## Instruction Schedule – Based on Part - Time Cohort

• 60 minutes: welcome and content read

• 60 minutes: lecture and demo lessons

• 30 minutes: work on challenges

• 30 minutes: walk through challenge solutions and wrap up

## Instruction Schedule – Based on Full - Time Cohort

### Morning session:

• 60 minutes: welcome and content read

• 60 minutes: lecture and demo lessons

• 30 minutes: work on challenges

• 30 minutes: walk through challenge solutions and wrap up

Break – 30 mins

### Afternoon session:

• 60 minutes: recap and content read

• 60 minutes: lecture and demo lessons

• 30 minutes: work on challenges

• 30 minutes: walk through challenge solutions and wrap up

## SQL Instructor Guide Introduction

Welcome to SQL! This workshop will introduce students to SQL and the fundamentals of databases. Your role as the instructor is critical to the student learning experience. You will be meeting with students in a synchronous environment three times a week and should be prepared with the teaching goals, strategies, and structure for each class. This is where the instructor guide comes in! Each synchronous class covers one module and you will be provided an instructor guide for every module. The modules are further subdivided into lessons. The guide is meant to offer you the goals and key considerations for each module as whole, but also each lesson. We hope this will help and make you feel supported in your very important role.

For SQL modules you will want to set students up with [DBeaver](https://dbeaver.io/) and then work through connecting to existing module data sets.

If students are not comfortable downloading they can use this [online sql editor](https://www.sqltutorial.org/seeit/) as well. Keep in mind that the online option does not connect students to existing databases using their raw data. It is set up with preexisting data and will allow students to apply SQL concepts.

# Module 1 – Instructor guide

### Databases

### Module Learning Outcomes

In this module students will,

1. Define relational databases.
2. Identify the difference between a database and a spreadsheet.
3. Define structured query language.
4. Recognize what Structured Query Language (SQL) can do.

### Module Overview Description

This module will introduce students to relational databases and structured query language (SQL). The module wraps up with a Challenge activity to review what they have learned in the module by answering several foundational questions.

### Considerations to Keep in Mind

* Some students may come with no experience in SQL whereas others might already be proficient.
* There is a workbook for the entire SQL workshop. Encourage students to use the workbook to take notes.
* Each class will start with a lesson video that will introduce key concepts and examples for the designated modules and how data analysts use SQL.
* The module will end with a challenge activity.

### Lesson 1: What is a Database

* Build students knowledge and guide them through the following ares:
  + Define database.
  + Describe some database features.
  + Identify what unique values are called in a database.

### Lesson 2: Relational Databases

* Walk students through examples of the following:
  + Define relational database.
  + Define a query.
  + Describe relational database advantages.
  + Identify the advantages that databases offer analysts.

### Lesson 3: Databases versus Excel

* Build students’ knowledge and guide them through the following areas:

### Describe some key differences between Excel spreadsheets and databases.

### List the overall findings after applying complex functions to Excel data.

### List the overall findings after pulling records from database tables using primary keys.

### Lesson 4: Introduction to SQL

### Build the student’s knowledge and guide them through the following areas:

* + Define SQL and what SQL stands for.
  + Describe what SQL can do.
  + Define the following SQL key terms.
  + Define the following SQL key terms.
    - Schema
    - Objects
    - Syntax
    - Run
    - Identifiers
    - Parameters
    - Statements
    - Conditions
    - Queries
    - Constraints

# Module 2 – Instructor guide

### SQL Fundamentals

### Module Learning Outcomes

In this module students will,

1. Recognize how SQL is structured.

2. Demonstrate basic SQL Fundamentals.

3. Identify clause order of operations.

4. Create simple SQL syntax using SELECT and FROM.

### Module Overview Description

### This module will introduce students to SQL order of operations, building SQL queries, and syntax rules. The module wraps up with a Challenge activity to review what they have learned in the module and do a sample query.

### Lesson 1: SQL Order of Operations

* Walk students through the following:
  + Describe some database features.
  + Identify what unique values are called in a database.
  + SQL order of operations

### Lesson 2: Building SQL Queries

* Walk students through examples of the following:
  + Define syntax.
  + Identify the most commonly used commands to interact with relational databases.
  + Describe how to write a SELECT statement.
  + Describe how to write a FROM clause.
  + Identify what the asterisk symbol is used for.

