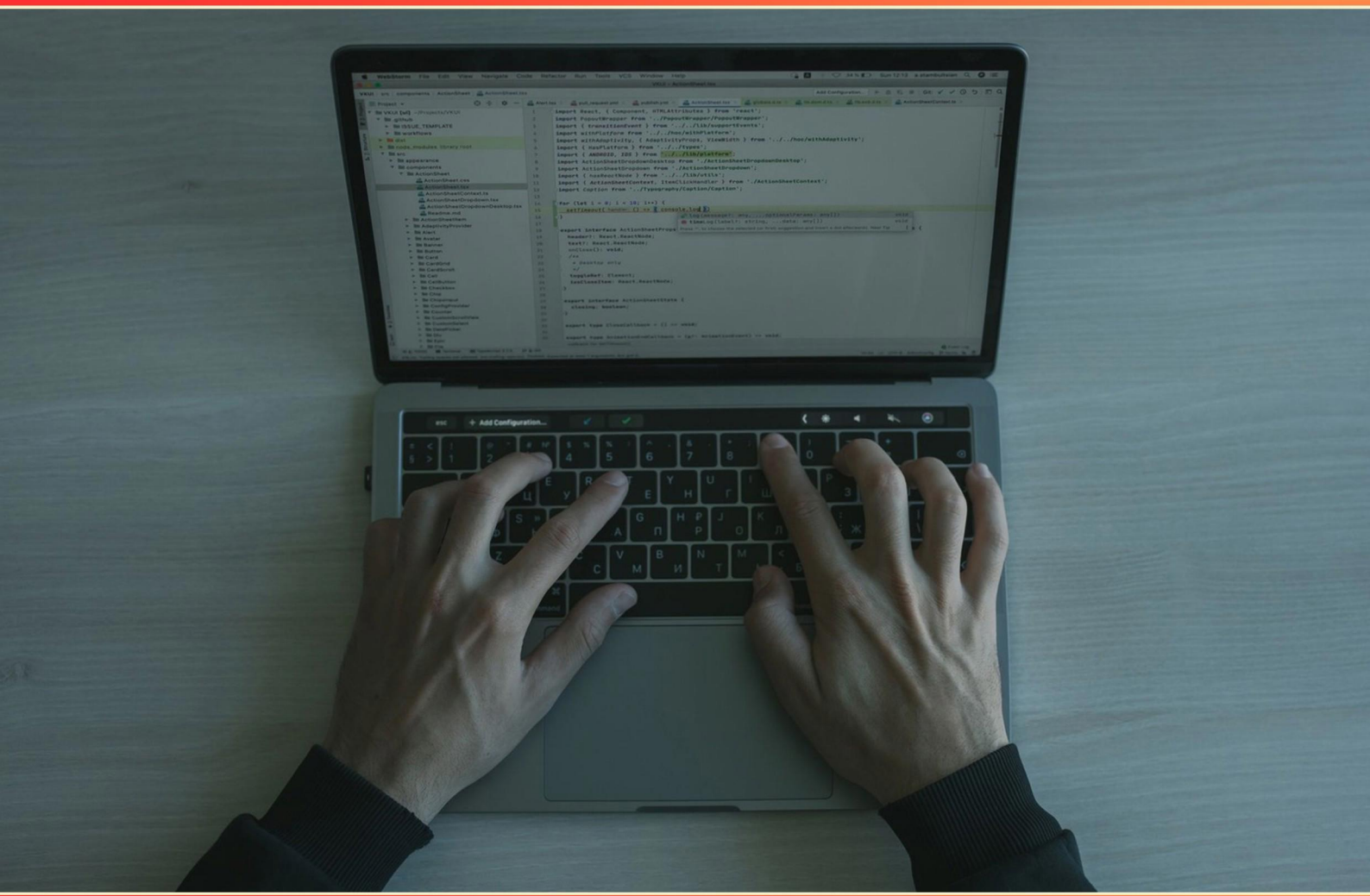
OMNISTUDIO DEVELOPER PRACTICE TEST (SAMPLE)

STUDY GUIDE



SAMPLE STUDY GUIDE WITH LIMITED QUESTIONS

VISIT US ONLINE FOR THE FULL VERSION STUDY GUIDE

https://omnistudiodeveloper.examzify.com

Copyright © 2026 by Examzify - A Kaluba Technologies Inc. product.

