- 1.6 Discuss Oracle CASE.
- 1.7 Describe Oracle database structure
- 1.8 Describe tablespaces
- 1.9 Discuss transaction logs
- 1.10 Describe an Oracle instance

## Unit 2. Writing a Basic SQL Statement

### General Outcome:

2.0 The students should be able to code SQL statements using the select.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

2.1 Discuss the goal of the select statement.

2.2 Code a simple select statement to display a table.

2.3 Code a select statement to display selected fields from a table.

## Unit 3. Restricting and Sorting Data

### General Outcome:

3.0 The students should be able to restrict records to be displayed and sort the data to be displayed.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 3.1 Display specific records based on a single restriction
- 3.2 Display specific fields within records based on a single restriction.
- 3.3 Display specific records based on multiple restrictions.
- 3.4 Display specific fields within records based on multiple restrictions.
- 3.5 Perform ascending sort on a single field with and without restrictions.
- 3.6 Perform descending sort on a single field with and without restrictions.
- 3.7 Perform ascending sort on two fields with and without restrictions.
- 3.8 Perform descending sort on two fields with and without restrictions.
- 3.9 Sort primary field ascending, and a secondary field descending.

## Unit 4. Single Row Functions

### General Outcome:

4.0 The students should be able to extract one record based on one or several restrictive values.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 4.1 Display one record based on the primary key value.
- 4.2 Display specific fields from one record based on the primary key value.
- 4.3 Display one record based on multiple conditions set.
- 4.4 Display specific fields from on record based on multiple conditions set.

## Unit 5. Displaying Data from Multiple Tables

### General Outcome:

5.0 The students should be able to select data from multiple tables using a foreign key.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 5.1 Discuss the concept of a foreign key
- 5.2 Discuss cardinality
- 5.3 Describe entity relationships
- 5.4 Code a select statement to display records from two tables using a foreign key
- 5.5 Code a select statement to display selected fields from two tables using a foreign key
- 5.6 Code a select statement to display selected fields from two tables using a foreign key and utilizing a sort.

## Unit 6. Aggregating Data Using Group Functions

### General Outcome:

6.0 The students should be able to group sets of data together for the purpose of summary information.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

6.1 Discuss the concept of the group by clause

6.2 Code the group by clause

6.3 Utilize the avg function

6.4 Utilize the count function

6.5 Code the max function

6.6 Code the min function

6.7 Code the sum function

6.8 Discuss the having clause

6.9 Utilize the having clause

6.10 Code the intersect clause

## Unit 7. Writing Subqueries

### General Outcome:

7.0 The students should be able to code subqueries along with their SQL and PL/SQL programs.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 7.1 Discuss modularity
- 7.2 Define a subquery
- 7.3 Describe the purpose of a subquery
- 7.4 Code and utilize a subquery

## Unit 8. Multi-Column Subqueries

### General Outcome:

8.0 The students should be able to code subqueries along with their SQL and PL/SQL programs in order to extract selected columns from fields within the database.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

8.1 Identify subqueries extracting selected columns from fields within the database.

8.2 Code subqueries to extract selected columns from fields within the database.

## Unit 9. Producing Readable Output with SQL \*Plus

### General Outcome:

9.0 The students should be able to format output headings and subheadings

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

9.1 Create a heading

9.2 Create a subheading

9.3 Format data from the tables under headings

## Unit 10. Manipulating Data

### General Outcome:

10.0 The students should be able to properly maintain a database.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

10.1 Update data within the table

10.2 Add records to a table

10.3 Delete records from a table

10.4 Discuss the concepts and actions of the update, add and delete

10.5 Discuss when backups are desirable or mandated for safety and security

## Unit 11. Creating and Managing Tables

### General Outcome:

11.0 The students should be able to create and manage tables utilizing Oracle.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

11.1 Describe rules for naming tables and field names

11.2 Discuss cardinality

11.3 Discuss entity relationships

11.4 Discuss the rules for naming tables and columns

11.5 Define the various data types

11.6 Create a database table

11.7 Establish a primary key field

11.8 Establish field sizes

11.9 Enter data into a table

11.10 Update data within a table

11.11 Delete data with a table

11.12 Backup a table

11.13 Modify a table structure

11.14 Change lengths of character fields

## Unit 12. Including Constraints

### General Outcome:

12.0 The students should be able to edit input entered into databases using masks, default values, as well as establishing required fields.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 12.1 Establish required fields within a table
- 12.2 Create masks for data input
- 12.3 Set default values within fields

## Unit 13. Creating Views

### General Outcome:

13.0 The students should be able to create views.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

13.1 Define the concept of view

13.2 Display contents of a table using a view

13.3 Update a table using a view

13.4 Insert data into a new table using a view

## Unit 14. Other Database Objects

### General Outcome:

14.0 The students should be able to demonstrate distinct and synonym concepts.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

14.1 Discuss and code the distinct clause within the select

14.2 Code the select synonym

14.3 Code the create synonym

## Unit 15. Controlling User Access

### General Outcome:

15.0 The students should be able to restrict and grant user access to databases.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

15.1 Discuss user table rights and restrictions

15.2 Code grant clauses to gain user access

15.3 Discuss how to use views for security purposes

## Unit 16. SQL Workshop

### General Outcome:

16.0 The students should be able to use SQL and PL/SQL commands to construct programs designed to fulfill user requests.

### Specific Learning Outcomes:

Upon successful completion of this unit, the students should be able to:

- 16.1 Design code to create a user screen for input
- 16.2 Design code to create a user screen for viewing data
- 16.3 Design code to update table information
- 16.4 Describe candidate keys
- 16.5 Discuss secondary keys

## Unit 17: Declaring Variables

### General Outcome:

17.0 This unit teaches students how to declare and utilize variables.

### Specific Learning Outcomes:

Upon successful completion of this unit, students will be able to:

17.1 Define an alias

17.2 Declare a variable based upon calculations of numeric values within the database

17.3 Display the calculated variable

## Unit 18: Writing Executable Statements

### General Outcome:

18.0 The students should be able to create, store and execute executable PL/SQL code.

### Specific Learning Outcomes:

Upon successful completion of this unit, students will be able to:

18.1 Create executable programs

18.2 Store executable programs

18.3 Describe the execution phase

18.4 Execute SQL and PL/SQL programs

## Unit 19: Interacting with the Oracle Server

### General Outcome:

19.0 The students should be able to interact with oracle server.

### Specific Learning Outcomes:

Upon successful completion of this unit, students will be able to:

19.1 Describe Oracle networking

19.2 Describe Oracle network security

19.3 Discuss Oracle management

19.4 Describe the Oracle Server/Client relationship

## **Unit 20: Writing Control Structures**

### General Outcome:

20.0 This unit teaches students to utilize the control structures within their program code.

### Specific Learning Outcomes:

Upon successful completion of this unit, students will be able to:

20.1 Describe the If/Then structure

20.2 Code the If/Then structure

20.3 Describe the If/Then/Else structure

20.4 Code the If/Then/Else structure