# Macintosh HD:Users:deepa:Downloads:sps_long_color.jpg

# **Advanced SQL Programming**

# **General Course Information**

**Assistant Professor:** Sam Sultan

**Email:** sam.sultan@nyu.edu

**Course title/number:**  Advanced SQL Programming – INFO-CE5150

**CE Units:** 2 units

**Semester/Year**: Fall 2018

**Class meeting/location:** Saturdays at 1:30pm-5:30pm

**Office Hours** Email me to request an appointment

# **Course Description:**

Expand your skills in SQL, one of the most sought-after technical skills in the market today. Proficiency with SQL ranks among the top required skills on nearly every major information technology job search site. SQL also is used by business analysts for data analytics, one of the fastest-growing fields in business today. In this course, study data normalization rules and SQL subqueries. Learn how to create views; work with the database data dictionaries; and build SQL functions, stored procedures, and triggers to automate data manipulation tasks in both Oracle and MySQL, two of the most dominant database servers. Learn how to import and export data and how to perform database backups. The need for SQL developers is constantly on the rise, as are their salaries. According to Business Insider, strong knowledge in SQL could potentially earn you $125K per year.

# **Course Prerequisites:**

Introductory/basic knowledge of the SQL programming language

# **Course Structure/Method:**

This course will be delivered in-person, over a 3 week period. The class will encompass lectures, assignments, examples, demos, and a final exam. All class content and assignments will be made available online via http://oit2.sps.nyu.edu/~sultans/sql2. Student should check the web site on a daily basis for any announcements

# **Course Learning Outcomes:**

By the end of this course, students will be able to:

* Obtain a working knowledge of the various SQL functions
* Become proficient with SQL sub-queries
* Design databases for OLTP applications, ODS and data marts
* Create and drop databases tables and indexes
* Execute insert, update and delete data from database tables
* Become familiar with both Oracle and MySql data dictionary tables
* Create custom functions and procedures in Oracle PL/SQL procedural language
* Create custom functions and procedures in MySQL procedural language
* Create database triggers

# **Communication Policy:**

Please email instructor using your NYU assigned email (if provided). All email inquiries will be responded to within 24 hours during Monday through Friday 5pm. Email sent on Saturday or Sunday will not be responded to until Monday.

# **Course Expectation:**

Students are expected to participate in each class session by offering their understanding of the subject, sharing ideas or discussing/commenting on another students comment. In addition, students must complete and submit all assigned homework on time. Late submission of homework will either not be accepted, or will result in a lower grade. Students are also expected to take and pass a final exam.

See full detail of expectations under “Assessment Strategy” below. Further information about specific assignments can also be found in the “Course Outline” section

**Attendance:** Students are expected to attend all classes. Excused absences are granted in cases of documented serious illness, family emergency, religious observance, or civic obligation. In the case of religious observance or civic obligation, this should be reported in advance. Unexcused absences from sessions may have a negative impact on a student’s final grade. Students are responsible for assignments given during any absence. Each unexcused absence may result in a student’s grade being lowered by 3 point.

University Calendar Policy on Religious Holidays <https://www.nyu.edu/about/policies-guidelines-compliance/policies-and-guidelines/university-calendar-policy-on-religious-holidays.html>

**Class Participation:** To receive full credit for the course, you should attend all classes since much of the learning occurs during class lecture, presentation and class discussions. Please contact the instructor if you anticipate missing any part of the class. Please arrive on time so as not to disturb the flow of the lecture. Excessive lateness’s may result in lower overall grade.

**Required & Recommended Material:**

* Teach Yourself SQL in One Hour a Day (5th Edition)
* **Authors** - Ryan Stephens, Ronald Plew & Arie Jones
* **Publisher** - SAMS
* Instructor will also provide session by session content available online at class web site
* <http://oit2.scps.nyu.edu/~sultans/sql>2

**Additional Recommended Material**

* SQL Bible (Second Edition)
* **Author** - Alex Kriegel and Boris Trukhnov
* **Publisher** - Wiley

**Assessment Strategy:**

Contributing factors for determining your course grade include:

* Class Attendance and Participation - **10%**
* Homework - **20%**
* Final Exam - **70%**
* **Class Participation:** To receive full credit for the course, you should attend all classes since much of the learning occurs during class lecture, presentation and class discussions. Please contact the instructor if you anticipate missing any part of the class. Participation grades will be based on:
* Involvement in class discussions and activities
* Participation which demonstrates integration of reading, class work, relevance and application.
* Willingness to learn by accepting feedback, trying new skills and approaches, etc.
* Quality/quantity of providing effective and balanced feedback.
* **Homework:** Homework assignments must be submitted on time within 1 week of date assigned (unless otherwise instructed). Late submission will severely impact your homework grade, or may not be accepted altogether at instructor’s discretion. Late submission will incur 10 points penalty for each session we meet and not submitted. All homework pages must be stapled together (paper clips not accepted).
* **Final Exam:** There will be a final exam. The exam will be an open book, open notes/internet style exam. The exam will test the student's acquisition of topics, concepts and competencies learned in this class.

**NYU SPS Policies:**

“NYUSPS policies regarding the Family Educational Rights and Privacy Act (FERPA), Academic Integrity and Plagiarism, Students with Disabilities Statement, and Standards of Classroom Behavior among others can be found on the NYU Classes Academic Policies tab for all course sites as well as on the University and NYUSPS websites.
Every student is responsible for reading, understanding, and complying with all of these policies.”

The full list of policies can be found at the web links below:

* University: <http://www.nyu.edu/about/policies-guidelines-compliance.html>
* NYUSPS: <http://sps.nyu.edu/academics/academic-policies-and-procedures.html>

**NYU School of Professional Studies Graduate Grading Scale**

Grading for graduate programs is by letter grade: A, A-, B+, B, B-, C+, C, C-, and F. For NYUSPS’s complete graduate grading policies, including criteria for a grade of incomplete, taking a course on a pass/fail basis, and withdrawing from a course, see:
 <http://scps.nyu.edu/academics/academic-policies-and-procedures/graduate-academic-policies-and-procedures.html#Grades>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Letter** | **%** | **GPA** | **Descriptions** | **Definitions** |
| A | 95-100 | 4.0 | Exceptional | Demonstrates exceptional mastery of all learning outcomes of the course and thorough and complete understanding of all concepts. |
| A- | 90-94 | 3.7 | Excellent | Demonstrates highly competent mastery of all learning outcomes of the course and strong understanding of all concepts. |
| B+ | 87-89 | 3.3 | Very Good; exceeds course standards | Demonstrates mastery of all learning outcomes of the course and understanding of core concepts. |
| B | 83-86 | 3.0 | Good; meets course standards | Demonstrates mastery of some learning outcomes; understanding of some core concepts could be improved. |
| B- | 80-82 | 2.7 | Somewhat Satisfactory; meets some course standards and requires improvement | Demonstrates basic understanding of some learning outcomes; improved understanding of all core concepts is needed. |
| C+ | 77-79 | 2.3 | Less than Satisfactory; requires significant improvement | Demonstrates partial understanding of all learning outcomes and core concepts; requires significant improvement. |
| C | 73-76 | 2.0 | Unsatisfactory; requires substantial improvement | Demonstrates partial understanding of some learning outcomes and core concepts; requires substantial improvement. |
| C- | 70-72 | 1.7 | Unsatisfactory; requires extensive improvement | Demonstrates poor understanding of all learning outcomes and core concepts; requires extensive improvement. |
| F | Below 70 | 0.0 | Fail | Demonstrates minimal to no understanding of all key learning outcomes and core concepts; work is unworthy of course credit towards the degree. |

### **Course Outline:**

**Session 1, Week 1, SQL Built-in Functions**

* SQL built-in Functions
* Numeric functions - CEIL, FLOOR, ROUND, TRUNCATE, etc.
* String functions - CONCAT, LENGTH, SUBSTR, REPLACE, etc.
* The CASE expression, 2 flavors
* Date functions - MySQL and Oracle
* Current date, date manipulation, date formatting

 **Reading:** Chapter 7 (Skip aggregate functions), chapter 12

**Session 2, Week 1, Aggregating and Grouping**

* Aggregating and Grouping
* Aggregate functions - SUM, COUNT, AVG, MIN, MAX
* The GROUP BY clause
* The HAVING clause
* Finding Duplicate Records
* GROUP BY with ROLLUP feature
* The ORDER BY clause Pivoting rows into columns

