# Vanitha Rajendran

**1. Deleting Personal Data**

* **Hint:** To permanently delete personal data, use the dataErasure=true flag in a DELETE request. This ensures compliance with GDPR. Refer to the Data Erasure documentation.

**2. API Method Asynchronous Processing**

* **Hint:** The ApiMethod.execute() call in the commercetools Java SDK returns a java.util.concurrent.CompletableFuture for asynchronous processing. Review the Java SDK documentation for examples.

**3. Cart Discounts with Line Item Conditions**

* **Hint:** To apply discounts based on the number of line items, use lineItemCount(1 = 1) >= 5. Study the Cart Discounts documentation for predicate examples.

**4. Query Parameters Affecting Facets**

* **Hint:** Parameters like filter.query and filter.facets affect facet results. Review the Search and Facets documentation for details.

**5. Product Discounts with Predicates**

* **Hint:** Product Discounts can target specific variants or prices using predicates. Refer to the Product Discounts documentation.

**6. Adding New Languages**

* **Hint:** New languages can be added via the Merchant Center under Project Settings or using the API. Study the Project Settings documentation.

**7. HTTP Requests in Java SDK**

* **Hint:** The commercetools Java SDK supports both blocking and non-blocking HTTP requests. Learn more in the Java SDK guide.

**8. Adding a Billing Address to an Order**

* **Hint:** Use the OrderUpdateBuilder with the OrderSetBillingAddressActionBuilder to add a billing address to an order. Refer to Order Update Actions documentation.

**9. External Tax Mode**

* **Hint:** Use ExternalAmount for tax rates and calculations managed externally. Refer to the TaxMode documentation.

**10. Duplicate Emails in Stores**

* **Hint:** Customers can register with the same email address if they belong to different stores. Review the Customers and Stores documentation.

**11. Retrieving Shipping Methods**

* **Hint:** Use the Shipping Methods endpoint to retrieve all methods matching a cart. Review the Shipping Methods documentation.

**12. Orders and Order Edits**

* **Hint:** New line items can be added through Order Edits but not directly to Orders. Understand the Order Edits documentation.

# Shrikant Sonawane

**1. Product Projection Consistency**

* **Hint:** Changes in product projections after an update action are **eventually consistent**, not strongly consistent. Review the Product Projection documentation.

**2. API Extensions - Multiple Triggers**

* **Hint:** An API Extension can have multiple trigger actions for different events. Refer to the API Extensions documentation.

**3. TaxMode for External Calculations**

* **Hint:** Use ExternalAmount when tax rates and calculations are handled outside commercetools. See the TaxMode documentation.

**4. Subscriptions vs. API Extensions**

* **Hint:** Subscriptions execute behavior **after** an API call, while API Extensions execute behavior **before** the API call is completed. Learn more in the Subscriptions and API Extensions documentation.

**5. Query for a Customer by Email**

* **Hint:** Use apiRoot.customers().get().withWhere("email=\"email@example.com\"") to query customers by email in the Java SDK. Review the Customers documentation.

**6. Query Parameters Affecting Facets**

* **Hint:** Parameters like filter.query and filter.facets affect facet results. See the Search and Facets documentation.

**7. Multiple Shipping Addresses and Methods**

* **Hint:** Commercetools supports multiple shipping addresses and methods per order. Refer to the Shipping documentation.

**8. Payment Transactions**

* **Hint:** Calculate the effective amount collected by summing successful charges minus refunds. In this case, 40 GBP was effectively collected. Review Payment Transactions.

**9. Inventory Mode**

* **Hint:** Use ReserveOnOrder for limited supply items that cannot be restocked. Learn about Inventory Mode.

**10. Category Resource**

* **Hint:** Categories must have a parent, but they can also have child and ancestor information. See the Categories documentation.

**11. Current Product Representation**

* **Hint:** Use Product Projections for the "current" representation of a product. Review the Product Projections documentation.

**12. ISO 3166 State Names in Zones**

* **Hint:** State names in Zones do not need to comply with ISO 3166 standards. Review the Zones documentation.

**13. Creating a Category**

* **Hint:** Use CategoryDraftBuilder.of().name(localizedStringName).slug(localizedStringSlug).build() to create a category. Refer to Category Creation.

**14. API Limits**

* **Hint:** API limits are documented in the API Reference Documentation. See API Limits.

**15. Cart Discounts for Gifts**

* **Hint:** The quantity of a gift line item cannot be changed for a cart discount. Review Cart Discounts.

**16. Cart Freezing**

* **Hint:** Cart and Product Discounts that expire after freezing remain applied. Learn more in the Cart Freezing documentation.

**17. Cart Version Updates**

* **Hint:** After a successful update, the cart version number increases to >= current version + 1. Review Version Updates.

**18. Discount Codes with Frozen Carts**

* **Hint:** Discount Codes that no longer match predicates are retained in the cart with DiscountCodeState.DoesNotMatchCart. Refer to the Discount Code documentation.

