ATMIC NETWORKS
YOUR GUIDE TO GLOBAL CERTIFIATION

"Please note that these files may not be up to date. However, the questions will help you understand the exam format and typical question patterns."

www.atmicnetworks .com

Warning: Keep connected with our support team for latest updates

Question: 1

In which case is the customization of a native OAuth provider not immediately ready to be used?

A. One or more "Transform" policies are used in the assembly.
B. Grant types are added.
C. A logic operator is used in the assembly.
D. A policy with a reference to a TLS profile is added.

Answer: D

Explanation:

OAuth Provider Customization: When customizing a native OAuth provider in IBM API Connect, certain configurations and policies can affect its readiness for immediate use.
TLS Profile Reference: Adding a policy that references a TLS (Transport Layer Security) profile introduces additional security configurations that need to be validated and applied. This process can delay the immediate readiness of the OAuth provider.
Impact of TLS Profiles: TLS profiles are used to secure communications and ensure data integrity and confidentiality. When a policy with a TLS profile reference is added, the system must ensure that the TLS settings are correctly configured and operational, which can take additional time.
Other Options:
Transform Policies: These are used to modify the request or response messages and do not inherently delay the readiness of the OAuth provider.
Grant Types: Adding grant types involves configuring the OAuth provider to support different methods of obtaining access tokens, which is a standard part of customization and does not delay readiness.
Logic Operators: Using logic operators in the assembly is related to the flow control of the API assembly and does not directly impact the readiness of the OAuth provider.
Reference:
IBM API Connect documentation and best practices for OAuth provider customization.
General principles of TLS and its impact on API security configurations.

Question: 2

Given the API Endpoint, "https://cor.client.rtc/savings/annual/", which statement is correct with default Catalog settings?

A. "annual" is the name of the Catalog the API is deployed to.
B. "savings" is the name of the gateway cluster the Catalog is set to.
C. "savings" is the name of the Catalog the API is deployed to.
D. "annual" is the name of the API being called.

Answer: C

Explanation:

API Endpoint Structure: In IBM API Connect, the structure of an API endpoint URL typically includes the base URL, followed by the catalog name, and then the API name.

Catalog Name: The segment of the URL immediately following the base URL (in this case, “savings”) is generally the name of the catalog to which the API is deployed. This is a default setting in IBM API Connect.
API Name: The last segment of the URL (in this case, “annual”) is usually the name of the API being called.
Default Catalog Settings: With default catalog settings, the catalog name is included in the URL to distinguish between different catalogs. This helps in organizing and managing APIs across different environments or stages (e.g., development, testing, production).

Reference:

IBM API Connect documentation on API endpoint structure and catalog settings.
General principles of API management and deployment in IBM API Connect.

Question: 3

Which of the following is true for Products?

A. A Product can be published to selected communities.
B. A Product must be published before it is staged.
C. APIs become accessible when a Product is staged on the Developer Portal.
D. When a Product is staged or published, it is visible for all communities.

Answer: A

Explanation:

Product Publication: In IBM API Connect, a Product is a collection of APIs and Plans that can be published to a Developer Portal. When publishing a Product, you have the option to select specific communities to which the Product will be available.
Selected Communities: This feature allows API providers to control access to their APIs by making them available only to certain groups or communities. This is useful for managing access based on different criteria such as user roles, subscription levels, or organizational units.
Staging vs. Publishing:
Staging: When a Product is staged, it is deployed to a staging environment where it can be tested and validated before being made publicly available.
Publishing: After successful staging, the Product can be published to the Developer Portal, making it accessible to the selected communities.
Accessibility of APIs: APIs within a Product become accessible to developers when the Product is published to the Developer Portal, not just when it is staged.
Visibility: The visibility of a Product is controlled by the API provider. It can be restricted to specific communities or made available to all, depending on the publication settings.

Reference:

IBM API Connect documentation on Product management and publication.
Best practices for managing API access and visibility in IBM API Connect.

Question: 4

Which statement is true about the use of $ref?

A. Gatewayscript can use $ref instead of included files.
B. $ref can only be defined in YAML files.

C. $ref must be defined at the root level of the YAML file.
D. When an API is published, the $ref is replaced with the contents of the referenced file.

Answer: D

Explanation:

$ref Usage: The $ref keyword is used in OpenAPI (formerly Swagger) specifications to reference reusable components, such as schemas, parameters, and responses, defined elsewhere in the document or in external files.
Replacement Mechanism: When an API is published, the $ref is processed and replaced with the actual contents of the referenced file or component. This allows for modular and maintainable API definitions, where common elements can be defined once and reused across multiple APIs.
YAML and JSON: The $ref keyword can be used in both YAML and JSON files, which are common formats for defining OpenAPI specifications. It is not limited to YAML files.
Location Flexibility: $ref can be defined at various levels within the YAML or JSON file, not necessarily at the root level. It can be used within objects, arrays, and other structures as needed.
Gatewayscript: While Gatewayscript can include external files, it does not use the $ref keyword in the same way as OpenAPI specifications. Gatewayscript has its own mechanisms for including and referencing external scripts.
Reference:
IBM API Connect documentation on OpenAPI specifications and the use of $ref.
General principles of API design and modularization using OpenAPI.

Question: 5

Which statement is true regarding API Analytics Dashboards?

A. Data from multiple Catalogs cannot be visualized on a single dashboard.
B. A visualization should be created and saved after creating a custom dashboard.
C. A single dashboard may be created that includes data from multiple Catalogs.
D. All users have the ability to create custom dashboards.

Answer: C

Explanation:

API Analytics Dashboards: In IBM API Connect, analytics dashboards provide insights into API usage, performance, and trends. These dashboards are essential for monitoring and optimizing API strategies.
Multiple Catalogs: IBM API Connect allows the creation of a single dashboard that can aggregate and visualize data from multiple catalogs. This feature is particularly useful for organizations that manage APIs across different environments or stages (e.g., development, testing, production).
Visualization Creation: While visualizations are an integral part of dashboards, they can be created and saved at any time, not necessarily after creating a custom dashboard.
User Permissions: The ability to create custom dashboards may be restricted based on user roles and permissions. Not all users may have the necessary permissions to create custom dashboards.
Data Aggregation: By including data from multiple catalogs in a single dashboard, API providers can gain a comprehensive view of their API ecosystem, making it easier to identify patterns, detect anomalies, and make informed decisions.
Reference:

IBM API Connect documentation on analytics and dashboard creation.
Best practices for API management and monitoring in IBM API Connect.

Question: 6

Which statement is correct about migration of a subscribed external application to a new Product?

A. A Community Manager manages migration of a subscribed external application to a new Product.
B. The admin of a Developer Portal can enforce the migration to a new Product.
C. The Consumer manages migration of their subscribed application to a new Product.
D. The Plans of the new product should be the same or less restrictive than those of the original Product.

