hr | avatar
Illuminating Talent

Test Results and Interview Guide

Candidate: **Elizabeth Wantsajob**
Assessment: YAML
Completed: November 4, 2026
Prepared for: Sara Maple
Example Company

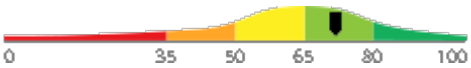
What's Included

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

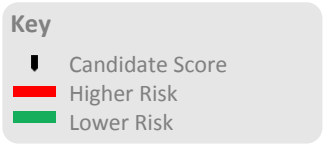
Important Note: The YAML assessment measures key factors related to high performance and tenure in this job. Attribute types measured vary by test, but can include cognitive ability, skills, knowledge, personality characteristics, emotional intelligence, and past behavioral history. This report includes 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.

Proprietary and Confidential

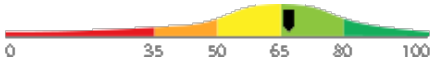
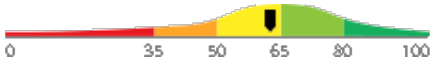
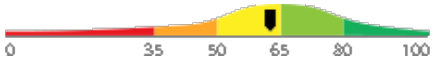
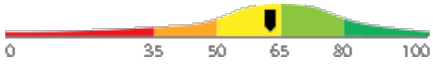
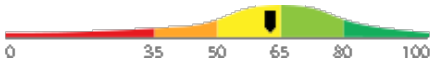
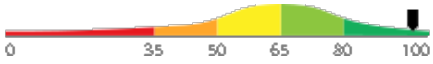
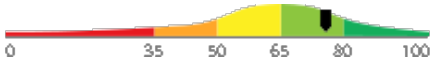
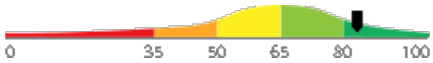
Overall

Candidate	Score	Interpretation
Elizabeth Wantsajob beth.wantsajob@gmail.com YAML November 4, 2026	71	

The candidate demonstrates a solid working knowledge of YAML, including syntax rules, collections, scalar styles, anchors and aliases, and error resolution. They are well-equipped to write, debug, and maintain YAML files in most professional contexts, with only occasional gaps in advanced or edge-case scenarios.



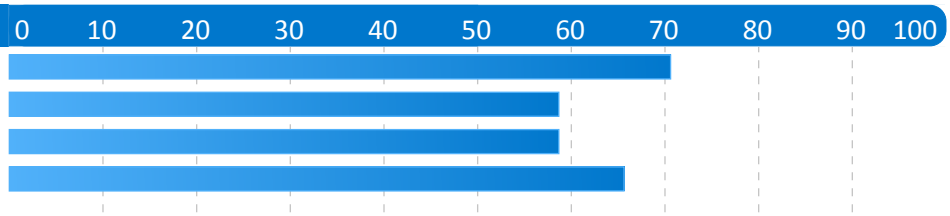
Competency Summary

Competency	Score	Interpretation
Skills/Knowledge (relates to immediate readiness)		
Anchors, Aliases, and Reuse	67	
Anchors, Aliases, and Reuse (Coding Tasks)	62	
Data Types and Collections (Coding Tasks)	62	
Multiline Strings and Scalar Styles (Coding Tasks)	62	
Syntax, Structure, and Indentation (Coding Tasks)	62	
Data Types and Collections	96	
Multiline Strings and Scalar Styles	76	
Syntax, Structure, and Indentation	83	

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
Global	71st
North America	59th
United States	59th
Example Company	66th



Copyright 2026, HR Avatar, Inc. All Rights Reserved.

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	-	-	Summary Points (AI): • (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. Narrative (AI): Elizabeth Wantsajob demonstrates a mixed profile of knowledge, skills, and abilities across the assessed competencies. 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. 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. 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. 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. Computed on: April 2, 2026, 11:09:49PM EDT

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Detail

Candidate:	Elizabeth Wantsajob, beth.wantsajob@gmail.com
Assessment:	YAML
Authorized:	November 4, 2026, by Sara Maple, Example Company, qamailsaram.mike@hravatar.com
IP Address:	96.241.65.10
IP Country:	United States
IP State:	Virginia
IP City:	Gainesville
Started:	November 4, 2026, 4:40:03PM EST
Completed:	November 4, 2026, 4:40:03PM EST
Overall Score:	71

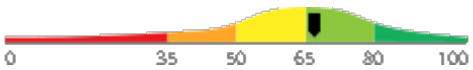
Knowledge and Skills Detail

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

Anchors, Aliases, and Reuse

Score: 67



Description:

Covers how to define reusable content in a YAML file using anchors and reference it elsewhere using aliases. This is a practical technique used in configuration files to avoid repeating the same values or blocks of content across multiple sections.

