PDFVCE

Choose the version that fits your needs	PDF Version	Desktop Test Engine	Online Test Engine
Latest and Up-to-Date exam dumps with real exam questions answers.	✓	✓	✓
Get 12-Months free updates without any extra charges.	✓	✓	✓
Experience same exam environment before appearing in the certification exam.	✗	✓	✓
100% exam passing guarante in the first attempt.	✓	✓	✓
20% discount on more than one license and 30% discount on 5+ license purchases.	✗	✓	✓
100% secure purchase on SSL.	✓	✓	✓
Completely private purchase without sharing your personal info with anyone.	✓	✓	✓

Exam : **AD0-E902**

Title : Adobe Workfront Fusion Professional

Vendor : Adobe

Version : DEMO

NO.1 A Fusion scenario is making too many requests to a third-party API, which returns a 429 "Too Many Requests" error Which technique reduces the number of API requests?

A. Using a Search module to get record IDs and then read those IDs with a Read Record module to pull other data

B. Moving Search and GET modules earlier in the scenario instead of pulling more data about the same record multiple times

C. Adding a Retry error handling directive to the Fusion scenario

Answer: B

Explanation:

* Understanding the Issue:
* The scenario is making too many API requests, causing the third-party API to return a429 "Too Many Requests"error, which indicates that the rate limit has been exceeded.
* The solution needs to reduce unnecessary or redundant API requests to prevent hitting the API limits.
* Why Option B is Correct:
* Avoid Redundant Requests:
* PlacingSearchandGETmodules earlier in the scenario ensures that all required data is retrieved in one batch or in fewer requests, rather than repeatedly querying the same record later in the scenario.
* This technique reduces the overall number of API requests sent to the third-party system.
* Efficient Data Flow:
* By structuring the scenario to retrieve all necessary data at the beginning, subsequent modules can reuse the retrieved data instead of making additional API calls.
* Why the Other Options are Incorrect:
* Option A ("Using a Search module and then a Read Record module"):
* This approach can increase API requests, as theSearch moduleretrieves record IDs, and the Read Record modulemakes separate API requests for each record. This often results in more requests than necessary.
* Option C ("Adding a Retry error handling directive"):
* Adding aRetry directivedoes not reduce the number of requests. Instead, it retries failed requests, which could worsen the problem by increasing API traffic.
* Best Practices to Reduce API Requests:
* Consolidate data retrieval into a single module or a smaller number of requests.
* Use caching or intermediate storage (like Fusion Data Stores) to avoid re-fetching the same data.
* Limit the scope of Search modules by using filters or pagination to process smaller, relevant data sets.

References and Supporting Documentation:

* Adobe Workfront Fusion Best Practices: Managing API Rate Limits
* Workfront Community: Error 429 Solutions

NO.2 Which action makes it possible to see the exact API request and the response a module executes?

A. Using the Bundle Inspector

B. Using the execution history

C. Using the Fusion DevTool scenario debugger

D. Using the Fusion DevTool error evaluator

Answer: B

Explanation:

* Understanding the Requirement:
* The user needs to view the exactAPI requestand the correspondingresponsea module executes in Adobe Workfront Fusion.
* This is critical for debugging, troubleshooting, or validating API operations within scenarios.
* Why Option B is Correct:
* Execution History:
* Theexecution historylogs detailed information about every module that runs in a scenario.
* It provides access to theAPI requestsent, including the headers, parameters, and body.
* It also displays theAPI responsereceived, including HTTP status codes, returned data, and error messages (if applicable).
* This feature is indispensable for debugging and verifying the behavior of modules.
* Why the Other Options are Incorrect:
* Option A ("Using the Bundle Inspector"):
* The Bundle Inspector provides a view of processed data bundles but does not include API request/response details.
* Option C ("Using the Fusion DevTool scenario debugger"):
* Fusion does not have a specific "DevTool debugger." The execution history serves this purpose.
* Option D ("Using the Fusion DevTool error evaluator"):
* While error logs help evaluate issues, they do not directly show the API request/response unless an error occurs. Execution history is a more comprehensive source of this data.
* Steps to View Execution History:
* Run the scenario or inspect a previously executed scenario.
* Navigate to theExecution Historytab for the scenario.
* Select a specific module to view itsdetails.
* Inspect theAPI request and response, which includes all relevant parameters and data.

References and Supporting Documentation:

* Adobe Workfront Fusion Documentation: Execution History
* Workfront Community: Debugging with Execution History

NO.3 A Fusion scenario uses an HTTP module to create a new record.

Which response code indicates that the connection was successful?