Answer: C

Explanation:

Consumer Responsibility: In IBM API Connect, the consumer (or developer) who has subscribed to an API Product is responsible for managing the migration of their subscribed application to a new Product. This ensures that the consumer has control over their application's dependencies and can test the new Product before fully migrating.
Migration Process: When a new Product is introduced, consumers need to subscribe to the new Product and update their application configurations to use the new API endpoints and plans. This process involves:
Reviewing the new Product's documentation and capabilities.
Testing the new Product in a staging environment.
Updating application code and configurations to point to the new Product's endpoints.
Subscribing to the new Product through the Developer Portal.
Role of Admins and Community Managers:
Admins: While the admin of a Developer Portal can facilitate the migration process by providing necessary tools and support, they do not enforce the migration.
Community Managers: They manage the overall community and can provide guidance, but the actual migration task is performed by the consumer.
Product Plans: The plans of the new Product do not necessarily have to be the same or less restrictive than those of the original Product. The new Product may offer different plans with varying levels of access and features.
Reference:
IBM API Connect documentation on managing API Products and subscriptions.
Best practices for API consumer management and migration in IBM API Connect.

Question: 7

Which statement is true about API properties?

A. API properties are pre-defined.
B. Values of API properties can be changed during runtime.
C. A property can have a Catalog specific value.
D. If Spaces are enabled, a property can have a Space specific value.

Answer: C

Explanation:

API Properties: In IBM API Connect, API properties are configuration settings that can be used to customize the behavior of APIs. These properties can be defined at various levels, including the API, Product, and Catalog levels.

Catalog-Specific Values: A property can be assigned a specific value for each Catalog. This allows for different configurations and behaviors of the same API when deployed in different Catalogs. For example, an API might have different endpoint URLs or security settings in development, testing, and production environments.

Runtime Changes: While some properties can be modified at runtime, this is not a general rule for all API properties. The ability to change property values at runtime depends on the specific property and its configuration.

Pre-defined Properties: API properties can be both pre-defined and custom-defined. Pre-defined properties are provided by IBM API Connect, while custom properties can be created by API developers to suit specific needs.

Spaces and Properties: If Spaces are enabled, properties can indeed have Space-specific values. This allows for further granularity in configuration, enabling different settings within the same Catalog based on the Space.

Reference:

IBM API Connect documentation on API properties and their configurations.

Best practices for managing API properties in IBM API Connect.

Question: 8

The security team will allow OIDC to be used for Portal access. Which two OIDC provider types are available out of the box?

A. Azure
B. Github
C. Instagram
D. Facebook
E. IBM Cloud

Answer: A E

Explanation:

In IBM API Connect v10.0.3, OpenID Connect (OIDC) is a widely used authentication mechanism that allows clients to verify the identity of users based on the authentication performed by an authorization server. API Connect supports various OIDC providers for portal access to integrate with third-party identity providers. Out of the box, API Connect v10.0.3 comes with support for Azure Active Directory (Azure AD) and IBM Cloud as pre-configured OIDC providers.

Azure AD is one of the most commonly used OIDC providers, especially in enterprise environments using Microsoft's cloud services.

IBM Cloud Identity and Access Management (IAM) also provides built-in OIDC support within the IBM ecosystem, making it a natural integration point for users of IBM Cloud services.

These built-in integrations simplify the configuration and management of OIDC providers without requiring additional custom configuration.

Reference:

IBM API Connect v10.0.3 - Now Available

Developer User Experience with API Connect

Question: 9

How can a Consumer see their API usage?

A. The calling application would need to keep track of the API usage.

B. The numbers of requests, for different APIs, that the Consumer's application has made are shown on their application page.

C. The Consumer can be permitted to access the analytics on API Manager.

D. The Consumer would need to request an API usage report from the API provider.

Answer: B

Explanation:

In IBM API Connect v10.0.3, consumers can see their API usage on their application page within the Developer Portal. This page provides details about the number of requests made by their application to different APIs, allowing consumers to monitor their API consumption directly. This feature helps consumers track their usage metrics without needing to keep track themselves or request reports from the API provider.

Other options are incorrect:

Option A: The calling application does not need to track API usage manually.

Option C: Consumers are not typically given access to the API Manager analytics directly.

Option D: API providers do not need to generate a separate report; usage details are automatically available to consumers.

Reference:

IBM API Connect v10.0.3 Documentation: Consumer Access to API Usage

IBM Community Blog: Developer User Experience with API Connect

Question: 10

How are Gateway extensions packaged to upload to the Gateways?

A. As .tar files

B. As .cfg files

C. As .zip files

D. As war files

Answer: C

Explanation:

Packaging Format: Gateway extensions in IBM API Connect v10.0.3 are packaged as .zip files. This format is used to bundle all necessary files and configurations required for the extension.

Upload Process: These .zip files are then uploaded to the Gateway through the API Manager interface. The API

Manager handles the deployment and integration of these extensions into the Gateway.
Documentation Reference: According to the IBM Certified Solution Implementer - API Connect v10.0.3 documentation, the correct packaging format for Gateway extensions is .zip files1.
1: IBM Certified Solution Implementer - API Connect v10.0.3 Documentation

Question: 11

Which HA concept applies for OAuth operations in a multi-node Kubernetes cluster?

A. Quorum
B. Heartbeat
C. STONITH
D. Quantum

Answer: A

Explanation:

High Availability (HA) Concept: In a multi-node Kubernetes cluster, the concept of “Quorum” is crucial for ensuring high availability and consistency, especially for operations like OAuth.
Quorum Definition: Quorum refers to the minimum number of nodes that must agree on a transaction or operation to ensure consistency and avoid split-brain scenarios. This is particularly important in distributed systems to maintain data integrity and availability.
OAuth Operations: For OAuth operations, maintaining a quorum ensures that the authentication and authorization processes are reliable and consistent across the cluster. This helps in preventing issues where different nodes might have conflicting states.
Documentation Reference: According to the IBM Certified Solution Implementer - API Connect v10.0.3 documentation, the concept of quorum is applied to ensure high availability and consistency in OAuth operations within a multi-node Kubernetes cluster1.
I. IBM Certified Solution Implementer - API Connect v10.0.3 Documentation

Question: 12

What is correct about using context variables in Gatewayscript policies?

A. context.set(’my.vars.amount’, 100) generates { "my.vars": { "amount": 100 } }
B. All API context variables are saved in an XML tree.
C. context.message.statuscode = ’200 Success' updates the status code of the message object.
D. context. request. statuscode = 400 updates the status code of the request object.

Answer: A

Explanation:

Setting Context Variables: In Gatewayscript policies, the context.set function is used to set context variables.
The syntax context.set('my.vars.amount', 100) creates a JSON structure where my.vars is an object containing the key amount with the value 100.

JSON Structure: This method of setting context variables generates a JSON structure, which is a common format for data interchange in APIs. The resulting structure would be { "my.vars": { "amount": 100 } }.
Usage in Policies: This approach allows for easy manipulation and access to variables within the GatewayScript, facilitating dynamic API behavior based on the context.
Documentation Reference: According to the IBM Certified Solution Implementer - API Connect v10.0.3 documentation, using context.set in this manner is the correct way to generate the specified JSON structure1.
1: IBM Certified Solution Implementer - API Connect v10.0.3 Documentation

Question: 13

What is a key requirement when creating an OpenAPI 3.0 API secured by basic authentication, API Key, or OAuth?