ALL RIGHTS RESERVED.

No part of this book may be reproduced or transferred in any form or by any means, graphic, electronic, or mechanical, including photocopying, recording, web distribution, taping, or by any information storage retrieval system, without the written permission of the author.

Notice: Examzify makes every reasonable effort to obtain accurate, complete, and timely information about this product from reliable sources.

SAMPLE

Table of Contents

Copyright 1

Table of Contents 2

Introduction 3

How to Use This Guide 4

Questions 5

Answers 9

Explanations 11

Next Steps 16

SAMPLE

Introduction

Preparing for a certification exam can feel overwhelming, but with the right tools, it becomes an opportunity to build confidence, sharpen your skills, and move one step closer to your goals. At Examzify, we believe that effective exam preparation isn't just about memorization, it's about understanding the material, identifying knowledge gaps, and building the test-taking strategies that lead to success.

This guide was designed to help you do exactly that.

Whether you're preparing for a licensing exam, professional certification, or entry-level qualification, this book offers structured practice to reinforce key concepts. You'll find a wide range of multiple-choice questions, each followed by clear explanations to help you understand not just the right answer, but why it's correct.

The content in this guide is based on real-world exam objectives and aligned with the types of questions and topics commonly found on official tests. It's ideal for learners who want to:

- Practice answering questions under realistic conditions,
- Improve accuracy and speed,
- Review explanations to strengthen weak areas, and
- Approach the exam with greater confidence.

We recommend using this book not as a stand-alone study tool, but alongside other resources like flashcards, textbooks, or hands-on training. For best results, we recommend working through each question, reflecting on the explanation provided, and revisiting the topics that challenge you most.

Remember: successful test preparation isn't about getting every question right the first time, it's about learning from your mistakes and improving over time. Stay focused, trust the process, and know that every page you turn brings you closer to success.

Let's begin.

SAMPLE

How to Use This Guide

This guide is designed to help you study more effectively and approach your exam with confidence. Whether you're reviewing for the first time or doing a final refresh, here's how to get the most out of your Examzify study guide:

1. Start with a Diagnostic Review

Skim through the questions to get a sense of what you know and what you need to focus on. Your goal is to identify knowledge gaps early.

2. Study in Short, Focused Sessions

Break your study time into manageable blocks (e.g. 30 – 45 minutes). Review a handful of questions, reflect on the explanations.

3. Learn from the Explanations

After answering a question, always read the explanation, even if you got it right. It reinforces key points, corrects misunderstandings, and teaches subtle distinctions between similar answers.

4. Track Your Progress

Use bookmarks or notes (if reading digitally) to mark difficult questions. Revisit these regularly and track improvements over time.

5. Simulate the Real Exam

Once you're comfortable, try taking a full set of questions without pausing. Set a timer and simulate test-day conditions to build confidence and time management skills.

6. Repeat and Review

Don't just study once, repetition builds retention. Re-attempt questions after a few days and revisit explanations to reinforce learning. Pair this guide with other Examzify tools like flashcards, and digital practice tests to strengthen your preparation across formats.

There's no single right way to study, but consistent, thoughtful effort always wins. Use this guide flexibly, adapt the tips above to fit your pace and learning style. You've got this!

SAMPLE

Questions

SAMPLE

1. **Which option uses a valid Extract JSON Path for a nested field from related objects when configuring a DataRaptor Extract Action?**

   A. Contact.Name

   B. Contact.Account.Name

   C. Account.Name

   D. Contact.Account.Name.Name

2. **For the integration that sends REST commands after retrieving device details, which OmniScript element should be used to perform the REST API call?**

   A. REST API Action

   B. DataRaptor Extract Action

   C. HTTP Action

   D. Navigation Action

3. **What is the purpose of the Context Data in OmniScript?**

   A. Context Data is used to store user authentication tokens for the session.

   B. Context Data stores values collected during the script and provides a shared data source for steps and mappings.

   C. Context Data is used to log errors only.

   D. Context Data stores values collected during the script and provides a shared data source for steps, mappings, and subsequent actions.