A. GREEN

B. 200

C. 402

D. 500

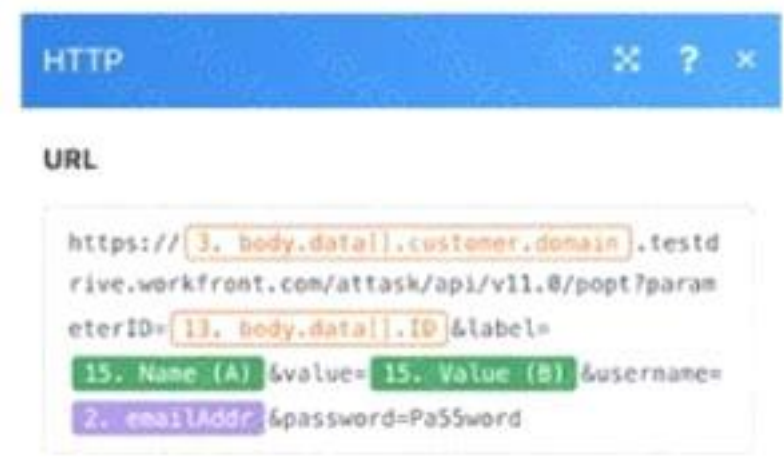
Answer: B

Explanation:

* Understanding HTTP Response Codes:HTTP response codes are standardized codes that indicate the result of a request made to a server:
* 2xx (Success): Indicates that the request was successfully received, understood, and processed by the server.
* 200 OK: Specifically means that the request was successful, and the response contains the

requested data or confirms the operation's success.
* Response Code for Creating a Record:
* When using an HTTP module in Fusion to create a new record, a response code of200confirms that the request to the server was successfully processed and the record creation was successful.
* Why Not Other Options?
* A. GREEN: This is not a valid HTTP response code. It might represent a status in some systems but is unrelated to HTTP standards.
* C. 402: This code indicates a payment required error, meaning the request cannot be fulfilled until payment is made.
* D. 500: This is a server-side error, indicating that something went wrong on the server during processing.
References:
* HTTP Status Code Documentation: 200 Success Response
* Adobe Workfront Fusion Documentation: HTTP Module and Response Codes

NO.4 A user needs to dynamically create custom form field options in two customer environments.



Given this image, which type of Workfront module is referenced in the formula with the parameterID value?
A. Custom API Call
B. Misc Action
C. Read Related Records
D. Search
Answer: A
Explanation:
* Understanding the Image and Context:
* The image provided represents anHTTP modulein Workfront Fusion with a URL that dynamically references various data points (e.g., parameterID, customer.domain, emailAddr).
* The structure of the URL indicates a call to the Workfront API (/api/v1.0/), using parameters to pass dynamic data such as parameterID, username, and password.
* Why Option A ("Custom API Call") is Correct:
* The HTTP module shown in the image is acustom API callbecause it interacts with Workfront's API endpoints by passing dynamic parameters through the URL.
* Custom API Callmodules allow users to manually configure requests to endpoints in cases where no

predefined Workfront Fusion module exists for the operation. This is evident in the example, where specific fields like parameterID, customer.domain, and others are manually mapped to the API URL.
* Example Use Case: Dynamically creating custom form field options by sending a POST/PUT request to the Workfront API with specific parameters (like label and value) for each environment.
* Why the Other Options are Incorrect:
* Option B ("Misc Action"): This refers to predefined actions in Workfront Fusion for handling simple tasks. The HTTP module is not categorized under Misc Actions as it involves direct API interaction.
* Option C ("Read Related Records"): This module is used to fetch data related to Workfront objects (e.g., related tasks or documents). It doesn't allow dynamic parameter passing or URL customization as seen here.
* Option D ("Search"): The Search module is used for querying Workfront objects based on specific criteria but does not involve making direct API calls or sending HTTP requests with custom parameters.
* Steps to Configure a Custom API Call in Workfront Fusion:
* Add theHTTP Moduleto your scenario.
* Select the appropriate HTTP method (e.g., GET, POST, PUT). In this case, aPOSTorPUT method would be used to create or update custom form fields.
* Enter the API endpoint in theURLfield, as shown in the image.
* Map dynamic values to the parameters by referencing fields from previous modules in the scenario. For instance:
* customer.domain: Extracted from prior steps.
* parameterID, label, and value: Dynamically passed based on input data.
* Authenticate the request using a username and password or an API token.
* Test the module to ensure the API call works as expected.
* How This Solves the Problem:
* By using a Custom API Call (via the HTTP module), the user can dynamically interact with the Workfront API to create or modify custom form field options across multiple customer environments, passing the required parameters programmatically.
References and Supporting Documentation:
* Adobe Workfront Fusion HTTP Module Documentation
* Workfront API Documentation
* Workfront Fusion Community Forum: Using HTTP Module for API Calls

NO.5 In a scenario that searches for recently completed tasks, a user notices the following input and output for a date transformation.
Input: March 3, 2021 10:34 AM Output: March 1, 2021 10:34 AM
Which expression produces this date transformation?

A. subDays(now,2)

B. addHours(now; -48)

C. addDays(today; -4)

Answer: A
Explanation:
* Understanding the Date Transformation:
* Input:March 3, 2021, 10:34 AM
* Output:March 1, 2021, 10:34 AM
* The transformation subtracts2 daysfrom the input date without altering the time.

* Why Option A is Correct:
* subDays(now,2) subtracts exactly 2 days from the given date and time.
* It preserves the time component of the input (10:34 AM) while shifting the date backward by 2 days, which matches the given output.
* Why the Other Options are Incorrect:
* Option B ("addHours(now; -48)"): While subtracting 48 hours also results in a 2-day difference, this approach directly modifies the time. The resulting time could shift if the operation crosses daylight saving changes or edge cases with leap seconds. It is less reliable compared to subDays.
* Option C ("addDays(today; -4)"): This would subtract 4 days, which does not match the transformation shown in the example.
References and Supporting Documentation:
* Adobe Workfront Fusion: Date and Time Functions
* Workfront Community: Using Date and Time Expressions