

Test Results and Interview Guide

Candidate: **Elizabeth Wantsajob**
Assessment: SQL - Basic (Short)
Completed: June 28, 2026
Prepared for: Sara Maple
Example Company

What's Included

- Overall Score
- Competency Summary Table
- Comparison Matrix
- Detailed Competency Results with Interview Guide

Important Note: The SQL - Basic (Short) assessment measures key factors related to high performance and tenure in this job. Attribute types measured vary by test, but can include cognitive ability, skills, knowledge, personality characteristics, emotional intelligence, and past behavioral history. This report includes a one page summary, followed by detailed results with an embedded interview guide. Note that these results should always be used as a part of a balanced candidate selection process that includes independent evaluation steps, such as interviews and reference checks.

Overall

Candidate	Score	Interpretation
Elizabeth Wantsajob beth.wantsajob@gmail.com SQL - Basic (Short) June 28, 2026	68	

The candidate demonstrates a solid foundational to intermediate understanding of SQL, with competence across most core areas including data retrieval with filtering and grouping, data manipulation, constraint definition, and aggregate function usage. Some gaps may exist in more nuanced topics such as correlated subqueries, advanced NULL handling, or query optimization through indexes, but overall the candidate is well-positioned to perform entry-level to mid-level SQL tasks in a business environment.

Key

- Candidate Score
- Higher Risk
- Lower Risk

Competency Summary

Competency	Score	Interpretation
<i>Skills/Knowledge (relates to immediate readiness)</i>		
Aggregate Functions and GROUP BY	75	
JOIN Operations (Coding Tasks)	62	
SELECT Statements and Filtering (Coding Tasks)	62	
INSERT, UPDATE, and DELETE Statements	63	
JOIN Operations	79	
SELECT Statements and Filtering	66	

Comparison

Percentile scores indicate how the candidate compares to other test-takers within various groups. The candidate scored equal to or better than the fraction of test-takers indicated by the percentile.

Test-Taker Group	Percentile	0	10	20	30	40	50	60	70	80	90	100	
Global	68th												
North America	56th												
United States	56th												
Example Company	62nd												

Artificial Intelligence (AI) Generated Scores

This table includes one or more scores derived from a large language model AI query. AI-derived scores are non-deterministic. That is, they are not precisely repeatable. Therefore, these scores should always be treated as supplementary information and should never be used exclusively or compared to hard cutoff values.

Estimated Value	Score	Confidence	Interpretation
Knowledge, Skills, and Abilities Summary	-	-	<p>Summary Points (AI):</p> <ul style="list-style-type: none"> (Generic Text for Sample Report) Strong performer in Drag and Drop Files tasks, indicating comfort with file management and basic computer interactions. Demonstrates solid numerical accuracy in Recognizing and Confirming Numbers, a valuable asset in detail-oriented roles. Moderate overall performance in Analytical Thinking and Attention to Detail, with adequate grammar skills but room for improvement. Struggles with Reading and Analyzing Problems, which may limit effectiveness in roles requiring critical reading and complex problem-solving. Lowest performance in Navigating Between Screens, suggesting difficulty with multi-screen software workflows that could impact productivity in computer-intensive roles. <p>Narrative (AI): Elizabeth Wantsajob demonstrates a mixed profile of knowledge, skills, and abilities across the assessed competencies.</p> <p>Elizabeth shows a strong aptitude in Drag and Drop Files, performing well on this technical task and suggesting she is comfortable with this type of computer interaction. This is a notable strength that would translate well into roles requiring file management and basic computer navigation tasks.</p> <p>In the area of Analytical Thinking and Attention to Detail, Elizabeth performs at a moderate level. She demonstrates solid ability in Recognizing and Confirming Numbers, which suggests she is careful and accurate when working with numerical data — a valuable skill in detail-oriented work environments. Her Grammar performance is adequate but leaves room for improvement, indicating she may occasionally make written communication errors. Her weakest area within this competency is Reading and Analyzing Problems, where she struggled to consistently interpret and work through written problem scenarios. This may impact her effectiveness in roles that require critical reading, written comprehension, or complex problem-solving.</p> <p>Elizabeth's most significant area for development is Navigating Between Screens, where she scored considerably lower than the other competencies. This suggests she may have difficulty efficiently moving through software interfaces or multi-screen workflows, which could slow productivity in roles that rely heavily on navigating computer applications or data entry systems.</p> <p>Overall, Elizabeth brings some useful technical strengths, particularly in file management and numerical accuracy, but would benefit from targeted development in software navigation and analytical problem-solving to be fully effective in roles that demand these skills.</p> <p>Computed on: April 2, 2026, 11:09:49PM EDT</p>