4. **Which JSON code correctly represents the Step1 structure in the OmniScript Structure panel?**

   A. "Step1":{ "Block1":{ "Text1":"Text","Block2":{ "Telephone1":"1234567890" },"Block3":{ "Multi-select1":"Value A;Value B" } } }

   B. "Step1":{ "Block1":{"Text1":"Text","Telephone1":"1234567890","Block2":{"Checkbox1":false}} , "Block3":{"Multi-select1":"ValueA;ValueB"} }

   C. "Step1":{ "Block1":{"Text1":"Text","Telephone1":"1234567890" }, "Block2":{"Checkbox1":false,"Block3":{"Multi-select1":"Value A;value B"}}}

   D. "Step1":{ "Block1":{"Text1":"Text" }, "Block2":{"Telephone1":"1234567890","Checkbox1":false}, "Block3":{"Multi-select1":"ValueA;ValueB"} }

SAMPLE

5. Which OmniStudio element retrieves data from fields in a Salesforce record?

A. DataRaptor Extract Actions

B. HTTP Actions

C. Lookup

D. Select

6. What is the role of a Script in OmniScript?

A. Scripts replace all predefined steps and handle rendering.

B. Scripts are used only for styling and visual appearance.

C. Script steps can perform logic, calculations, validations, or dynamic behavior not covered by predefined steps.

D. Scripts are used to define data sources.

7. What could cause a field in a FlexCard to display only its label while other fields show data?

A. The Field name in the FlexCard state does not match the DataRaptor Extract's Output JSON Path name.

B. In the Setup Panel, {recordId} is misspelled in the Data Source Input Map.

C. There are No Test Data Test Parameters configured.

D. In the FlexCard state, the Field type for the field does not match the type in the Salesforce record.

8. A DataRaptor must convert a string field to an integer before sending to an external service. Which action fixes this?

A. In the DataRaptor Output tab, enter the default value as 0 on the field.

B. In the DataRaptor Extract tab, set the Input Data Type field to Integer.

C. In the DataRaptor Formula tab, define a formula using the function TOINTEGER and use the formula as output.

D. In the DataRaptor Output tab, select the Output Data Type of that field to Integer.

9. During testing of a FlexCard, most fields render correctly but one field only shows the label. Which is the most likely cause?

A. The fields have exceeded the maximum number allowed on the card.

B. The data source key is invalid.

C. There is a mismatch in the Data Source Input Map causing only the field to fail.

D. The field is null in the Salesforce record.

SAMPLE

10. What is the purpose of a Wait step in an Integration Procedure?

A. To fetch data from external service.

B. Introduce a deliberate delay or synchronize timing between steps, often used for polling or queue delays.

C. To cause errors.

D. To execute steps in parallel.

SAMPLE

Answers

SAMPLE

1. B
2. C
3. D
4. B
5. A
6. C
7. A
8. C
9. D
10. B

SAMPLE

Explanations

SAMPLE

1. Which option uses a valid Extract JSON Path for a nested field from related objects when configuring a DataRaptor Extract Action?

A. Contact.Name

B. Contact.Account.Name

C. Account.Name

D. Contact.Account.Name.Name

In a DataRaptor Extract Action, you pull values by following the data shape with a JSON Path that can traverse related records. To get a field from a related object, you navigate through the relationship from the primary object using dot notation. If you start from a Contact and want the Name on its related Account, the correct path is Contact.Account.Name. This directly reflects the nested relationship and retrieves the Name from the linked Account. The other options either reference a field on the primary object itself (Contact.Name), reference a root object that isn't the starting point for this nested lookup (Account.Name), or add an extra, invalid segment (Contact.Account.Name.Name) that doesn't correspond to a real field in the data structure.

2. For the integration that sends REST commands after retrieving device details, which OmniScript element should be used to perform the REST API call?

A. REST API Action

B. DataRaptor Extract Action

C. HTTP Action

D. Navigation Action

In OmniScript, when you need to reach out to an external REST service after you've collected device details, you use the HTTP Action. This element is built specifically to perform HTTP requests, letting you choose the method (GET, POST, PUT, DELETE), specify the endpoint URL (which can incorporate values from the device details), set headers (like Content-Type or Authorization), and include a request body if needed. The response can then be mapped and used by subsequent steps in the script. DataRaptor Extract Action is for pulling data through a DataRaptor, not for making external API calls, and a Navigation Action only changes the flow. There is also a REST API Action in some setups, but the standard, universally supported choice to perform REST calls within OmniScript is the HTTP Action.