A. It needs an external security service.
B. The relevant information can only be passed in the header.
C. The API can only be enforced by Datapower API Gateway.
D. The security-schema-name must follow a strict pattern.

Answer: D

Explanation:

When creating an OpenAPI 3.0 API in IBM API Connect v10.0.3 that is secured by basic authentication, API Key, or OAuth, it is essential that the security-schema-name follows a specific pattern. This pattern is required to ensure proper validation and application of the security definitions according to the OpenAPI 3.0 specification. The security definitions help define the methods of authentication that are enforced for accessing the API endpoints, which is crucial for maintaining the API's integrity and security.
Reference:
IBM API Connect v10.0.3 Now Available
IBM API Connect Support Lifecycle Policy

Question: 14

Which statement is correct regarding the creation of a SOAP proxy API from an existing SOAP service?

A. The WSDL file describing the SOAP service is uploaded and the dependencies are set target services.
B. A single .zip file that contains the WSDL file describing the SOAP service, and its dependent documents is uploaded.
C. To expose a SOAP service in API Connect, it has to be mapped to a REST API.
D. In API Connect, a SOAP API can be created only from a stand-alone WSDL.

Answer: B

Explanation:

When creating a SOAP proxy API from an existing SOAP service in IBM API Connect, a key step is to upload a single .zip file that contains the WSDL (Web Services Description Language) file and any associated dependent documents. This enables IBM API Connect to understand the service definition and its dependencies, allowing

it to create the proxy API correctly. This method is necessary to ensure all components required for the SOAP API are packaged together and recognized during the import process.

Reference:

IBM API Connect v10.0.3 Documentation

Creating SOAP APIs

Question: 15

What is OpenAPI?

A. An XML-based messaging protocol for exchanging information among computers.
B. A standard, programming language-agnostic interface description.
C. A set of constraints for how the architecture of an Internet-scale distributed hypermedia system should behave.
D. A query language for APIs and a runtime.

Answer: B

Explanation:

Definition: OpenAPI is a specification for building APIs that is both language-agnostic and standardized. It allows developers to describe the structure of their APIs in a way that is easily understood and implemented across different programming languages.

Purpose: The main goal of OpenAPI is to provide a clear and consistent way to define APIs, making it easier for developers to create, share, and consume APIs. This standardization helps in reducing the complexity and potential errors in API development.

Components: OpenAPI includes various components such as paths, operations, parameters, and responses, which collectively describe the API's functionality and behavior.

Documentation Reference: According to the IBM Certified Solution Implementer - API Connect v10.0.3 documentation, OpenAPI is indeed a standard, programming language-agnostic interface description1.

1: IBM Certified Solution Implementer - API Connect v10.0.3 Documentation

Question: 16

Which options would be selected to manage the lifecycle of a version of a Product?

A. Set endpoint
B. Set migration target
C. Set burst limit
D. Set rate limit

Answer: B

Explanation:

Lifecycle Management: Managing the lifecycle of a version of a Product in IBM API Connect involves setting

various parameters to ensure smooth transitions and updates. One key option is to set the migration target.

Migration Target: This option allows administrators to specify the target environment or version to which the Product should be migrated. It ensures that the Product is correctly aligned with the desired state and environment.

Documentation Reference: According to the IBM Certified Solution Implementer - API Connect v10.0.3 documentation, setting the migration target is a crucial step in managing the lifecycle of a Product version1.

1: IBM Certified Solution Implementer - API Connect v10.0.3 Documentation

Question: 17

How can a Consumer organization's ownership be transferred?

A. In the navigation pane of the API Manager Ul on the Consumers tab, click the options icon (three dots) alongside the Consumer organization to work with, then click Transfer. Select the user that will be the new owner. Click Confirm.

B. Using the CLI, execute the apic consumer-orgs : transfer-owner command.

C. Using the REST interface, send a PATCH request to https : // {apimserver} /consumer-org/ {org} /transfer-owner.

D. It is not possible to change the Consumer organization owner once created.

Answer: A

Explanation:

Navigation Pane: To transfer the ownership of a Consumer organization, navigate to the API Manager Ul.

Consumers Tab: Click on the Consumers tab to view the list of Consumer organizations.

Options Icon: Find the Consumer organization you want to transfer ownership of and click the options icon (three dots) next to it.

Transfer Option: Select the “Transfer” option from the dropdown menu.

Select New Owner: Choose the user who will be the new owner from the list of available users.

Confirmation: Click “Confirm” to finalize the transfer of ownership.

Documentation Reference: According to the IBM Certified Solution Implementer - API Connect v10.0.3 documentation, this is the correct procedure to transfer the ownership of a Consumer organization1.

1: IBM Certified Solution Implementer - API Connect v10.0.3 Documentation

Question: 18

Which API Event Record field indicates N/A when a client ID is not used or is invalid on the API?

A. api_id

B. log_policy

C. product_name

D. org_id

Answer: C

Explanation:

In IBM API Connect v10.0.3, the product_name field in the API Event Record will indicate "N/A" if a client ID is not used or is invalid on the API. This field typically shows the name of the API product associated with the request, but if no valid client ID is present, it cannot associate the request with a product, resulting in "N/A."

Other options are incorrect:

api_id would display the API identifier.

log_policy relates to the logging policy and would not be "N/A" due to an invalid client ID.

org_id refers to the organization ID and is unrelated to client ID validity.

Reference:

IBM API Connect v10.0.3 Documentation: API Event Record Fields

Question: 19

In which two places can the rate limit for an API be defined?

A. Plan
B. Space
C. Catalog
D. API definition
E. User role

Answer: A, C

Explanation:

In IBM API Connect v10.0.3, rate limiting for APIs can be defined in two primary places: Plan and Catalog.

Plan: Rate limiting is typically defined at the Plan level. A Plan in API Connect specifies how an API or a group of APIs is exposed to consumers. It can include rate limits, quotas, and other restrictions. Defining rate limits at the plan level ensures that each application subscribed to the plan adheres to the specified API consumption limits, such as the number of API calls allowed within a given time period.

Catalog: The Catalog is another place where rate limits can be defined. A Catalog represents a collection of APIs and their configurations, including security and rate limiting policies. Rate limits defined at the Catalog level apply globally to all APIs and applications within that Catalog, providing an overarching control mechanism to enforce consumption limits across different APIs.

These two locations allow flexibility in controlling API usage, either by restricting usage per plan (at a more granular level) or globally across all APIs within a catalog.

Reference:

IBM API Connect v10.0.3 - Now Available

IBM API Connect Documentation

Question: 20

Which role is required to access the "Email Subscribers" wizard in the Developer Portal?

A. Administrator
B. Consumer
C. Provider
D. Publisher

Answer: A

Explanation:

The "Email Subscribers" wizard in the Developer Portal is a tool used by administrators to send emails to subscribers of specific products. To access this wizard, you need to have the Administrator role in the Developer Portal. This role grants you the necessary permissions to manage and interact with various aspects of the Developer Portal, including sending emails to subscribers.