Interpretation:

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

The candidate demonstrates a solid and competent understanding of YAML anchors and aliases, and can effectively use them to avoid content repetition across configuration file sections. Minor gaps in knowledge may exist for advanced or edge-case applications.

You are working on a YAML configuration file where the same set of environment variables needs to appear under three different service definitions. How would you use anchors and aliases to avoid repeating that block three times?

1	2	3	4	5
Cannot describe how to use anchors and aliases or proposes copying the block manually.		Understands the concept but makes errors in syntax or cannot fully explain how the alias references the anchor.		Accurately describes defining an anchor with ampersand, referencing it with an asterisk alias, and explains the benefit clearly.

Have you ever seen an ampersand or asterisk symbol in a YAML file? Do you know what they are used for?

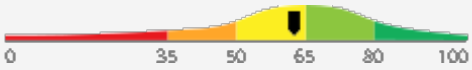
1	2	3	4	5
Has not seen these symbols or cannot associate them with any YAML feature.		Recognizes the symbols are related to YAML but cannot accurately explain their purpose.		Correctly identifies ampersand as an anchor and asterisk as an alias and explains how they enable content reuse.

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Detail

Anchors, Aliases, and Reuse (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.

Interview Guide

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	2	3	4	5
Cannot write correct allocation code; unaware of free() or memory leak risks.		Writes mostly correct malloc/free code; identifies memory leaks but misses other risks.		Correct malloc, use, and free; identifies leaks, dangling pointers, and NULL check on allocation.

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

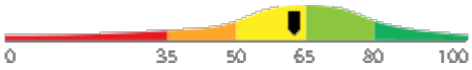
1	2	3	4	5
Vague or incorrect definition; cannot describe a practical use case.		Correct basic definition; gives a simple but valid use case with some gaps.		Clear definition with accurate use case; mentions address storage, dereferencing, or dynamic memory.

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Detail

Data Types and Collections (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.

Interview Guide

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	2	3	4	5
Cannot write correct allocation code; unaware of free() or memory leak risks.		Writes mostly correct malloc/free code; identifies memory leaks but misses other risks.		Correct malloc, use, and free; identifies leaks, dangling pointers, and NULL check on allocation.

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

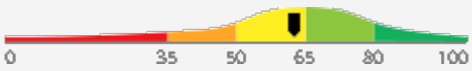
1	2	3	4	5
Vague or incorrect definition; cannot describe a practical use case.		Correct basic definition; gives a simple but valid use case with some gaps.		Clear definition with accurate use case; mentions address storage, dereferencing, or dynamic memory.

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Detail

Multiline Strings and Scalar Styles (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.

Interview Guide

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	2	3	4	5
Cannot write correct allocation code; unaware of free() or memory leak risks.		Writes mostly correct malloc/free code; identifies memory leaks but misses other risks.		Correct malloc, use, and free; identifies leaks, dangling pointers, and NULL check on allocation.

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

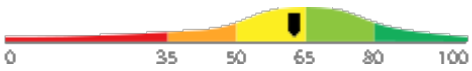
1	2	3	4	5
Vague or incorrect definition; cannot describe a practical use case.		Correct basic definition; gives a simple but valid use case with some gaps.		Clear definition with accurate use case; mentions address storage, dereferencing, or dynamic memory.

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Detail

Syntax, Structure, and Indentation (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.

Interview Guide

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	2	3	4	5
Cannot write correct allocation code; unaware of free() or memory leak risks.		Writes mostly correct malloc/free code; identifies memory leaks but misses other risks.		Correct malloc, use, and free; identifies leaks, dangling pointers, and NULL check on allocation.

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

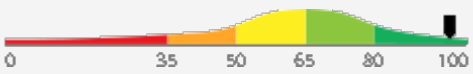
1	2	3	4	5
Vague or incorrect definition; cannot describe a practical use case.		Correct basic definition; gives a simple but valid use case with some gaps.		Clear definition with accurate use case; mentions address storage, dereferencing, or dynamic memory.