3. What is the purpose of the Context Data in OmniScript?

A. Context Data is used to store user authentication tokens for the session.

B. Context Data stores values collected during the script and provides a shared data source for steps and mappings.

C. Context Data is used to log errors only.

D. Context Data stores values collected during the script and provides a shared data source for steps, mappings, and subsequent actions.

Context Data acts as the run-time data store for an OmniScript. As the script runs, values captured from user inputs, API responses, or computed results are written into Context Data, and because it's global within the script, every step can read and use those values. This shared source is what you bind in steps and mappings, and it also drives later actions based on what was gathered. It's not limited to authentication tokens or error logging; it's the central place for all data the script uses and passes along, enabling dynamic behavior as the flow progresses. In short, Context Data stores values collected during the script and provides a shared data source for steps, mappings, and subsequent actions.

SAMPLE

4. Which JSON code correctly represents the Step1 structure in the OmniScript Structure panel?

A. "Step1":{ "Block1":{ "Text1":"Text","Block2":{ "Telephone1":"1234567890" },"Block3":{ "Multi-select1":"Value A;Value B" } } }

B. "Step1":{ "Block1":{"Text1":"Text","Telephone1":"1234567890","Block2":{"Checkbox1":false}}, "Block3":{"Multi-select1":"ValueA;ValueB"} }

C. "Step1":{ "Block1":{"Text1":"Text","Telephone1":"1234567890" }, "Block2":{"Checkbox1":false,"Block3":{"Multi-select1":"Value A;value B"}}}

D. "Step1":{ "Block1":{"Text1":"Text" }, "Block2":{"Telephone1":"1234567890","Checkbox1":false}, "Block3":{"Multi-select1":"ValueA;ValueB"} }

In OmniScript, a Step is represented as an object whose properties are blocks, and each block holds the fields for that section. The correct JSON shows Block1 containing its own fields Text1 and Telephone1, and it also includes a nested sub-block Block2 with its field Checkbox1. Block3 stands as a separate block that holds the Multi-select1 field. This layout reflects a hierarchical structure where a main block can have nested sub-blocks while other blocks remain at the same level as the main block. The other options either nest blocks incorrectly or place all blocks at the same level without the intended nesting, which doesn't align with how the Step1 structure is organized in the Structure panel.

5. Which OmniStudio element retrieves data from fields in a Salesforce record?

A. DataRaptor Extract Actions

B. HTTP Actions

C. Lookup

D. Select

DataRaptor Extract is built to pull data from Salesforce records by selecting the specific object and fields you need. It runs the data extraction and returns those field values which you map into your OmniScript flow. This makes it the right tool for retrieving data from fields on a Salesforce record. HTTP Actions fetch data from external services, not Salesforce records. Lookup resolves values from datasets or related records but isn't designed for extracting a full set of fields from a Salesforce record. Select is for filtering or choosing data you already have, not for performing the initial extraction of Salesforce record fields.

SAMPLE

6. What is the role of a Script in OmniScript?

A. Scripts replace all predefined steps and handle rendering.

B. Scripts are used only for styling and visual appearance.

C. Script steps can perform logic, calculations, validations, or dynamic behavior not covered by predefined steps.

D. Scripts are used to define data sources.

Scripts in OmniScript provide a way to run custom logic that isn't covered by the built-in steps. They let you perform calculations, validations, and dynamic behavior that go beyond what predefined steps can do. Because a Script can access and update the OmniScript data model, it can read values, compute new ones, enforce complex rules, and influence which parts of the flow or UI should run next. This makes Script steps ideal for handling logic that requires more than the standard steps can supply. This isn't about replacing all predefined steps or handling rendering, and it isn't solely for styling or visual appearances. It also isn't used to define data sources, which are managed by data-related steps like Read or Connect.