Reference:
IBM API Connect: Emailing product subscribers
IBM API Connect: Developer Portal Roles

Question: 21

Why would an administrator run the apic apic-config: get Portal CLI command?

A. To view the configured user registries, payment methods, permissions, and TLS profiles for the Portal
B. To setup notifications
C. To initiate backup of the Portal
D. To restore the Portal backup

Answer: A

Explanation:

An administrator would run the apic apic-config: get Portal CLI command to view the configured user registries, payment methods, permissions, and TLS profiles for the Portal. This command provides a comprehensive overview of the Portal's configuration settings, allowing the administrator to verify and modify them as needed.

Reference:
IBM API Connect: Developer Portal CLI commands
IBM API Connect: Managing Portal Configuration

Question: 22

What are the two ways to create a new Consumer organization?

A. Invite a user to be a Consumer organization owner, specifying the title
B. Create the Consumer organization with a specific owner. The user will specify the title.
C. Invite an organization owner to create a new Consumer organization, specifying the title. All the assets will be moved to the new organziation and the old one will be deleted.
D. Create the Consumer organization, specifying the owner and title.

E. Invite a user to be a Consumer organization owner. The user will specify the title.

Answer: BD

Explanation:

In IBM API Connect v10.0.3, a Consumer organization represents a group of users who consume APIs published to a Catalog. There are two primary ways to create a new Consumer organization: Create the Consumer organization with a specific owner. The user will specify the title (B): In this method, the Consumer organization is created by an administrator, who specifies the owner. After the organization is created, the owner can then specify additional details, such as the organization's title. This approach allows the administrator to delegate the customization of the organization title to the owner.
Create the Consumer organization, specifying the owner and title (D): In this case, the administrator creating the Consumer organization specifies both the owner and the title at the time of creation. This method provides complete control over the organization's creation, allowing both the owner and the title to be set during the initial configuration.
These methods allow flexibility in how organizations are set up, either by allowing the owner to define certain details or by having everything specified during creation by the administrator. Reference:
IBM API Connect v10.0.3 - Now Available
IBM API Connect Documentation

Question: 23

Which two image formats are supported for use in the developer portal?

A. JPG
B. GIF
C. PPM
D. EXIF
E. WebP

Answer: A, B

Explanation:

The Developer Portal in API Connect supports the following image formats:
JPEG (.jpg): This is a widely used image format that is commonly used for photos and other images.
GIF (.gif): This format is often used for animated images and simple graphics.
While other formats may be technically possible to use in the Developer Portal, these two are the most widely supported and recommended formats.
Reference:
IBM API Connect: Developer Portal User Guide
IBM API Connect: Creating and Managing Products

Question: 24

Which role creates a Consumer organization, assigns a customer representative as the owner, and manages

the relationship between the Provider organization and each Consumer organization?

A. Organization Manager
B. Product Manager
C. API Manager
D. API Lifecycle Manager

Answer: A

Explanation:

An Organization Manager is responsible for creating Consumer organizations, assigning customer representatives as owners, and managing the relationship between the Provider organization and each Consumer organization. This role has the necessary permissions to create new organizations, invite users to become owners, and oversee the interactions between the Provider and Consumer organizations.

Reference:
IBM API Connect: Creating and Managing Organizations
IBM API Connect: Inviting Users to Organizations
IBM API Connect: Managing Organization Relationships

Question: 25

How can an application developer create a better app using APIs?

A. By using multiple APIs to create composite apps to get extra value.
B. By dealing with the long, arduous task as part of the partner on-boarding process to use their APIs.
C. By using as few APIs as possible to avoid complex apps, errors, and delays in development.
D. By using internal APIs to get data from a third-party company as internal data is more valuable than public data.

Answer: A

Explanation:

Application developers can create better apps using APIs by combining multiple APIs to create composite apps. This approach allows developers to leverage the functionality of different APIs to create more valuable and innovative applications. By combining APIs, developers can access a wider range of data and functionality, create more personalized experiences, and solve complex problems more effectively.

Reference:
IBM API Connect: Creating Composite APIs
IBM API Connect: Building Better Apps with APIs

Question: 26

Which component provides insight into API usage?

A. Cloud Manager
B. API Manager
C. API Analytics
D. Developer Toolkit

Answer: C

Explanation:

API Analytics provides insight into API usage. It offers a comprehensive set of tools and metrics to monitor and analyze API performance, usage patterns, and user behavior. This information is valuable for understanding API adoption, identifying trends, and making data-driven decisions to improve API management.

Reference:

IBM API Connect: API Analytics

IBM API Connect: Monitoring and Analyzing API Usage

Question: 27

What is a vanity API endpoint?

A. The server URL on which the API is running.
B. The basepath as defined in the OpenAPI definition for the API.
C. The gateway endpoint at which the API is invoked.
D. The endpoint that is visible to the Consumer in the Developer Portal.

Answer: D

Explanation:

A vanity API endpoint is the endpoint that is visible to the Consumer in the Developer Portal. It is a user-friendly and memorable endpoint that masks the underlying gateway endpoint where the API is actually invoked. Vanity endpoints provide a more convenient and intuitive way for Consumers to interact with APIs.

Reference:

IBM API Connect: Creating and Managing APIs

IBM API Connect: Vanity Endpoints

Question: 28

The GraphQL developer incorporated a field that should not be available for introspection or validation by the client.

How would the API developer obfuscate the field so clients will not view them?

A. Click the gear icon and choose Hide on the Show/Hide setting.
B. Apply the @hide directive to the GraphQL schema.
C. Add the Redaction policy to the GraphQL Assemble.
D. Choose the field in the schedule and click the delete icon.

Answer: B

Explanation:

To obfuscate a field in a GraphQL schema so that it is not available for introspection or validation by

the client, the API developer can apply the @hide directive to the field. This directive tells the GraphQL engine to exclude the field from the schema introspection and validation process, making it invisible to clients.

Reference:

IBM API Connect: GraphQL API Development

IBM API Connect: GraphQL Schema Directives

Question: 29

Which two grant types are supported for native and third-party OAuth providers?

A. Explicit
B. Application (client-credentials)
C. Inhibit
D. Exhibit
E. Implicit

Answer: B, E

Explanation:

API Connect supports two grant types for native and third-party OAuth providers:

Application (client-credentials): This grant type is used when the application itself is the client and does not require user interaction. It is suitable for machine-to-machine authentication.

Implicit: This grant type is used when the application obtains an access token directly from the authorization server without going through a redirect URI. It is suitable for web applications that are embedded in other applications.

Reference:

IBM API Connect: Configuring a native OAuth provider

IBM API Connect: Configuring a third-party OAuth provider

OAuth 2.0 Authorization Framework: https://datatracker.ietf.org/doc/html/rfc6749

Question: 30

How many languages are available to send and personalize notification emails with?

