



Download Free Salesforce Plat-Con-201 Exam PDF | PrepBolt

Don't miss out! Download the latest free Salesforce Certified Omnistudio Consultant exam PDF questions. Access real Plat-Con-201 dumps with verified answers and boost your chances to pass your certification on the first try with [PrepBolt](https://prepbolt.com/Plat-Con-201.html) Plat-Con-201 exam pdf questions and answers.

Thank you for Downloading Plat-Con-201 exam PDF Demo

<https://prepbolt.com/Plat-Con-201.html>

QUESTIONS & ANSWERS
DEMO VERSION
(LIMITED CONTENT)

Question 1

Question Type: MultipleChoice

Which of the following is a key difference between Integration Procedures and DataRaptors?

Options:

- A- DataRaptors can access data from external sources.
- B- DataRaptors can invoke multiple actions in a single server call.
- C- Integration Procedures can access data from external sources.
- D- Integration Procedures can retrieve data from multiple related objects.

Answer:

C

Question 2

Question Type: MultipleChoice

An OmniStudio Consultant needs to link a new Case to an existing Contact, but the input data only contains the Contact's Name, not the Salesforce ID.

How should the consultant use the Data Mapper Load to resolve this Salesforce ID mapping?

Options:

- A- Map the Contact Name to the Case ContactId field in the Mapping tab.
- B- Enable the Auto-Link property within the Data Mapper Options tab.
- C- Configure a Formula to convert the Name string into a Record ID.
- D- Define a Lookup mapping on the Contact object using the Name field.

Answer:

D

Explanation:

The Data Mapper Load must resolve the related Contact record before it can populate the Case

ContactId lookup field. Mapping the Contact Name directly into ContactId is invalid because ContactId expects a Salesforce record ID, not a text name. A formula cannot reliably convert a name string into a Salesforce ID. 'Auto-Link' is not the correct Data Mapper Load mechanism. The proper approach is to define a lookup mapping that searches the Contact object using the available field value, then uses the resolved Contact record ID for the Case relationship. Salesforce Data Mapper Load documentation confirms that Loads write data to Salesforce objects, and related-object/link mapping patterns are used to populate lookup relationships between records. In a real project, a unique field is safer than Name because names may not be unique.

=====

Question 3

Question Type: MultipleChoice

A public sector agency wants to digitize a paper-based permit application. The process requires a citizen to step through several screens of questions, upload a PDF document, and sign the application digitally.

Which OmniStudio tool serves this specific use case?

Options:

- A- Calculation Matrices
- B- OmniScript
- C- Integration Procedures
- D- FlexCards

Answer:

B

Explanation:

OmniScript is the correct tool because the requirement is a guided, step-by-step digital interaction. The citizen must move through several screens, answer questions, upload supporting documentation, and complete an application process. Calculation Matrices are for rules-based calculations, not user-facing permit forms. Integration Procedures handle server-side orchestration and integrations, but they do not provide the guided UI experience. FlexCards are used to display contextual information and actions, not to run a multi-screen application journey. Salesforce specifically positions OmniScripts as guided processes for customers, partners, and internal users, and Salesforce Public Sector documentation describes OmniScripts as tools for dynamic application forms for licenses, permits,

programs, and services. Salesforce also supports file upload elements within OmniScripts.

=====

Question 4

Question Type: MultipleChoice

An enterprise architecture team is deciding between using Salesforce Flow or OmniStudio Integration Procedures for a specific integration scenario involving complex JSON transformation and multiple external API calls.

Which justification supports choosing an Integration Procedure in this landscape?

Options:

- A- Restrict JSON handling.
- B- Perform server-side orchestration and JSON transformation.
- C- Require HTTP callouts exclusively.
- D- Execute entirely on the client side.

Answer:

B

Explanation:

Integration Procedures are the stronger fit when the scenario requires server-side orchestration, multiple external API calls, and JSON transformation. They are designed to coordinate Data Mapper actions, HTTP actions, conditional logic, response shaping, and integration steps in a single server-side process. That makes them suitable for OmniStudio solutions where the client should receive a clean, optimized response rather than manage multiple complex service calls. Option A is the opposite of the requirement because Integration Procedures support JSON handling rather than restrict it. Option C is too narrow because Integration Procedures do more than HTTP callouts. Option D is incorrect because Integration Procedures execute server-side, not entirely in the browser. Salesforce describes Integration Procedures as server-side processes for automating data operations between Salesforce and external systems.

=====

Question 5

Question Type: MultipleChoice

During a Data Mapper Load execution, a record locking error occurs.

What does this indicate about the data processing or volume?