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Detail	Interview Guide

Data Types and Collections

Score: 96



Description:

Covers how YAML represents common data types such as strings, integers, booleans, and null values, as well as how to write and work with sequences (lists) and mappings (key-value dictionaries). These structures appear in nearly every YAML file used for configuration or data exchange.

Interpretation:

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

The candidate demonstrates a strong and comprehensive mastery of YAML data types and collections, including precise handling of strings, integers, booleans, null values, sequences, and mappings. They are highly capable of working with complex YAML structures in a wide variety of configuration and data exchange contexts.

You are reviewing a YAML configuration file and notice that a value you expect to be the boolean 'true' is written as the string 'true' in quotes. What is the difference, and when would this distinction matter?

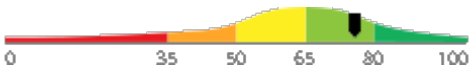
1	2	3	4	5
Does not recognize a difference between quoted and unquoted 'true' or cannot explain the impact.		Recognizes there is a difference but struggles to explain when or why it would cause a problem.		Clearly explains that quoting forces a string type, describes type coercion, and gives a practical example of when it matters.

In a YAML file, how would you write a list of three items, and how would you write a key paired with a number value?

1	2	3	4	5
Cannot produce a correct list or key-value pair with a number; syntax is invalid.		Writes a recognizable list or key-value pair but makes minor syntax errors.		Correctly writes both a sequence with dashes and a numeric key-value pair without errors.

Multiline Strings and Scalar Styles

Score: 76



Description:

Covers how to write string values that span multiple lines using literal block style (pipe character) and folded block style (greater-than character). These techniques are commonly used in configuration files to write commands, messages, or formatted text cleanly and readably.

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 multiline string techniques and scalar styles in YAML. They are capable of effectively applying literal and folded block styles to write clean, readable configuration files across a variety of practical use cases.

What is the difference between using a pipe character and a greater-than character when writing a multiline string in YAML? Can you give an example of when you would choose one over the other?

1	2	3	4	5
Cannot distinguish between the two styles or confuses their behavior.		Correctly identifies one style but cannot clearly explain the difference or give a practical use case.		Accurately explains that pipe preserves newlines while greater-than folds them, and gives a clear practical example for each.

Have you ever needed to write a value in a YAML file that was too long to fit on one line? If so, how did you handle it?

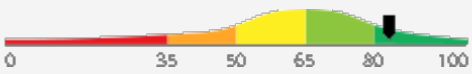
1	2	3	4	5
Has not encountered this or cannot describe any valid approach for multiline values.		Describes a workaround or recalls using a special character but cannot explain how it works.		Describes literal block or folded block styles with the pipe or greater-than character and explains the difference.

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Detail

Syntax, Structure, and Indentation

Score: 83



Description:

Covers the foundational rules for writing valid YAML, including correct use of indentation, spacing, colons, dashes, and overall document structure. This is the most frequently applied area of knowledge when writing, reading, or debugging any YAML file.

Interpretation:

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

The candidate exhibits a comprehensive and advanced understanding of YAML, including complex nested structures, anchors and aliases, multi-document files, type coercion, quoting, escaping, and validation. They are well-equipped to independently write, debug, and maintain sophisticated YAML files and are likely to serve as a knowledgeable resource on YAML best practices and conventions.

Interview Guide

A colleague shares a YAML file that throws a parser error. When you look at it, you notice some lines are indented with tabs and others with spaces. How would you explain the problem and fix it?

1	2	3	4	5
Does not identify tabs vs. spaces as an issue or cannot explain why it causes errors.		Identifies the tab/space conflict but gives an incomplete or partially correct fix.		Clearly explains YAML requires consistent spaces, not tabs, and describes how to correct and prevent it.

Can you describe what a YAML file looks like and explain one or two rules you need to follow when writing one?

1	2	3	4	5
Vague or incorrect description; cannot identify any valid syntax rules.		Identifies basic rules like colons or indentation but lacks detail or accuracy.		Clearly describes key-value pairs, indentation-based nesting, and spacing rules with confidence.

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

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.