A. The notifications can be sent in 15 languages. Each notification email can be personalized for each language separately.
B. The notifications are sent in English only. Each notification email can be personalized.
C. The notifications can be sent in 15 languages. Each notification email can be personalized in English only.
D. The notifications can be sent in 15 languages. Each notification email can be personalized with the same template for all the languages.

Answer: A

Explanation:

API Connect allows you to send and personalize notification emails in 15 languages. You can create separate templates for each language, allowing you to tailor the content to the specific needs of your audience. This

provides a more personalized and engaging experience for your users.

Reference:

IBM API Connect: Configuring Notifications

IBM API Connect: Customizing Notification Templates

Question: 31

A developer would like to run the step debugger on the Gatewayscript policy that is part of the Assembly.
What would allow step debugging on the Gatewayscript?

A. Add a debugger; statement.
B. Enable step debug in the Test tab.
C. Click the Trace in the Test tab.
D. Enable debug checkbox in Assemble.

Answer: A

Explanation:

To enable step debugging on a GatewayScript policy in IBM API Connect v10.0.3, a developer must add a debugger; statement within the GatewayScript code. This statement serves as a breakpoint that will pause execution, allowing the developer to step through the code to identify and resolve issues.

Other options are incorrect:

Option B and D ("Enable step debug in the Test tab" and "Enable debug checkbox in Assemble") do not enable the step debugger in GatewayScript.

Option C ("Click the Trace in the Test tab") allows tracing but not step debugging.

Reference:

IBM API Connect v10.0.3 Documentation: Debugging GatewayScript Policies

Question: 32

A User registry was created in the Cloud Manager.
In order for the registry to be used on the Developer Portal, what must be completed?

A. On the appropriate Catalog, configure the Onboarding setting.
B. On the API Manager resources page, make the User registry public.
C. On the Portal settings page, choose the User registry.
D. On the appropriate Catalog, configure the resources settings.

Answer: A

Explanation:

After creating a User registry in the Cloud Manager, it must be configured on the appropriate Catalog's Onboarding settings to be used on the Developer Portal in IBM API Connect v10.0.3. This configuration allows the User registry to authenticate and manage users effectively when they interact with the Developer Portal.

Other options are incorrect:

Option B: Making the User registry public on the API Manager resources page does not apply. Option C: Configuring the Portal settings page is not sufficient for enabling the registry.

Option D: Configuring resources settings on the Catalog does not directly involve user registry onboarding.

Question: 33

After a developer selects Secure using Client ID, what header is required when executing the API?

A. client-identification
B. Authorization code
C. Auth bearer token
D. X-IBM-Client-Id

Answer: D

Explanation:

When a developer selects "Secure using Client ID" in API Connect, the required header for executing the API is X-IBM-Client-Id. This header is used to authenticate the client and ensure that it has the necessary permissions to access the API.

Reference:

IBM API Connect: Securing APIs with Client IDs

IBM API Connect: API Security

Question: 34

What occurs if the Catalog associated with the Developer Portal has two gateway services enabled, each of which is associated with different analytics services?

A. The Developer Portal displays separate charts for each analytics service.
B. The Developer Portal does not display analytics data.
C. The Developer Portal only displays Total Calls for the applications.
D. The Developer Portal displays one chart which combines the API stats.

Answer: B

Explanation:

In IBM API Connect v10.0.3, if a Catalog associated with the Developer Portal has two gateway services enabled, and each of these gateway services is associated with different analytics services, the Developer Portal will not display any analytics data. This is because IBM API Connect does not support combining or displaying analytics data from multiple analytics services in a single portal view.

Each gateway service would be linked to its own analytics service, and since the Developer Portal is designed to integrate with a single analytics service per Catalog, conflicting analytics configurations prevent the display of any analytics data. As a result, the system does not display any statistics like API calls, response times, or error rates in the Developer Portal when multiple analytics services are configured.

Reference:

IBM API Connect Documentation

IBM API Connect v10.0.3 - Now Available

Question: 35

A developer is working on an API which supports the v5c-gateway and wants to also support the datapower-api-gateway.
Which statement is true about selecting a gateway?

A. Only one gateway type is allowed per API.
B. Disabling enforce will allow multiple gateways.
C. Selecting multiple gateways is accomplished in the API Manager.
D. No changes are needed to switch gateways.

Answer: A

Explanation:

In IBM API Connect, each API is associated with a specific gateway type, and only one gateway type is allowed per API. The v5c-gateway and datapower-api-gateway are different gateway types, and a single API cannot simultaneously support both types. If a developer wants to support multiple gateways, they must create separate APIs for each gateway type.

Reference:

IBM API Connect v10.0.3 Documentation

Question: 36

Which action will reload all the default notifications and overwrite any customizations of notification templates?

A. Updating the curly brace template_name variable in the handlebars syntax
B. Disabling and re-enabling template customization toggle
C. Transferring ownership of the customized template to another user
D. Renaming the customized notification template

Answer: B

Explanation:
Disabling and re-enabling the template customization toggle in API Connect will reload all the default notifications and overwrite any customizations of notification templates. This action effectively resets the templates to their original state, restoring the default settings and removing any modifications that were made.

Reference:

IBM API Connect: Configuring Notifications
IBM API Connect: Customizing Notification Templates

Question: 37

Which component enforces runtime policies to secure and control API traffic?

A. Cloud Manager
B. API Manager
C. Developer Portal
D. API Gateway

Answer: D

Explanation:

The API Gateway is responsible for enforcing runtime policies to secure and control API traffic. It acts as a central point of control for API requests, intercepting and analyzing them to apply security measures, rate limits, quotas, and other policies. This ensures that APIs are accessed and used in a secure and controlled manner.

Reference:

IBM API Connect: API Gateway
IBM API Connect: Securing APIs with the API Gateway
IBM API Connect: Managing API Traffic with the API Gateway

Question: 38

Using the API Manager Ul, which two pieces of information can be viewed from the subscriptions in a Product?

A. The Plan that the application is subscribed to
B. The Product that contains the Plan that the application is subscribed to
C. The spaces where the subscribing Product is published
D. The subscribing APIs
E. The Catalogs where the subscribing Product is published

Answer: A, B

Explanation:

Using the API Manager UI, you can view the following two pieces of information from the subscriptions in a Product:

The Plan that the application is subscribed to: This information tells you which specific plan the application is using to access the APIs in the Product.

The Product that contains the Plan that the application is subscribed to: This information shows you the overall Product that the application is subscribed to, which includes the Plan and associated APIs. While you cannot directly view the spaces or catalogs where the subscribing Product is published from the subscriptions list, you can find this information by navigating to the Product itself and examining its properties.

Reference:

IBM API Connect: Managing Subscriptions
IBM API Connect: Managing Products
IBM API Connect: Managing Spaces
IBM API Connect: Managing Catalogs

Question: 39

Which statement describes a requirement for monetizing a Product?

A. An administrator must have both soft and hard rates limits set for the API product.
B. A billing integration resource must first be configured for the Provider organization.
C. Any existing free plans for the Product must be disabled.
D. The Payment Cloud method module must be enabled in the Developer Portal.