Detail

Candidate: Elizabeth Wantsajob, beth.wantsajob@gmail.com
 Assessment: SQL - Basic (Short)
 Authorized: June 28, 2026, by Sara Maple, Example Company, qamailsaram.mike@hravatar.com
 Started: June 28, 2026, 9:15:16AM EDT
 Completed: June 28, 2026, 9:15:16AM EDT
 Overall Score: 68

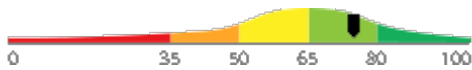
Knowledge and Skills Detail

This section contains a list of job-related knowledge areas and skills that have been evaluated. Low scores in these areas often indicate that additional learning may be required before top performance can be achieved.

Detail
Interview Guide

Aggregate Functions and GROUP BY

Score: 75



Description:

The use of aggregate functions such as COUNT, SUM, AVG, MIN, and MAX to summarize data, combined with GROUP BY to group rows by one or more columns and HAVING to filter grouped results. These are core tools for reporting and data analysis tasks.

Interpretation:

Candidate should achieve above average job performance in this area with little or no training.

The candidate demonstrates a solid working knowledge of SQL aggregate functions, GROUP BY, and HAVING, and is likely capable of independently completing most standard reporting and data analysis tasks. Some gaps may exist in handling more complex or edge-case scenarios, but overall competence in this area is evident.

Using a 'Sales' table with columns for SalespersonID and SaleAmount, write a query to find each salesperson's total sales, but only include salespersons whose total sales exceed \$10,000.



1

Cannot correctly use SUM, GROUP BY, or uses WHERE instead of HAVING to filter grouped results.



2

Uses SUM and GROUP BY correctly but uses WHERE instead of HAVING or has minor syntax errors.



3



4

Writes a fully correct query using SUM, GROUP BY, and HAVING with clean, accurate syntax.



5

What is an aggregate function in SQL? Can you name one and explain how GROUP BY is used alongside it?



1

Cannot name or correctly describe an aggregate function or GROUP BY.



2

Names an aggregate function correctly but struggles to explain how GROUP BY groups rows.



3



4

Clearly explains aggregate functions and GROUP BY with an accurate, practical example.



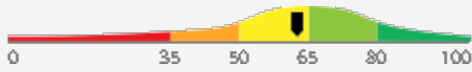
5

Detail

Interview Guide

JOIN Operations (Coding Tasks)

Score: 62



Description:

Covers the use of pointers to reference and manipulate memory addresses, along with dynamic memory allocation and deallocation using malloc, calloc, realloc, and free. Includes pointer arithmetic, dereferencing, and avoiding common issues like memory leaks and dangling pointers.

Interpretation:

The candidate exhibits average writing skills, which can hinder high performance in some jobs.

The candidate possesses a moderate working knowledge of C programming, demonstrating familiarity with core concepts including data types, control flow, functions, and basic file I/O. They may require some guidance when working with more advanced topics such as dynamic memory allocation, modular design, or debugging complex logic.

Overall AI Score:	65.0
Lines of Code:	15.0
Syntax Errors:	5.0
AI Confidence Level:	50
Match with Ideal Response (AI):	30.0
Structure:	50.0
Syntax:	30.0

Please see below to view the essay submitted.

Walk me through how you would dynamically allocate memory for an array of 10 integers, use it, and then properly release it. What issues might arise if you don't follow best practices?



1

Cannot write correct allocation code; unaware of free() or memory leak risks.



2

Writes mostly correct malloc/free code; identifies memory leaks but misses other risks.



3



4

Correct malloc, use, and free; identifies leaks, dangling pointers, and NULL check on allocation.



5

Can you explain what a pointer is in C and describe a situation where you would use one?



1

Vague or incorrect definition; cannot describe a practical use case.



2

Correct basic definition; gives a simple but valid use case with some gaps.



3



4

Clear definition with accurate use case; mentions address storage, dereferencing, or dynamic memory.

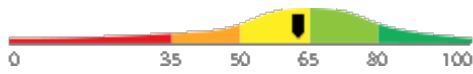


5

Detail Interview Guide

SELECT Statements and Filtering (Coding Tasks)

Score: 62



Description:

Covers the use of pointers to reference and manipulate memory addresses, along with dynamic memory allocation and deallocation using malloc, calloc, realloc, and free. Includes pointer arithmetic, dereferencing, and avoiding common issues like memory leaks and dangling pointers.