### Lesson 3: Syntax Rules

* Build students’ knowledge and guide them through the following areas:

### Identify the syntax rules needed to create an SQL statement.

### Describe a few common syntax errors.

### Lesson 4: Multiple Columns using SELECT

* Build the student’s knowledge and walk them through the following areas:
  + Describe how to tell a database to pull data from multiple columns in a database
  + Write a sample query that demonstrates this.

# Module 3 – Instructor guide

### SQL Fundamentals: Creating Complete Queries Using SFWGHO

### Module Learning Outcomes

In this module students will,

### 1. Combine SFWGHO to make basic queries.

### 2. Identify clause operators.

### 3. Identify the different types of clauses.

### Module Overview Description

### This module will introduce students to the clauses WHERE, GROUP BY, HAVING, AND ORDER BY using the SELECT statement. The module wraps up with an interactive Challenge activity.

### Lesson 1: SQL WHERE Clause

* Walk students through the following:

### Define operators.

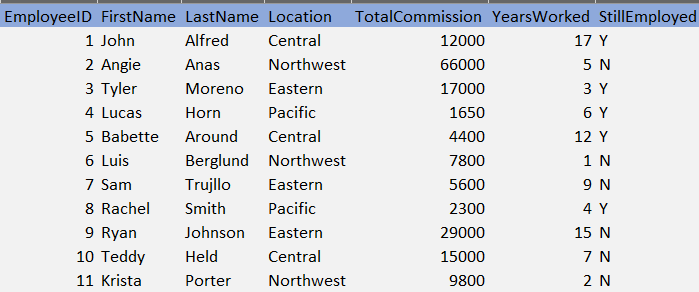
### Define the WHERE clause and how it is used.

### Describe some WHERE clause uses.

### Illustrate what a WHERE clause query will look like.

### Lesson 2: SQL GROUP BY clause

* Walk students through examples of the following:
  + Describe the purpose of the GROUP BY clause.
  + Describe one thing to keep in mind about using the GROUP BY clause.
* Using the data table below to select employee ids from the database, whose total commission is less than 500 and group them by first name.



* Walk students through the following queries and examples:
  + You will be pulling records from the employee table.
  + The columns you want are FirstName and EmployeeID.
  + You want to group your data output by the employee's first name.
  + You want to set a filter for total commission of less than 5000.
  + What columns will be in your SELECT statement?
  + What column will you GROUP BY?
  + What do you need to include for pulling multiple columns?
  + What clause acts like a filter and sets conditions on columns?
  + How to create additional queries based on the table data.

### Lesson 3: SQL HAVING Clause

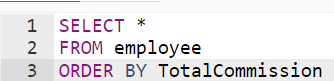
* Build the student’s knowledge and walk them through the following areas:
  + Describe the purpose of the HAVING clause.
  + Define aggregate functions.
  + Identify some examples of aggregate functions.
  + Illustrate what a query looks like that runs a HAVING clause and what purpose it serves.

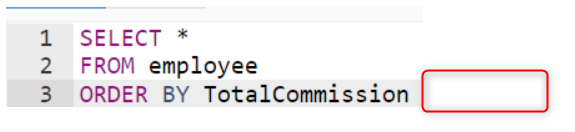
### Lesson 4: WHERE versus HAVING

* Walk students through how to complete the table below that compares WHERE vs HAVING clauses:

| WHERE | HAVING |
| --- | --- |
| [insert information on the WHERE clause] | [insert information on the HAVING clause] |

### Lesson 5: SQL ORDER BY clause

* Build the student’s knowledge and walk them through the following areas:
  + Describe the purpose of the ORDER BY clause.
  + Explain the difference between ascending and descending order.
  + What does the query below tell SQL to do?
  + 
* How will the data be sorted based on the way the query is written?
* Let's say that you do not want to sort the data results in ascending order.
* How would you complete the query below?



This module has the [*SQL Order Of Operations*](https://content.bridgepointeducation.com/curriculum/file/f1c80322-f38d-4e58-baba-400c04e2b368/1/fs_da_sql_order_of_operations.zip/story.html) interactive available for students to interact with and test their newly acquired skills. This interactive will be used for the Challenge Activity at the end of this module. Students will need to complete a query using the SQL clause order of operations.