7. What could cause a field in a FlexCard to display only its label while other fields show data?

A. The Field name in the FlexCard state does not match the DataRaptor Extract's Output JSON Path name.

B. In the Setup Panel, {recordId} is misspelled in the Data Source Input Map.

C. There are No Test Data Test Parameters configured.

D. In the FlexCard state, the Field type for the field does not match the type in the Salesforce record.

The binding between the FlexCard field and the data source is what determines whether a value shows up. In a FlexCard, each field pulls its value from the DataRaptor Extract output using a specific Output JSON Path name. If the Field name in the FlexCard state doesn't exactly match that Output JSON Path, the card can't locate the value to display. The result is that the field renders only its label (which is static) while the actual data stays unreadable because the binding points to nothing. The fix is to align the FlexCard field binding with the exact Output JSON Path name produced by the DataRaptor Extract. Other scenarios would produce different symptoms: a misspelled recordId would likely prevent data retrieval entirely or cause an error; no test data parameters affect testing rather than the live binding; a type mismatch could cause data formatting issues but wouldn't specifically explain a missing bound value while the label remains.

SAMPLE

8. A DataRaptor must convert a string field to an integer before sending to an external service. Which action fixes this?

A. In the DataRaptor Output tab, enter the default value as 0 on the field.

B. In the DataRaptor Extract tab, set the Input Data Type field to Integer.

C. In the DataRaptor Formula tab, define a formula using the function TOINTEGER and use the formula as output.

D. In the DataRaptor Output tab, select the Output Data Type of that field to Integer.

Converting the value at transformation time with a formula is the reliable way to ensure the string becomes an integer before it's sent out. In the DataRaptor Formula tab you can create a formula that uses the TOINTEGER function on the string field, then use that formula's result as the output value. This explicit conversion guarantees the external service receives an integer, rather than a string. Other approaches don't perform a proper runtime conversion. Setting a default value only fills in when the field is missing and doesn't convert the incoming string. Changing the input data type in the Extract tab affects how data is read, not how it's transformed for output. Changing the output data type alone doesn't convert the actual value; it merely declares how the value should be treated on output without performing the conversion itself.

9. During testing of a FlexCard, most fields render correctly but one field only shows the label. Which is the most likely cause?

A. The fields have exceeded the maximum number allowed on the card.

B. The data source key is invalid.

C. There is a mismatch in the Data Source Input Map causing only the field to fail.

D. The field is null in the Salesforce record.

FlexCard data binding shows a field's label when the value from the data source is missing. In this setup, the label is static text that identifies the field, while the value area reflects the data resolved from the data source. If the Salesforce record's field value is null, there is nothing to display in the value portion, so only the label remains visible. This behavior indicates the data binding exists but the underlying data is absent for that record. To confirm, check the actual value in Salesforce for that field on the specific record; if it's null, the observed UI is expected. Other issues, like an invalid data source key or a mismatch in the Data Source Input Map, would typically disrupt the binding more broadly—often preventing the value from rendering at all or causing an error—rather than leaving only the value blank while the label stays.

10. What is the purpose of a Wait step in an Integration Procedure?

A. To fetch data from external service.

B. Introduce a deliberate delay or synchronize timing between steps, often used for polling or queue delays.

C. To cause errors.

D. To execute steps in parallel.

Wait steps create a deliberate pause in the flow of an Integration Procedure. The purpose is to delay the next action to give time for an asynchronous process to complete or to space out repeated checks. This is especially useful for polling a backend or a queue, where you wait a short period between status checks instead of hammering the external service. It doesn't fetch data by itself—that's what a data fetch or HTTP action does. It also isn't used to intentionally produce errors, nor to run steps in parallel; the flow pauses at the wait and resumes only after the specified delay or once a condition is met.

SAMPLE

Next Steps

Congratulations on reaching the final section of this guide. You've taken a meaningful step toward passing your certification exam and advancing your career.

As you continue preparing, remember that consistent practice, review, and self-reflection are key to success. Make time to revisit difficult topics, simulate exam conditions, and track your progress along the way.

If you need help, have suggestions, or want to share feedback, we'd love to hear from you. Reach out to our team at hello@examzify.com.

Or visit your dedicated course page for more study tools and resources:

https://omnistudiodeveloper.examzify.com

We wish you the very best on your exam journey. You've got this!

SAMPLE