Interpretation:

The candidate exhibits average writing skills, which can hinder high performance in some jobs.

The candidate possesses a moderate working knowledge of C programming, demonstrating familiarity with core concepts including data types, control flow, functions, and basic file I/O. They may require some guidance when working with more advanced topics such as dynamic memory allocation, modular design, or debugging complex logic.

Overall AI Score:	65.0
Lines of Code:	15.0
Syntax Errors:	5.0
AI Confidence Level:	50
Match with Ideal Response (AI):	30.0
Structure:	50.0
Syntax:	30.0

Please see below to view the essay submitted.

Walk me through how you would dynamically allocate memory for an array of 10 integers, use it, and then properly release it. What issues might arise if you don't follow best practices?



1

Cannot write correct allocation code; unaware of free() or memory leak risks.



2

Writes mostly correct malloc/free code; identifies memory leaks but misses other risks.



3



4

Correct malloc, use, and free; identifies leaks, dangling pointers, and NULL check on allocation.



5

Can you explain what a pointer is in C and describe a situation where you would use one?



1

Vague or incorrect definition; cannot describe a practical use case.



2

Correct basic definition; gives a simple but valid use case with some gaps.



3



4

Clear definition with accurate use case; mentions address storage, dereferencing, or dynamic memory.

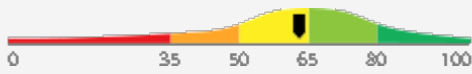


5

Detail Interview Guide

INSERT, UPDATE, and DELETE Statements

Score: 63



Description:

The ability to add new rows to a table using INSERT, modify existing rows using UPDATE, and remove rows using DELETE. Understanding how to use WHERE clauses with UPDATE and DELETE to target specific rows is critical to avoid unintended data changes.

Interpretation:

Candidate appears capable of average job performance in this area with little or no training.

The candidate possesses a moderate understanding of INSERT, UPDATE, and DELETE statements and can perform basic data manipulation tasks. However, their ability to consistently and correctly apply WHERE clauses to precisely target specific rows may be inconsistent or incomplete.

You have an 'Employees' table with columns for EmployeeID, Name, and Salary. Write a SQL statement to increase the salary by 10% for all employees whose current salary is below \$50,000.

- ★
1
- ★
2
- ★
3
- ★
4
- ★
5

Cannot write a correct UPDATE statement or omits the WHERE clause entirely.

Writes a mostly correct UPDATE but has minor errors in the calculation or WHERE clause syntax.

Writes a fully correct UPDATE statement with accurate salary calculation and proper WHERE filtering.

Can you describe what the INSERT, UPDATE, and DELETE statements do in SQL and why it is important to use a WHERE clause with UPDATE and DELETE?

- ★
1
- ★
2
- ★
3
- ★
4
- ★
5

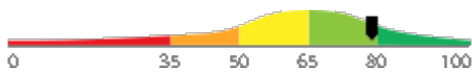
Cannot accurately describe all three statements or does not recognize the risk of omitting WHERE.

Correctly describes all three statements but gives a vague explanation of why WHERE matters.

Accurately describes all three and clearly explains that omitting WHERE affects all rows in the table.

JOIN Operations

Score: 79



Description:

The ability to combine rows from two or more tables using INNER JOIN, LEFT JOIN, RIGHT JOIN, and other join types based on related columns. JOINS are essential for working with relational databases where data is spread across multiple tables.

Interpretation:

Candidate should achieve above average job performance in this area with little or no training.

The candidate demonstrates a solid and proficient understanding of SQL JOIN operations, including the ability to combine rows from multiple tables using various join types such as INNER JOIN, LEFT JOIN, and RIGHT JOIN. They are generally capable of working effectively with relational databases where data is distributed across multiple tables.

You have a 'Customers' table and an 'Orders' table linked by CustomerID. Write a query to return all customers and any orders they have placed, including customers who have not placed any orders.

- ★
1
- ★
2
- ★
3
- ★
4
- ★
5

Uses INNER JOIN or incorrect syntax, missing unmatched customer rows.

Uses LEFT JOIN correctly but may have minor errors in ON clause or column references.

Writes a correct LEFT JOIN query that includes all customers and handles missing orders cleanly.

Can you explain what a JOIN does in SQL and describe the difference between an INNER JOIN and a LEFT JOIN?

- ★
1
- ★
2
- ★
3
- ★
4
- ★
5

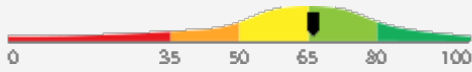
Cannot explain JOINS or confuses INNER and LEFT JOIN behavior.