Answer: B

Explanation:

To monetize a Product in API Connect, you must first configure a billing integration resource for the Provider organization. This resource defines the payment gateway or other payment method that will be used to process payments for API subscriptions. Once a billing integration resource is configured, you can create paid plans for your Products and enable billing for those plans.

Reference:

IBM API Connect: Monetizing APIs

IBM API Connect: Configuring Billing Integration Resources

Question: 40

Which statement is true when adding extra DataPower enforcement capabilities to a Gateway service?

A. The DataPower extensibility function does not allow encryption and decryption.
B. Multiple Gateway extension .zip files may be added to the same Gateway service.
C. The network route to a physical DataPower device is configured through the API Manager.
D. More than one Gateway extension .zip file cannot be uploaded to the same Gateway service.

Answer: B

Explanation:

When adding extra DataPower enforcement capabilities to a Gateway service in API Connect, you can upload multiple Gateway extension .zip files to the same Gateway service. This allows you to customize the Gateway's behavior and add specific enforcement capabilities as needed. Reference:

IBM API Connect: Configuring DataPower Enforcement Capabilities

IBM API Connect: Gateway Extensions

Question: 41

How can a user be added that was previously removed from a Provider organization that uses a Local User Registry (LUR)?

A. The user must re-activate their account by using the Sign Up option, not by using the Sign In option.
B. A deleted user from a LUR cannot be added to the same registry.

C. The user must re-register using the invitation link sent by the Provider organization owner or a user with the appropriate role.
D. The user must re-activate their account by using the Sign In option, not by using the Sign Up option.

Answer: C

Explanation:

To add a user that was previously removed from a Provider organization that uses a Local User Registry (LUR), the user must re-register using the invitation link sent by the Provider organization owner or a user with the appropriate role. This ensures that the user is added back to the organization with the correct permissions and settings.

Reference:
IBM API Connect: Managing Users in Local User Registries
IBM API Connect: Inviting Users to Organizations

Question: 42

Who creates applications that use the APIs available in the Developer Portal?

A. API Developer
B. API Lifecycle Manager
C. A member of the Consumer organization
D. A member of the Provider organization

Answer: C

Explanation:

A member of the Consumer organization creates applications that use the APIs available in the Developer Portal. Consumer organizations represent the entities that will be consuming the APIs, and their members are responsible for developing and managing applications that utilize those APIs. Reference:
IBM API Connect: Developer Portal Roles
IBM API Connect: Creating and Managing Applications

Question: 43

What can be done to make sure all errors are caught within an assembly?

A. Create an invoke that collects information on all causes from a service.
B. Add a default catch.
C. The errors provided cover all possible causes, and nothing needs to be done.
D. Tweak the assembly with logical policies to ensure all possible cases are covered.

Answer: B

Explanation:

To ensure that all errors are caught within an assembly in API Connect, you can add a default catch policy to

the assembly. This catch policy will capture any errors that are not handled by other policies in the assembly, providing a centralized location for error handling and logging.

Reference:

IBM API Connect: Designing and Building Assemblies

IBM API Connect: Handling Errors in Assemblies

Question: 44

Which statement is true if the admin login has been disabled on the Developer portal?

A. Registered users will receive email with link to login as admin.
B. Admin login is still accessible by navigating to <siteurl>/user/login?registry_url=/admin.
C. All users have admin authority when they login.
D. Published APIs are hidden from all users.

Answer: B

Explanation:

If the admin login has been disabled on the Developer Portal, you can still access the admin login page by navigating to <siteurl>/user/login?registry_url=/admin. This URL bypasses the normal login process and allows you to directly access the admin login screen.

Reference:

IBM API Connect: Developer Portal Security

IBM API Connect: Managing Developer Portal Settings

Question: 45

Which statement is correct about API Manager?

A. Manages the data retention of API event data
B. Manages the connection to the user registries that validates users of Provider and Consumer organizations
C. Configure analytics data offloading
D. Invokes the request to the backend service(s) defined in an API implementation

Answer: B

Explanation:

The API Manager in IBM API Connect v10.0.3 manages the connection to the user registries that validate users of Provider and Consumer organizations. This functionality ensures that the appropriate user authentication and authorization mechanisms are in place, allowing users to access various services and resources in the API Management environment.

Other options are incorrect:

Option A: Data retention of API event data is handled by API Analytics.

Option C: Configuring analytics data offloading is related to the API Analytics component.

Option D: Invoking requests to the backend services is the role of the API Gateway, not the API Manager.

Reference:

IBM API Connect v10.0.3 Documentation: API Manager Overview

Question: 46

Which of these actions is allowed?

A. Linking the same API and version inside different Products
B. Linking the same Product and version inside different APIs
C. Reusing the same Plan inside different APIs
D. Linking the same Product and version several times on the same Space

Answer: A

Explanation:

In IBM API Connect v10.0.3, it is possible to link the same API and version inside different Products. This flexibility allows an API to be offered under multiple Products, each potentially having different plans, pricing, or rate limits, depending on the business needs.

Other options are incorrect:

Option B: Linking the same Product and version inside different APIs does not apply.

Option C: Plans are linked to Products, not directly to APIs.

Option D: Linking the same Product and version multiple times in the same space is not allowed.

Reference:

IBM API Connect v10.0.3 Documentation: Managing APIs and Products

Question: 47

A developer would like to clean up old Products on the development environment. Which CLI command parameters can be used to find all Products?

A. catalog:get-products
B. products : list-all —scope catalog
C. products:list —realm [providerOrg]
D. products:list --showall

Answer: C

Explanation:

To find all Products in a specific Provider organization using the API Connect CLI, you can use the following command:

products:list --realm [providerOrg]

This command will list all Products that belong to the specified Provider organization. You can then use additional filtering options to find Products based on specific criteria, such as their creation date or status.

Reference:

IBM API Connect: API Connect CLI Reference

Question: 48

A developer has asked to modify the default global behavior of ratelimit enforcement to allow execution of the API even if the ratelimit is exceeded.
When creating the global policy yaml file which is true?

A. Add to the info section "full-custom: true".
B. At the beginning of the YAML add global-reflow-policy: 1.0.0.
C. At the beginning of the YAML add policy: 1.0.0.
D. Ensure the version at the beginning of the YAML is the same as the version in the info section.

Answer: D

Explanation:

When creating a global policy YAML file to modify the default behavior of rate limit enforcement in IBM API Connect v10.0.3, it is crucial to ensure that the version specified at the beginning of the YAML file matches the version in the info section. This alignment is necessary for the system to recognize and apply the correct policy settings across all configurations.
Other options are incorrect:
Option A is not a valid configuration setting for global policies.
Option B and C do not represent the correct approach to configuring global policy YAML files. Reference:
IBM API Connect v10.0.3 Documentation: Creating and Modifying Global Policies

Question: 49

An app developer has registered an app and has received a Client ID and secret.
Where can the developer request an additional Client ID and secret?

A. Use the Portal admin Ul to enable the multiple Client ID option.
B. An additional Client ID and secret can be added on the Subscriptions tab.
C. Use the app alias link on the existing app page.
D. Only one Client ID and secret is allowed per app.