# Module 4 – Instructor guide

### SQL Operators

### Module Learning Outcomes

In this module students will,

### Identify when to use logical operators.

### Define terminology of operators.

### Identify the most commonly used SQL operators.

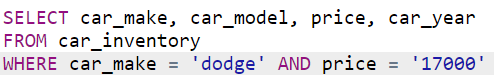
### Create SQL queries and combine operators.

### Module Overview Description

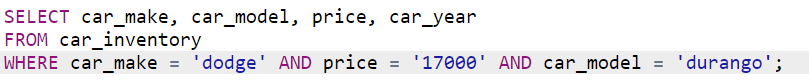
### This module will introduce you to SQL operators and how to combine them when creating queries. The module wraps up with a Challenge activity to create queries, using sample scenarios. Students will be introduced to using [myCompiler](https://www.mycompiler.io/) to walk through the order of operations and create their queries.

### Lesson 1: AND Operator

* Walk students through the following:
  + Describe the purpose of logical operators used with the WHERE clause.
  + Describe what the AND operator does.
  + Illustrate an example of a query using the AND operator.
  + Identify the number of conditions in the query below.

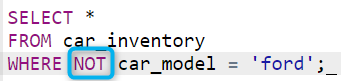


* + Identify the number of conditions in the query below and what data will be generated.

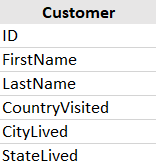


### Lesson 2: NOT Operator

* Walk students through examples of the following:
  + How are the NOT operators used in SQL?
  + Describe what NOT operators do.
  + what is the NOT operator excluding from the data output in the query below?



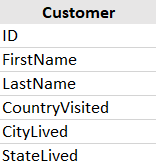
* + NOT operators are used with which clause?
  + Using myCompiler walk through the order of operations and create your queries.
    - Here is what you need to know:
    - Table name = Customer
    - Column names:
    - ID
    - FirstName
    - LastName
    - CountryVisited
    - CityLived
    - StateLived
  + Below is a sample of the column names in the table:



* + Walk students through how to complete the following steps:
    - Run a query that will pull all customers records in the table but exclude the customers that visited Albania.
    - How many conditions will this query have?
    - What column name will you be using to apply the NOT operator?
    - Run a query that will pull all customer records in the table, but exclude the customers that live in the city of Chicago.
    - How many conditions will this query have?
    - What column will the condition be applied to?

### Lesson 3: OR Operator

* Build student knowledge and guide them through the following areas:
  + Describe the purpose of the OR operator.
  + Describe when the OR operator is used.
  + Illustrate some syntax using the OR operator.
* Walk students through how to create queries based on the sample table data:



### Using the customer table above, you want to pull customer records that include the following information: FirstName, LastName, CountryVisited

### Customers who visited Mexico or France

* + - What column name will you be using to apply the OR operator?
    - What will the SQL query look like?
    - What customer records will be included in the data output?
    - Add customers who have also visited Switzerland.

### Lesson 4: Combining Operators

* Build the student’s knowledge and walk them through the following areas:
  + Describe when you would want to combine operators.
  + Illustrate some examples of multiple operator queries and describe what they do.

# Module 5 – Instructor guide

### More SQL Statements

### Module Learning Outcomes

In this module students will,

### 1. Identify SQL commands.

### 2. Identify which command is used for which scenario.

### 3. Create queries using SQL statements.

### Module Overview Description

### This module will introduce students to more SQL statements and how to create queries using each statement. The module wraps up with a Challenge activity using a demo database to answer questions and create queries from sample scenarios.

### Lesson 1: INSERT INTO statement

* Walk students through the following:
  + Describe what statements do.
  + Describe what the INSERT INTO statement does.
  + Identify and describe two ways that the INSERT INTO statement is used.
  + Describe how you add values into specific columns in a table using the INSERT INTO statement.

### Lesson 2: UPDATE Statement

* Walk students through the following:
  + Describe what the UPDATE statement does.
  + What do you need to be careful of when using the UPDATE statement?
  + Illustrate an UPDATE statement sample query and describe what it does.
  + Describe what a SET clause is.