Explains the general idea of JOINS but is unclear on when LEFT JOIN returns NULLS.

Clearly explains both JOIN types with accurate descriptions of how unmatched rows are handled.

Detail
Interview Guide
SELECT Statements and Filtering

Score: 66


Description:

The ability to write SELECT statements to retrieve data from one or more tables, including the use of WHERE clauses to filter rows, ORDER BY to sort results, and basic column selection and aliasing. This is the most fundamental and frequently used skill in SQL.

Interpretation:

Candidate should achieve above average job performance in this area with little or no training.

The candidate demonstrates a solid working knowledge of basic SQL, including proficiency in writing and interpreting SELECT statements, using aggregate functions, performing JOIN operations, and managing database structures with appropriate constraints. Minor gaps may exist in advanced areas such as set operations, index usage, or complex subqueries, but the candidate is well-positioned to contribute effectively to SQL-related tasks in a business environment with minimal oversight.

Suppose you have a table called 'Orders' with columns for CustomerID, OrderDate, and TotalAmount. Write a SQL query to return all orders placed after January 1, 2023, sorted from the highest to lowest TotalAmount.



1

Cannot write a correct query; missing or incorrect WHERE or ORDER BY syntax.



2

Writes a mostly correct query with minor syntax errors or missing DESC keyword.



3



4

Writes a fully correct, clean query with proper filtering, sorting, and optional aliasing.



5

Can you walk me through what a SELECT statement does and describe how you would use a WHERE clause to filter the results it returns?



1

Vague or incorrect description; cannot explain filtering or column selection.



2

Correctly explains SELECT and WHERE but struggles with examples or edge cases.



3



4

Clearly explains SELECT, WHERE, aliasing, and ORDER BY with confident, accurate examples.



5

IT Coding Tasks

During the assessment, the candidate was asked to write one or more programs or scripts. Their responses are included below for review.

Question or Task	Response
<p>Complete the provided partial C program by filling in the missing sections marked with TODO comments. Your completion must use standard C keywords and libraries.</p> <p>The program declares a function called <code>duplicate_array</code> that:</p> <ol style="list-style-type: none"> 1. Takes a const int pointer to a source array and its length as parameters. 2. Uses <code>calloc</code> to allocate a new int array of the same length. 3. Returns NULL if <code>calloc</code> fails. 4. Copies each element from the source array into the new array using pointer arithmetic (not array subscript notation). 5. Returns the pointer to the newly allocated copy. <p>In main, the program:</p> <ol style="list-style-type: none"> 1. Declares and initializes a stack array of 4 integers with values 5, 15, 25, 35. 2. Calls <code>duplicate_array</code> to create a heap-allocated copy. 3. Checks for NULL and prints an error and returns 1 if the call failed. 4. Prints each element of the duplicate using a loop. 5. Frees the duplicate array. <p>Use appropriate indentation, common C coding conventions, and add brief inline comments where needed. Type your completed source code as your response.</p>	<pre>#include <stdio.h> #include <stdlib.h> int *duplicate_array(const int *src, int length) { /* TODO: Use calloc to allocate a new array of 'length' integers, return NULL if calloc fails, copy elements from src using pointer arithmetic, and return the new pointer. */ calloc(303); } int main(void) { /* TODO: Declare and initialize a stack array of 4 integers: 5, 15, 25, 35, then call duplicate_array and store the result. Check for NULL and print an error message returning 1 if it failed. */ array[4]={5,15,25,35}; int i; /* Print each element of the duplicate */ for (i = 0; i < 4; i++) { printf("duplicate[%d] = %d\n", i, *(duplicate + i)); } /* Free the duplicate array */ free(duplicate); return 0; }</pre>

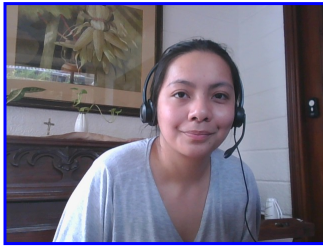
Comments (AI): The code segment has several syntax errors and incomplete implementation. The `duplicate_array` function does not correctly allocate memory or copy elements. The main function has syntax errors and does not properly call the `duplicate_array` function. However, the structure and intent of the code are somewhat clear, and the code attempts to follow the requirements.

Identity Confirmation Photos

The following photos of the candidate and any identification were uploaded during the assessment session.