Answer: D

Explanation:

In IBM API Connect, an app is typically associated with a single Client ID and secret to maintain a unique identity and secure access to APIs. Each app is expected to use this unique Client ID and secret for authentication and authorization purposes. Therefore, only one Client ID and secret pair is allowed per app, and additional Client IDs and secrets cannot be requested or generated for the same app.

Question: 50

Which set of APIs should be used to register users in the Developer Portal, create applications, and subscribe to APIs?

A. Management APIs
B. Consumer APIs
C. Subscription APIs
D. Portal Admin APIs

Answer: B

Explanation:

The Consumer APIs should be used to register users in the Developer Portal, create applications, and subscribe to APIs. These APIs provide the necessary endpoints and functionality for Consumers to interact with the Developer Portal and manage their API usage.

Reference:
IBM API Connect: Consumer APIs
IBM API Connect: Using the Consumer APIs

Question: 51

Which statement is correct about superseding one Product with another?

A. The Product to be superseded must be in the Staged, Retired, or Deprecated state.
B. The Product that was superseded is in the Retired state.
C. Existing customers of the Product that was superseded are automatically migrated to the superseding product.
D. If the access to the superseding Product is more restrictive than the Product to be superseded, the supersede operation fails.

Answer: D

Explanation:

When one Product is superseded by another in IBM API Connect, it is essential that the new (superseding) Product does not have more restrictive access controls than the original Product. If the superseding Product has more restrictive access policies, the supersede operation will fail because it could potentially disrupt access for existing customers or violate their expectations and agreements. Reference:
IBM API Connect Product Superseding Documentation

Question: 52

For the policy JSON to XML to work, what needs to be followed for the Datapower API Gateway?

A. Nothing needs to be done the policy can directly follow the invoke policy.
B. Input for the policy needs to be parsed data.

C. The service to perform the transformation needs to be configured.
D. The policy needs to be configured with the corresponding schemas to perform the transformation.

Answer: D

Explanation:

For the JSON to XML policy to work on the DataPower API Gateway, you need to configure the policy with the corresponding schemas to perform the transformation. The schemas define the structure and data types of the JSON and XML formats, allowing the policy to accurately convert between the two.

Reference:
IBM API Connect: DataPower API Gateway Policies
IBM API Connect: JSON to XML Policy

Question: 53

What is the effect of enabled Spaces for the management of Consumer (applications, subscriptions, etc.)?

A. There is no visible change.
B. Only analytics is specific per Space.
C. Subscription approvals and analytics are specific per Space.
D. Consumers, applications, subscriptions approvals, and analytics are now specific per Space.

Answer: D

Explanation:

When Spaces are enabled in API Connect, Consumers, applications, subscriptions approvals, and analytics are now specific per Space. This means that each Space becomes an isolated environment for managing and controlling API usage within that specific context.

Reference:
IBM API Connect: Managing Spaces
IBM API Connect: Understanding Spaces

Question: 54

The DevOps team would like to incorporate API unit testing into the build and deploy step. What could the API Connect Test application create to allow unit testing of their APIs?

A. API Hooks
B. API JUnit snippets
C. Mock tests
D. DataPower loopbacks

Answer: C

Explanation:

The API Connect Test application can create mock tests to allow unit testing of APIs. Mock tests simulate the behavior of real APIs, allowing developers to test their code in isolation without relying on external

dependencies. This can help to improve the quality and reliability of APIs.

Reference:

IBM API Connect: API Connect Test Application

IBM API Connect: Unit Testing APIs

Question: 55

Which two statements about the following code snippet are true?

```
assembly-setvar udp-basic_1.0.0_set-variable_0 reset title
  "set-variable"
  correlation-path "$.x-ibm-configuration.assembly.execute[0]*
  variable
    action set name "paraml" type string value
"$(local.parameter.credential)" exit
exit
```

A. The value to the param1 variable will be provided by the application calling an API with the policy.

B. It updates the pre-defined 'set-variable' policy.

C. The action type can be 'append'.

D. The policy sets a variable called param1.

E. The value to the param1 variable will be provided by the API developer.

Answer: AD

Explanation:

The given code snippet represents a configuration for the "set-variable" policy in IBM API Connect. This policy is used to define and set variables dynamically within the API assembly flow.

Statement D is true because the code explicitly sets a variable named param1 using the "setvariable" policy.

Statement A is also true as the value assigned to param1 is derived from ${local.parameter.credential}, which indicates that the value is dynamically provided by the application calling the API. The placeholder ${local.parameter.credential} implies that the credential parameter is provided by the calling application.

Reference:

IBM API Connect Assembly Policies Documentation

Question: 56

DRAG DROP

Select all that apply

Given an API that executes on an API Gateway service with pre-request, post-request, and error global policies, what is the order that the different assemblies will be executed if the process fails while executing the post-request?

Unordered Options

- Post-request global policy
- API assembly
- Error global policy
- Pre-request global policy

Answer:

Explanation:

In IBM API Connect, the sequence of execution for global policies and API assembly is crucial, especially in cases where the process fails. If the process fails while executing the post-request, the execution order is as follows:

Pre-request global policy: This is the first to execute before the API request is processed.

API assembly: After the pre-request global policy, the API assembly (which contains the core business logic of the API) is executed.

Post-request global policy: This is executed after the API assembly has been processed, but if the process fails here, the post-request global policy may not complete.

Error global policy: When the failure occurs, the error global policy is triggered to handle any errors that occur during the execution of the API, specifically after the failure in the post-request.

Thus, if a failure occurs in the post-request global policy, the subsequent step would involve invoking the Error global policy to handle the failure. The execution order is as follows: Pre-request global policy

API assembly

Post-request global policy (failure occurs here)

Error global policy

This is the correct flow based on API execution steps in the presence of global policies.

Question: 57

DRAG DROP

Select all that apply

A Catalog holds some published API Products before enabling Spaces.

What are the steps that need to be done for spaces to work?

Unordered Options

- Recreate application subscriptions
- Enable Spaces
- Remove all published Products.
- Retire published Products
- Republish Products.

Answer:

Explanation:

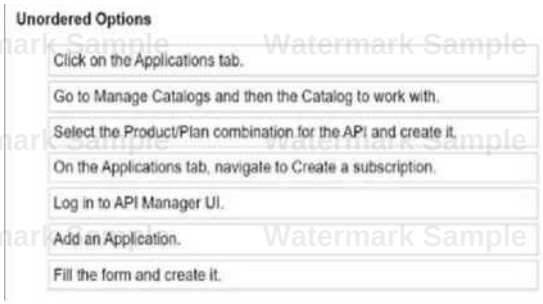
In IBM API Connect, when enabling Spaces in a Catalog that already holds published API Products, there are

specific steps that need to be followed to ensure proper transition and functionality.
Here are the necessary steps:
Retire published Products: First, you need to retire the existing published products. This is necessary because spaces require a different organization of products, and retiring the current products prevents conflicts.
Remove all published Products: Once the products are retired, they need to be removed from the Catalog before you enable spaces. This ensures that no previously published products interfere with the spaces configuration.
Enable Spaces: After retiring and removing the published products, you can proceed to enable Spaces within the Catalog. Spaces allow for more granular organization within a Catalog.
Republish Products: Once Spaces are enabled, you can republish the API products within the correct spaces. This step ensures that the products are organized within the spaces structure in the Catalog. Recreate application subscriptions: After republishing the products, you will need to recreate any application subscriptions to ensure that applications are correctly subscribed to the republished products within their respective spaces.
Thus, the correct steps for enabling spaces in a Catalog that holds published API products are: Retire published Products.
Remove all published Products.
Enable Spaces.
Republish Products.
Recreate application subscriptions.
These steps ensure that the Catalog and its associated products are restructured correctly after enabling Spaces.