### Lesson 3: DELETE Statement

* Walk students through the following:
  + Describe when the DELETE statement used.
  + What do you need to be careful of when using the DELETE statement?
  + Illustrate a few sample DELETE statements with conditions and describe what each does.

### Lesson 4: SELECT DISTINCT Statement

* Walk students through the following:
  + Describe when the SELECT DISTINCT is used.
  + Identify what the SELECT statement does.
  + Explain how the SELECT DISTINCT statement works.
  + Illustrate a sample SELECT DISTINCT statement and describe what it does.
  + Describe a sample use case for using the SELECT DISTINCT statement with an aggregate function.

# Module 6 – Instructor guide

### SQL Joins

### Module Learning Outcomes

In this module students will,

### 1.Identify SQL JOINS

### 2. Define terminology and key words of each join.

### 3. Identify when to use SQL JOINS.

### 4. Demonstrate how to create queries using JOINs.

### Module Overview Description

### This module will introduce students to the types of joins that SQL offers and how to create queries using each. The module wraps up with a Challenge activity using sample query builders to answer questions and create queries off of sample scenarios.

### Lesson 1: Types of Joins

* Walk students through the following:
  + Define the SQL JOIN clause and how it is used.
  + Define Inner Join:
  + Define Left Outer Join:
  + Define Right Outer Join:
  + Define Full Outer Join:
  + Describe what a join does when it is run.

### Lesson 2: Inner Join

* Walk students through the following:
  + Explain the purpose of the inner join.
  + Illustrate an example of an SQL statement that includes an inner join and describe what it will produce.

### Lesson 3: Left and Right Joins

* Walk students through the following:
  + Explain the purpose of the left join, also known as the left outer join.
  + Illustrate an example of an SQL statement that includes a left join and describe what it will produce.
  + Explain the purpose of the right join, also known as the right outer join.
  + Illustrate an example of an SQL statement that includes a right join and describe what it will produce.

### Lesson 4: Full Outer Join

* Walk students through the following:
  + Explain the purpose of the full outer join.
  + Illustrate an example of an SQL statement that includes a full outer join and describe what it will produce.

# Module 7 – Instructor guide

### SQL Aggregate Functions

### Module Learning Outcomes

In this module students will,

### 1. Identify aggregate functions.

### 2. Demonstrate SQL fundamentals using aggregate functions.

### 3. Create SQL syntax using aggregate functions.

### Module Overview Description

### This module will introduce students to the types of aggregate functions that SQL offers and how to create queries using each. The module wraps up with a Challenge activity using sample demo databases to answer questions and create queries from sample scenarios.

### Lesson 1: Aggregate Functions

* Walk students through the following:
  + Describe what an aggregate function does.
  + Identify when to use an aggregate function.
  + Define the following aggregate functions:
    - SUM
    - COUNT
    - MIN
    - MAX
    - AVG
  + Identify the clauses that aggregate functions are often with.

### Lesson 2: SUM and COUNT

* Walk students through the following:
  + Illustrate an example of an SQL query that uses the SUM function and describe the output.
  + Illustrate an example of an SQL query that uses the COUNT function and describe the output.

### Lesson 3: MIN and MAX

* Walk students through the following:
  + Describe when the MIN function is used.
  + Describe when the MAX function is used.
  + Illustrate an example of an SQL query that uses the MIN function and describe the output.
  + Illustrate an example of an SQL query that uses the MIN function and describe the output.

### Lesson 4: AVG and Round

* Walk students through the following:
  + Describe when the AVG function is used.
  + Illustrate an example of an SQL query that uses the AVG function and describe the output.
  + Describe when the ROUND function is used.
  + Illustrate an example of an SQL query that uses the ROUND function and describe the output.

This module has the [*SQL Aggregate Functions.*](https://content.bridgepointeducation.com/curriculum/file/02f1bd4a-904d-40db-9ee6-f73d6a44c944/1/SQL%20Aggregate%20Functions.zip/story.html) interactive available for students to interact with and test their newly acquired skills. This interactive is used as a learning activity to check knowledge on SQL aggregate functions.