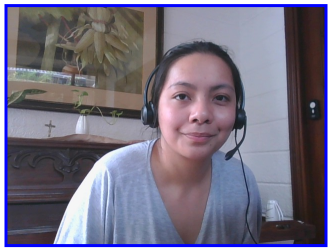
Question or Task	Response
Complete the provided partial C program by filling in the missing sections marked with TODO comments. Your completion must use standard C keywords and libraries. The program declares a function called duplicate_array that: 1. Takes a const int pointer to a source array and its length as parameters. 2. Uses calloc to allocate a new int array of the same length. 3. Returns NULL if calloc 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. In main, the program: 1. Declares and initializes a stack array of 4 integers with values 5, 15, 25, 35. 2. Calls duplicate_array 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. Use appropriate indentation, common C coding conventions, and add brief inline comments where needed. Type your completed source code as your response.	#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; } **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.

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Identity Confirmation Photos

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

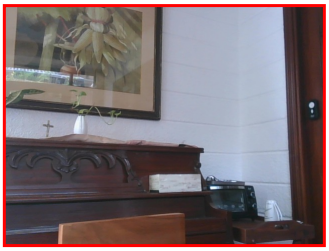
Photo Analysis Results	
- Risk:	Medium risk of cheating based on image inconsistencies
- Percent match among processed faces	100%
- Total images processed	17
- Total images with valid faces	14 (82%)
- Total pairs of faces compared	13
- Pairs in which faces matched	13 (100%)



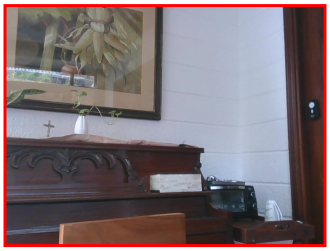
Pre/Post-Test Photo



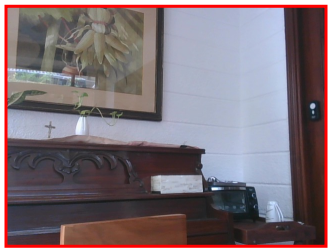
ID Photo



In-Test Error Detected (No Face Detected)



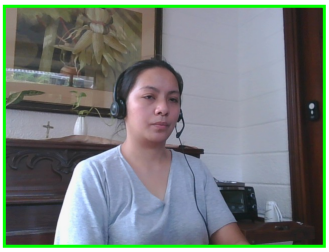
In-Test Error Detected (No Face Detected)



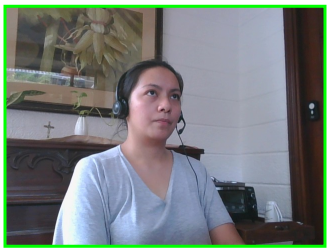
In-Test Error Detected (No Face Detected)



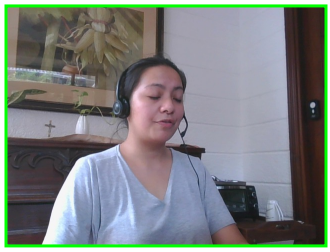
In-Test Photo



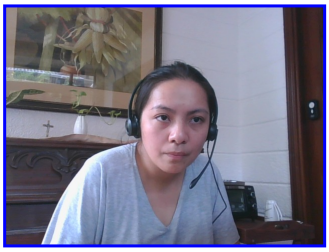
In-Test Photo



In-Test Photo



In-Test Photo



Pre/Post-Test Photo

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

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

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

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: 21150-1, Key: 0-0, Rpt: 68, Prd: 10075, Created: 2026-11-04 16:40 EST
- UA: Mozilla/5.0 (Windows NT 6.3; Trident/7.0; Touch; rv:11.0) like Gecko

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

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 (%)
Anchors, Aliases, and Reuse	67.4544	Numeric Score	67.4544	12.5000
Anchors, Aliases, and Reuse (Coding Tasks)	62.9784	Numeric Score	62.9784	12.5000
Data Types and Collections	96.6392	Numeric Score	96.6392	12.5000
Data Types and Collections (Coding Tasks)	62.9784	Numeric Score	62.9784	12.5000
Multiline Strings and Scalar Styles	76.1593	Numeric Score	76.1593	12.5000
Multiline Strings and Scalar Styles (Coding Tasks)	62.9784	Numeric Score	62.9784	12.5000
Syntax, Structure, and Indentation	83.3715	Numeric Score	83.3715	12.5000
Syntax, Structure, and Indentation (Coding Tasks)	62.9784	Numeric Score	62.9784	12.5000
Weighted Average:				71.9422
Final Overall Score:				71

Copyright 2026, HR Avatar, Inc. All Rights Reserved.

Notes

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

Copyright 2026, HR Avatar, Inc. All Rights Reserved.