Question: 58

DRAG DROP

Select all that apply
What is the correct order of these activities to create and then subscribe an application to a Product?



Answer:

Explanation:
To create and then subscribe an application to a product in IBM API Connect, the correct order of activities would be as follows:
Log in to API Manager UI: The first step is to log into the API Manager user interface, where the APIs and products are managed.
Go to Manage Catalogs and then the Catalog to work with: After logging in, navigate to the specific Catalog where you want to manage the products and subscriptions.
Click on the Applications tab: Once in the Catalog, you need to navigate to the Applications tab, which

allows you to manage applications.

Add an Application: In the Applications tab, you can add a new application that will be subscribed to an API product.

Fill the form and create it: After selecting to add an application, you must complete the required form and create the application.

On the Applications tab, navigate to Create a subscription: With the application created, navigate to the area where you can create a subscription for this application.

Select the Product/Plan combination for the API and create it: Finally, select the appropriate product and plan combination to which the application will subscribe and complete the subscription process. In summary, the correct order is:

Log in to API Manager UI.

Go to Manage Catalogs and then the Catalog to work with.

Click on the Applications tab.

Add an Application.

Fill the form and create it.

On the Applications tab, navigate to Create a subscription.

Select the Product/Plan combination for the API and create it.

Question: 59

DRAG DROP

Select all that apply

In what order do the following steps need to be performed, to enable CORS and restrict the pages from where the API can be called in an OpenAPI 3.0 definition?

Unordered Options

- **Select the Gateway tab and expand settings section**
- Open the requited API lot editing
- **Add policy**
- **Select and enable CORS**
- **Add allowed origins**

Answer:

Explanation:

To enable CORS (Cross-Origin Resource Sharing) and restrict the pages from where the API can be called in an OpenAPI 3.0 definition in IBM API Connect, the following steps should be performed in the correct order:

Open the required API for editing: Start by opening the API definition that you want to modify to enable CORS.

Select the Gateway tab and expand the settings section: Navigate to the Gateway tab and expand the settings section to access the relevant configurations.

Add policy: In the API assembly, add a new policy. This will allow you to add specific configurations for CORS.

Select and enable CORS: From the available policies, select the CORS policy and enable it. This step ensures that CORS headers are added to the API responses.

Add allowed origins: Finally, specify the allowed origins that are permitted to make requests to the API. This restricts access to certain domains or pages based on your configuration.

In summary, the correct order is:

Open the required API for editing.

Select the Gateway tab and expand the settings section.

Add policy.

Select and enable CORS.

Add allowed origins.

Question: 60

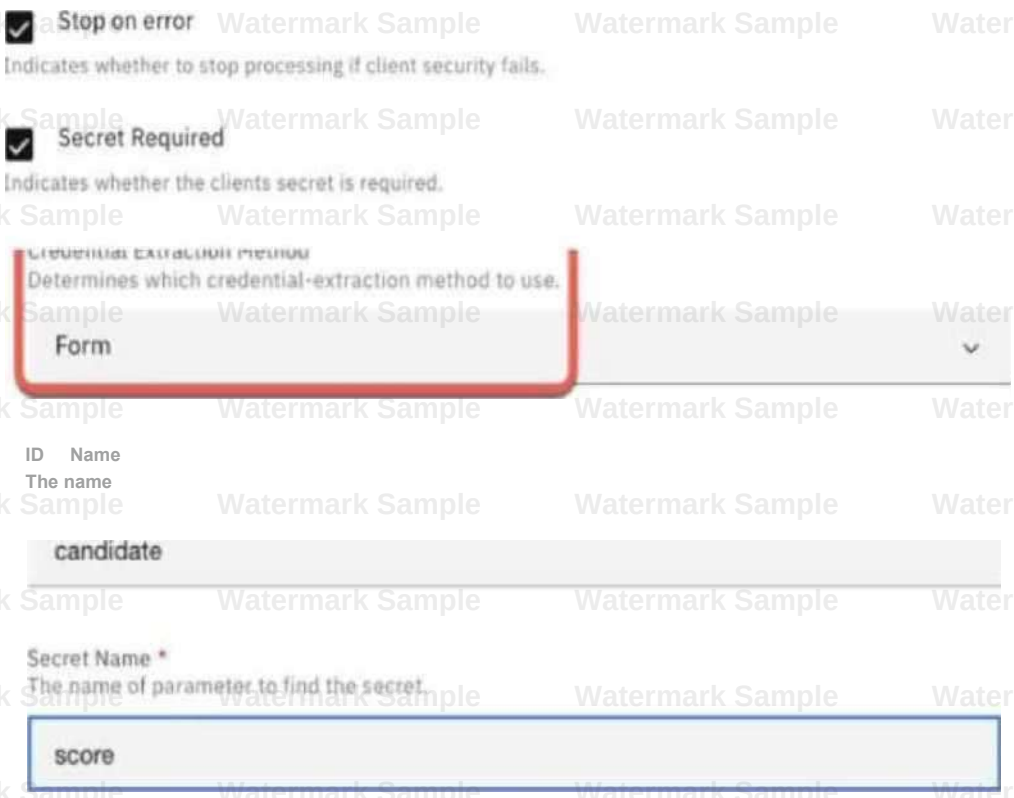
Within the client security policy, the credential extraction method is set to Form.

Which statement is a requirement in this case?

Title

client-security

Description



Authenticate Client Method

Native

A. "Stop on error" has to be active as well.

B. Credentials have to be gathered via a custom form.

C. A prefix will be added to the secret-name and id at runtime.

D. Client id and secret must be supplied via a POST request.

Answer: D

Explanation:

In IBM API Connect, when the credential extraction method is set to Form in the client security policy, it means that the client credentials (such as the client ID and secret) must be supplied in a specific format using a POST request. Specifically, the credentials are included in the body of the request in a URL-encoded form. This is typical in OAuth 2.0 workflows where client credentials need to be extracted from a form-based submission, often used in situations involving login or token exchanges.

"Client id" and "secret" are provided in the request body through form fields, rather than being passed in the URL or headers.

This method adheres to secure practices where sensitive credentials are sent via POST to avoid exposing them in the URL.

Therefore, the correct statement is that client ID and secret must be supplied via a POST request when using the Form credential extraction method.

Reference:

IBM API Connect Documentation

IBM API Connect v10.0.3 - Now Available