# Module 8 – Instructor guide

### SQL Single Line Comments, Alias Using AS, NULL Values

### Module Learning Outcomes

In this module students will,

### 1. Understand the importance of line comments.

### 2. Demonstrate how to include line comments in the query box.

### 3. Create SQL syntax and data output using aliases.

### 4. Combine line comments, aliases, and NULL values in SQL.

### Module Overview Description

### This module will introduce students how to identify your queries, null values, and rename data output. The module wraps up with a Challenge activity using sample demo databases to create queries using line comments, aliases, and identifying NULL values.

### Lesson 1: Single Line Comments

* Walk students through the following:
  + Describe what line comments are and why you should use them.
  + Illustrate an example of a single line comments being used as an explanation.
  + Describe why single line comment text must be contained in the dashes.
  + Describe some other uses for single line comments.
  + Illustrate the syntax for writing a multi-line comment.

### Lesson 2: Alias Using AS

* Walk students through the following:
  + Describe the purpose of using SQL aliases.
  + Illustrate the syntax for writing a table alias.
  + Illustrate the syntax for writing column alias.
  + Identify the clauses that column aliases can be used with.
  + Identify the clauses that column aliases cannot be used with. Why?
  + Illustrate a query that uses an alias and describe the data output.

### Lesson 3: Null Values

* Walk students through the following:
  + Define a NULL value.
  + Describe why it is important to communicate NULL values in the data to stakeholders.
  + Illustrate how to test for NULL values.
  + Discuss how aggregate functions deal with NULL values.

### Lesson 4: NULL Functions

* Walk students through the following:
  + Describe some reasons that a dataset may have NULL values.
  + Define the ISNULL function.
  + Illustrate a query that uses the ISNULL function and describe the data output.
  + Define the IFNULL function.
  + Illustrate a query that uses the IFNULL function and describe the data output.

# Module 9 – Instructor guide

### SQL Stored Procedures and Limits

### Module Learning Outcomes

In this module students will,

### 1. Identify the importance of creating and storing procedures in SQL.

### 2. Demonstrate how to create and execute stored procedures in SQL.

### 3. Create SQL syntax combining multiple operators.

### 4. Recognize when and how to limit records.

### Module Overview Description

### This module will introduce students how to create and store procedures and how to limit the data output based on specifications. The module wraps up with a Challenge activity using a sample demo database to create queries, store procedures, and limit the data pulled based on parameters.

### Lesson 1: Stored Procedures

* Walk students through the following:
  + Identify the checklist that you need to run through to prepare for complex query building.
  + Identify two steps needed to create a stored procedure.
  + Illustrate an example of the parameters for a stored procedure query and explain the output.
  + Illustrate how to run the stored procedure.

### Lesson 2: Limit

* Walk students through the following:
  + Define the LIMIT clause and identify its purpose.
  + Illustrate an example of a query with a LIMIT clause and explain the data output.
  + Illustrate an example of a query with a LIMIT clause that also includes an ORDER BY clause and explain the data output.

### Lesson 3: BETWEEN

* Walk students through the following:
  + Define the BETWEEN operator and identify its purpose.
  + Identify the statements that the BETWEEN operator can be used with.
  + Illustrate an example of a query with a BETWEEN operator and explain the data output.
  + Illustrate an example of a query with a BETWEEN operator and an ORDER BY clause and explain the data output.
  + Illustrate an example of a query with a BETWEEN operator and an OR operator and explain the data output.

### Lesson 4: OFFSET

* Walk students through the following:
  + Define the OFFSET clause and its purpose.
  + Illustrate an example of a query that uses the LIMIT and OFFSET clauses and explain the data output.
  + Identify things to consider when using the OFFSET clause.

# Module 10 – Instructor guide

### SQL DATA SEARCH AND DATA CLEANING

### Module Learning Outcomes

In this module students will,

### 1. Identify when to clean data.

### 2. Demonstrate how to bring data together.

### 3. Combine operators and functions to search data.

### 4. Create queries to clean data records.

### Module Overview Description

### This module will introduce students to how to search and clean your data using more operators and functions. The module wraps up with a Challenge activity using a sample demo database to create queries, clean data, and bring data together.