 **Reading:** Chapter 7 (Aggregate functions), Chapter 4

**Session 3, Week 2, Select Sub-Queries**

* Using SELECT Sub-Queries
* Subqueries as filters
* Subqueries as inline views
* Subqueries as additional derived columns
* Correlated Subqueries
* Where [NOT} EXISTS in Subquery
* Finding the last record from a set
* Pivoting rows into columns

 **Reading:** Chapter 6

**Session 4, Week 2, Database Design**

* The Logical and Physical Model
* Understanding data normalization
* First normal form
* Second normal form
* Third normal form
* Pros & cons of data normalization
* Denormalizing data
* Entity relationships
* One-to-one relationship
* One-to-many relationship
* Many-to-many relationship
* Designing Self-join relationship
* Designing for an ODS (Reporting Database)
* Designing for a Data Warehouse

 **Reading:** Chapter 8

**Session 5, Week 3, Creating Database Objects**

* What is a primary key?
* What is a foreign key?
* What is an index?
* Creating tables
* SQL data types
* Adding a primary key
* Adding constraints
* Creating Indexes
* Altering table definition
* Dropping tables
* MySql Auto Increment
* Oracle Sequences

 **Reading:** Chapter 9, 10, 15

**Session 6, Week 3, Manipulating Data in Tables**

* Manipulating data in database tables
* Adding data with the INSERT statement
* INSERT with a SELECT statement
* Changing data with the UPDATE statement
* UPDATE with a SELECT statement
* Removing data with the DELETE statement
* DELETE with a SELECT statement
* The TRUNCATE statement
* The REPLACE statement (MySql)
* The MERGE statement (Oracle)

 **Reading:** Chapter 11

**Session 7, Week 4, Advanced Topics**

* Creating and using views
* Using the Data Dictionary – MySql
* Using the Data Dictionary – Oracle
* The show statement (MySql)
* Loading data from a file
* Unloading data into a file
* Importing a database or table(s)
* Exporting a database of table(s)

 **Reading:** Chapter 13, 18

**Session 8, Week 4, MYSQL stored Procedures**

* What are Stored Procedures
* Stored Procedures, Functions & Triggers
* Creating Stored Procedures
* Calling Stored Procedures
* Creating Variables
* Global Variables
* Creating code blocks
* Conditional Testing- IF, CASE
* Loops and iterations - LOOP, WHILE, REPEAT
* Procedures with non-SELECT statements
* Using Unbounded SELECT statements
* Using SELECT INTO statements
* Using SELECT with cursor statements
* Creating Stored Functions

 **Reading:** online

**Session 9, Week 5, Introduction to ORACLE PL/SQL**

* The PL/SQL code block
* Creating Named and Anonymous Blocks
* Creating Variables
* Using %TYPE and %ROWTYPE
* Assigning values to variables using SELECT INTO
* Conditional Statements - IF, CASE
* Loops and Iterations - LOOP, FOR, WHILE
* Using Nested Loops
* Using DBMS\_OUTPUT and other DBMS\_ packages

 **Reading:** Chapter 26

**Session 10, Week 5, PL/SQL Functions and Procedures**

* Calling Functions and Procedures
* Nesting Functions and Procedures
* Saving and Retrieving Functions and Procedures
* Obtaining Compilation Error messages
* Creating and Working with Cursors
* Implicit and explicit cursors
* Declare, Open, Fetch and Close a cursor
* Looping through cursors
* The cursor FOR loop
* Using cursors with Parameters
* Nested Loops with nested Cursors

 **Reading:** Chapter 26

**Session 11, Week 6, PL/SQL Exception Processing**

* PL/SQL Exception Processing and Handling
* List of named exceptions
* Creating and raising your own exceptions
* Creating and Using Triggers
* Statement-level Triggers, Row-level Triggers
* INSTEAD OF Triggers for Views
* Collection Types - VARRAY, and Nested TABLE
* Loading data into a Collection
* Creating Packages

 **Reading:** Chapter 26

**Week 6, - Final Exam**