

site|folio

Production DBA Candidate

SQL Question Preparation

The assessment includes four custom SQL code questions. In order to prepare you for what to expect, the following notes on the questions and testing platform are provided. We advise printing out a hard copy of these notes for reference prior to beginning the test. Note: The entire assessment is to be completed in “full screen” mode. You will not have the benefit of accessing any other applications / windows from the computer you are using to take the test.

1. The SQL questions are not specific to database administration skills. They are focused on general query skills, principally SELECT statements.
2. The response time is limited to 10 minutes for each individual question. Please use the fully allotted time on each question as necessary. We are principally interested in how you approach each of the problems, as opposed to whether you are able to generate the correct output for each.
3. Prior to beginning the assessment, an “estimated completion time” is provided. For some reason, this estimate excludes these four custom SQL questions. Plan to spend an additional 30 or 40 minutes above and beyond the assessment-provided estimate to account for the SQL questions.
4. For each SQL question, code is entered and tested (executed) in a browser-based console UI. There are a few significant limitations / idiosyncrasies with the current platform to be aware of:
 - * The console application is embedded in the same page as the question. The question appears on the left side of the screen and the console on the right. The table definitions are included at the bottom of each question. Be sure to use the vertical scroll bar, between the question and the console window, to view the full question.
 - * The console window includes two panes. SQL code is entered into the top pane and the execution output is returned in the bottom pane. There is a limited amount of viewable area, particularly in the results pane, making it challenging to parse what is displayed (e.g. displays a limited number of narrow rows that are prone to line wrapping).
 - * Code is executed by clicking the ‘Run Test’ button, found between the input and output panes. Run as many tests as you like, including any supplemental queries you need along the way (e.g. to interrogate the table structures and included data, test supported SQL syntax, etc).
 - * The console UI only displays the results of a single query per execution. Specifically, only the results of the last SQL statement entered in the top pane are displayed. For example, if you have three separate SELECT statements entered, ‘Run Test’ will only display the results of the final SQL statement in the results pane. *** Important Note: *** SQL statements that are commented out still qualify. If the final SQL statement entered is commented out, no results will be returned in the output pane, even if preceded by a valid SQL statement that is not commented out.

* In addition to the returning the output of the SQL statement executed, the results pane includes supplemental information on every execution. Near the top of output you will see an “Initial structure:” line. Just below that will be a “Show console output” link. If you click this link, it will expand to show the scripts (table definitions and the data inserted into each) applied to create the database you are interacting with. If you choose to expand this link, be aware that subsequent executions will also necessarily be expanded (once it is expanded it can’t be collapsed, even on subsequent executions). Below that is another line that reads “Test Initialization:”, again followed by a “Show console output” link. This section is intended to render any setup scripts related to the test execution itself. None of the SQL questions include additional scripts, so there won’t be anything meaningful presented, should you choose to expand this link (again, once the link is expanded it is sticky and will show as expanded on all subsequent test runs). After these two sections, the output pane will display the results of the (last) SQL statement entered in the query pane (i.e. the output of your code). Finally, at the bottom of the output pane you will see the “Expected:” results. This is a visual representation of what output is expected (i.e. the expected “answer” to the question). Again, we are less concerned about whether you can generate the provided output as we are about the approach you used to address the question.

* Note: The 4th (final) SQL question does not support returning any query results in the output pane, though you can see the table structures and inserted data by expanding (clicking) the “Show console output” link. As such, there is no definitive way to test your work or confirm your code results on this particular question.

* The SQL dialect used for the assessment is not T-SQL. The testing platform uses SQLite. Much of the T-SQL syntax you may be familiar with is not supported. Some helpful tips to mitigate the differences in the SQL dialects:

1. All dates in the sample questions are persisted, and returned as, “YYYY-MM-DD” (no time component).
2. "SELECT date('now')" is a SQLite function that will return the current date formatted as 'YYYY-MM-DD'. Note: The “GETDATE()” function is not supported.
3. The "julianday" function returns the number of days since noon in Greenwich on November 24, 4714 B.C. (Note: The “DATEDIFF()” function is not supported). Examples:
 - * Number of days until Jan. 1st, 2020: "SELECT julianday('2020-01-01')"
 - * Number of days until the current date: "SELECT julianday('now')"
4. The “ISNULL()” function is not supported. Use “COALESCE()” instead.
5. While there should be no need to learn the inner working of SQLite, the complete SQLite documentation may be found at: <https://sqlite.org/index.html>

Post-assessment SQL video response:

You will have the opportunity to provide a video response that includes any additional information in respect to your SQL question responses.