Photo Analysis Results

- Risk:	Medium risk of cheating based on image inconsistencies
- Percent match among processed faces	100%
- Total images processed	17
- Total images with valid faces	14 (82%)
- Total pairs of faces compared	13
- Pairs in which faces matched	13 (100%)



Pre/Post-Test Photo



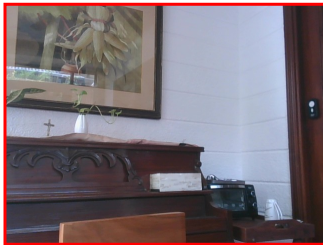
ID Photo



In-Test Error Detected (No Face Detected)



In-Test Error Detected (No Face Detected)



In-Test Error Detected (No Face Detected)



In-Test Photo



In-Test Photo



In-Test Photo



In-Test Photo



Pre/Post-Test Photo

Resume or CV

Summary

Updated on

Motivated career professional with extensive experience in office administration and management. Proven track record of improving efficiency, reducing costs, and enhancing office operations through strategic initiatives and technology implementation.

Objective

I am seeking a role where I can use my many skills and my exceptional judgment and empathy for customers to make a difference to a growing company.

Education

- Associate of Applied Science in Office Administration, Portland Community College, 2020

Experience

- General Office Clerk, Paramount Office Management, 09/2023 – Present
- Administrative Assistant, Global Enterprises Inc., 04/2021 – 08/2023
- Administrative Assistant, Innovative Business Solutions Ltd., 07/2019 – 03/2021

Other Qualifications

- Microsoft Office Specialist (MOS) Certification
- Certified Administrative Professional (CAP)
- International Association of Administrative Professionals (IAAP) Certification

Report Preparation Notes

- Hiring decisions should never be based on a single source of information. The most effective use of this assessment report is as a part of a multi-faceted program of candidate evaluation that includes resume review, interviews, and reference checks.
- Overall vs Percentiles Scores: The overall score reflects the success in the test, based on the mean (average) and standard deviation of the test scores. The percentile score reflects the percentage of test-takers who scored equal or below this overall score. We recommend you use the Overall Score as your primary evaluation criteria. However, percentile scores can often be useful in comparing specific candidates against one another and with a group, such as for test takers in a certain organization or within a certain account.
- Note that comparison information is calculated based on completed instances of this assessment at that time the assessment is scored. As additional instances are completed, the comparative data may change. You can always update a report to the current values by clicking on 'Recalculate Percentiles' within the online results viewing pages at www.hravatar.com.
- Most competency scores are norm-based, which means that they can be interpreted in terms of their distance from the average or mean score. For all scales, a score equal to the mean receives a score of 65 and scores above and below this value are set so that a score change of 15 equals one standard deviation.
- For linear competencies, higher is better across the entire scale. For these scales a score between 65 and 80 (light green) represents 0 to 1 standard deviation above the mean and a score above 80 (dark green) represents more than one standard deviation above the mean. Similarly, a score of 50 - 65 (yellow) represents 0 to 1 standard deviation below the mean, while a score of 35 - 50 (orange) equates to 1 to 2 standard deviations below the mean, and a score below 35 represents more than 2 standard deviations below the mean.
- Sim ID: 20805-1, Key: 0-0, Rpt: 68, Prd: 9627, Created: 2026-06-28 09:15 EDT
- UA: Mozilla/5.0 (Windows NT 6.3; Trident/7.0; Touch; rv:11.0) like Gecko

Score Calculation Detail

The following table provides a summary of how the overall score was calculated from each of the individual competency scores. First, all competency scores are calculated on a scale of 0-100. Note that some competencies use their color category rather than their actual numeric score in the overall calculation. For these, a standard score associated with the assigned color category is used in the overall score calculation rather than the actual numeric score. This is reflected in the "Score Value Used" column. Next, a weighted average of scores is computed using individual competency weights, typically set using job analysis data provided by the US Government Occupational Information Network (O*Net).

Competency	Score	How applied to overall	Score Value Used	Weight (%)
Aggregate Functions and GROUP BY	75.0700	Numeric Score	75.0700	16.6667
INSERT, UPDATE, and DELETE Statements	63.2013	Numeric Score	63.2013	16.6667
JOIN Operations	79.1205	Numeric Score	79.1205	16.6667
JOIN Operations (Coding Tasks)	62.9784	Numeric Score	62.9784	16.6667
SELECT Statements and Filtering	66.4272	Numeric Score	66.4272	16.6667
SELECT Statements and Filtering (Coding Tasks)	62.9784	Numeric Score	62.9784	16.6667
Weighted Average:				68.2960
Final Overall Score:				68

Notes

(This area is intentionally blank - it's reserved as space for your notes.)