Options:

- A- Individual record updates are failing due to a lack of field-level permissions.
- B- The target Salesforce record is currently being archived by a system job.
- C- The Data Mapper Load exceeds the maximum batch size for the object.
- D- Multiple concurrent processes are attempting to update the same record.

Answer:

D

Explanation:

A record locking error normally indicates concurrent update contention: two or more transactions are trying to modify the same Salesforce record, or related locked records, at the same time. During a Data Mapper Load, this can happen when multiple users, automations, batch jobs, integrations, or OmniStudio processes attempt to update the same parent or child record concurrently. It is not primarily a field-level security issue; permission failures produce authorization errors. It also does not automatically mean the record is being archived or that the object batch size has been exceeded. The fix is usually architectural: reduce concurrent writes to the same record, serialize critical updates, review automation on the object, and avoid unnecessary parent-record updates. Salesforce documents `UNABLE_TO_LOCK_ROW` style behavior as a concurrent update/record lock problem.

=====

Question 6

Question Type: MultipleChoice

A business process requires the user to upload a file in the OmniScript. The file ID needs to be linked to the parent record being created.

Which approach ensures the link is established correctly?

Options:

- A- The Data Mapper Load that creates the parent record should output its ID.
- B- OmniScripts cannot handle file uploads.
- C- Use a Data Mapper Extract to find the file on the user's computer.
- D- The file upload happens automatically without any ID linking.

Answer:

A

Explanation:

The parent record's Salesforce ID is required to create the correct relationship between the uploaded file and the newly created parent record. Therefore, the Data Mapper Load that creates the parent record must return/output the new record ID so the OmniScript or subsequent data action can use it to link the file properly. OmniScripts can support file upload use cases, so option B is false. A Data Mapper Extract cannot search the user's local computer; it retrieves Salesforce data. The link is not guaranteed automatically in every multi-step process unless the parent record ID is available and passed into the file-linking step. Salesforce Data Mapper Load examples show that linking related records requires carrying the created parent ID into the related mapping/linking process.

=====

Question 7

Question Type: MultipleChoice

A governance policy states that hard-coded IDs must never be present in deployed components.

How should an OmniStudio Consultant address this in Data Mappers?

Options:

- A- Hard-coded IDs are acceptable if they are for standard Record Types.
- B- Use relationship queries or parameters to resolve IDs dynamically.
- C- Hard code the ID in Development, then manually change it in Production after deployment.
- D- Use a lookup table in Excel to find the IDs.

Answer:

Explanation:

Hard-coded Salesforce IDs are a deployment anti-pattern because IDs vary across environments. A Record Type, Account, Contact, Price Book, or configuration record may have one ID in Development and a different ID in QA or Production. The correct design is to resolve IDs dynamically using parameters, relationship fields, unique keys, DeveloperName values, or Data Mapper lookup mappings. Manually changing IDs after deployment is fragile and violates governance. Excel lookup tables are outside the controlled metadata/configuration flow and are not reliable. Even standard-looking references should not be hard-coded unless Salesforce guarantees portability, which is not the normal case for org-specific record IDs. Salesforce Data Mapper Load configuration supports lookup behavior that uses a field value to query Salesforce data and write the resolved result to the output field.

=====

Question 8

Question Type: MultipleChoice

An OmniScript steps through a wizard. In Step 2, the user needs to see details of a specific Case selected in Step 1.

How should an OmniStudio Consultant fetch the data efficiently?

Options:

- A- Use a Data Mapper Load to read the Case data.
- B- Fetch all Cases in the Init action and filter them in Step 2, and use a new CaseId as the input filter.
- C- Use a Data Mapper Extract Action between Step 1 and Step 2, using the CaseId from Step 1 as the input filter.
- D- Ask the user to re-enter the Case details in Step 2.

Answer:

C

Explanation:

The efficient pattern is to retrieve only the selected Case when the user reaches Step 2. A Data Mapper Extract Action can use the CaseId selected in Step 1 as an input filter and return only the

required Case details. This avoids loading unnecessary records and keeps the OmniScript responsive. A Data Mapper Load is wrong because Loads write data to Salesforce; they do not read Case details. Fetching all Cases in the Init action is inefficient and can create performance and governor-limit risks. Asking the user to re-enter Case information is poor user experience and creates data-quality problems. Salesforce OmniStudio guidance separates Data Mapper Extract for reading Salesforce data from Data Mapper Load for saving updates.

=====

Thank You for trying Plat-Con-201 PDF Demo

To try our Plat-Con-201 practice exam software
visit link below

<https://prepbolt.com/Plat-Con-201.html>

Start Your Plat-Con-201 Preparation

Use Coupon “**SAVE50**” for extra 50% discount on the purchase of
Practice Test Software. Test your Plat-Con-201 preparation with actual
exam questions.