### Lesson 1: Wildcards

* Walk students through the following:
  + Define a wildcard in SQL and its purpose.
  + Identify the operators that wildcard characters are used with.
  + Complete the information for the following wildcard characters:
    - Wildcard Symbol: Percent sign: %
      * Description:
      * Example:
    - Wildcard Symbol: Underscore:\_
      * Description:
      * Example:
  + Illustrate an example of a query that uses a wildcard character and explain the data output.
  + Explain what placing percent symbols at the front and the end of the values does for your query.
  + Illustrate an example of a query that uses an underscore wildcard character and explain the data output.

### Lesson 2: IN

* Walk students through the following:
  + Define the IN operator and its purpose.
  + Identify the requirements of the IN operator syntax.
  + Illustrate an example of a query that uses an IN operator and explain the data output.
  + Define the NOT IN operator and its purpose.
  + Illustrate an example of a query that uses the NOT IN operator and explain the data output.

### Lesson 3: CONCAT

* Walk students through the following:
  + Define the CONCAT function and its purpose.
  + Illustrate an example of a query that uses the CONCAT function and explain the data output.
  + Describe the advantages of using the CONCAT operator.

### Lesson 4: UPPER and LOWER

* Walk students through the following:
  + Define the UPPER function and its purpose.
  + Identify the statement that the UPPER function works with.
  + Illustrate an example of a query that uses the UPPER function and explain the data output.
  + Define the LOWER function and its purpose.
  + Illustrate an example of a query that uses the LOWER function and explain the data output.

### Lesson 5: LTRIM and RTRIM

* Walk students through the following:
  + Define the LTRIM and RTRIM functions in SQL and their purpose.
  + Illustrate an example of a query that uses the LTRIM function and explain the data output.
  + Illustrate an example of a query that uses the RTRIM function and explain the data output.

# Module 11 – Instructor guide

### SQL Creating Tables and Column Manipulation

### Module Learning Outcomes

In this module students will,

### Define different data types and character widths.

### Demonstrate how to create a new table.

### Demonstrate how to alter an existing table.

### Combine statements and conditions to create constraints in a table.

### Identify different date data types and how to work with them in a table.

### Module Overview Description

### This module will introduce students to how to create tables, insert columns, alter table data, data constraints, and work with dates. The module wraps up with a Challenge activity using a sample demo database to create queries and bring data together.

### Lesson 1: Creating a Table

* Walk students through the following:
  + Define the following string data types:
    - Char (n)
    - Varchar (n)
    - Varchar (max)
  + Define the following string numeric types:
    - Integer
    - Float
    - Date
    - Time
    - Timestamp
    - Boolean
  + Describe how to create a table in SQL.
  + Illustrate a query that can be used to create a table in SQL.

### Lesson 2: Inserting Data into Tables

* Walk students through the following:
  + Describe how to insert data into a table.
  + Illustrate a query that can be used to insert data into a table and describe the data output.
  + Describe the process and keyword to auto-increment a table in SQL.

### Lesson 3: Drop/Alter Tables

* Walk students through the following:
  + Describe how to drop a table in SQL.
  + Illustrate an example of a query that uses the DROP TABLE statement.
  + Describe what the TRUNCATE TABLE statement does and when to use it.
  + Illustrate an example of a query that uses the TRUNCATE TABLE statement.
  + Describe what the ALTER TABLE statement does and when to use it.
  + Illustrate an example of a query that uses the ALTER TABLE statement.

### Lesson 4: Constraints

* Walk students through the following:
  + Define SQL constraints and their purpose.
  + Identify the two types of constraints and how they are applied.
  + Identify when constraints can be specified.
  + Identify how to use constraints with NULL values.
  + Illustrate an example of a query that sets constraints when creating a table.
  + Illustrate an example of a query that sets constraints when altering a table.

### Lesson 5: Working with Dates

* Walk students through the following:
  + Describe when date data types are useful.
  + Identify the function that is used for date data queries.
  + Illustrate a query example that is designed to pull data on a specific date.

This module has the  [*SQL Summary Self-Check*](https://content.bridgepointeducation.com/curriculum/file/02f1bd4a-904d-40db-9ee6-f73d6a44c944/1/SQL%20Aggregate%20Functions.zip/story.html) interactive available for students to interact with and test their newly acquired skills. This interactive is a self-check that will test your knowledge on what they have learned in the SQL modules. It will cover terminology, query building, and sample scenarios.