**19. Blocking vs. Non-Blocking Calls**

* **Hint:** Non-blocking calls are recommended for better performance in the Java SDK. Learn more in the Java SDK guide.

# Reshma MD

**1. Category Resource Information**

* **Hint:** A Category resource contains information about its **parent and all ancestors**, not just the parent. Review the Categories documentation for details.

**2. OAuth 2.0 Scopes**

* **Hint:** Scopes are defined per project, and manage\_project grants full API access. Refer to the OAuth 2.0 documentation.

**3. Variant Prices**

* **Hint:** In a product response, the prices field in a variant contains **Embedded Prices only**. See Product Variant documentation.

**4. Cart Discounts**

* **Hint:** Use Fixed value and Target rules to set the price of a line item to a constant value. Study Cart Discounts documentation.

**5. Deleting Attributes**

* **Hint:** When an attribute is deleted from a product type, it is removed from all product variants. Learn more in the Product Type documentation.

**6. Standalone Prices**

* **Hint:** Standalone prices are associated with products via the SKU field. Review Standalone Prices documentation.

**7. Custom Applications**

* **Hint:** Use Custom Applications to add functionality (e.g., viewing carts) in the Merchant Center. Refer to Custom Applications documentation.

**8. Best Practices for Categories**

* **Hint:** Keep front-end navigation structures in mind when creating categories. Avoid overly nested category trees. See Category Best Practices.

**9. Discounted Quantity in Cart**

* **Hint:** With a "buy 3, get 1 free" discount, a cart with 10 items will have 3 free items. Review Discounted Quantity rules.

**10. Pricing in Variants**

* **Hint:** Pricing information is stored as an **Array of Objects** in product variants. Refer to Product Variants documentation.

**11. Deleting Product Types**

* **Hint:** A product type can only be deleted if all associated products are deleted first. See Product Type Deletion.

**12. "In Production" Flag**

* **Hint:** Projects with the "In production" flag cannot be deleted. Review the Project Settings documentation.

**13. ISO 3166 Compliance**

* **Hint:** State names in Zones do not need to comply with ISO 3166 standards. Learn about Zones and Locations.

**14. Beta Features**

* **Hint:** A "beta" tag indicates the feature is not yet under the API contract guaranteeing backward compatibility. Refer to Beta Features documentation.

**15. Update Actions for Carts**

* **Hint:** Actions like addItemShippingAddress, addShippingMethod, and setLineItemShippingDetails are supported for multiple shipping scenarios. Review Multi-shipping documentation.

**16. SLA - Planned Servicing Period**

* **Hint:** Availability calculations exclude unavailability during Planned Servicing Periods. Learn about SLA and Planned Servicing.

**17. Retry Middleware**

* **Hint:** Use RetryRequestMiddleware in the Java SDK to handle gateway timeouts and version conflicts. Refer to the Java SDK documentation.

**18. Custom Line Items**

* **Hint:** Use customLineItems to add items like insurance or transportation costs to a cart. Review Custom Line Items documentation.

**19. Custom Object Reference**

* **Hint:** Use the key-value-document type to refer to custom objects. See Custom Objects documentation.

**20. Expanding ProductType in Product Query**

* **Hint:** Use .withExpand("productType") to expand the ProductType in a product query. Learn more in the Reference Expansion documentation.

# [Priyadarshini Athinamilgai](https://learn.commercetools.com/user/view.php?id=23018&course=90)

**1. Discount Codes and Frozen Carts**

* **Hint:** When a cart is frozen, **invalid Discount Codes** remain in the cart with the state DiscountCodeState.DoesNotMatchCart. Refer to Discount Code documentation.

**2. Deleting Attributes**

* **Hint:** Deleting an attribute from a product type will remove it from all associated product variants. Study the Product Type documentation.

**3. Mark Matching Variants**

* **Hint:** Enabling markMatchingVariants may **slow down the request**, especially for large datasets. Learn more about Search Parameters.

**4. Resource Fields vs. Draft**

* **Hint:** The number of fields in a resource is **greater than** its draft due to additional metadata or read-only fields. Review the Draft Resources.

**5. Attribute Type: Set of Set**

* **Hint:** Use Set of Set of Text for nested group-like attributes. Refer to the Attribute Definitions documentation.

**6. Custom Object Reference**

* **Hint:** Use key-value-document as the reference type to link attributes to custom objects. See Custom Objects documentation.

**7. "In Production" Project Flag**

* **Hint:** Projects flagged as "In production" **cannot be deleted**. This ensures stability for live projects. Review Project Settings documentation.

**8. Combination Unique Attribute**

* **Hint:** Variants with identical values for attributes marked as **combination unique** (e.g., color: red, size: small) will cause an error. Refer to Product Variant documentation.

**9. Inventory Mode**

* **Hint:** Use ReserveOnOrder for items with limited supply that cannot be restocked. Learn about Inventory documentation.

**10. Java SDK Updates**

* **Hint:** Retrieve the latest version of the Java SDK from **Maven Central** for stable and updated releases. See Java SDK documentation.

**11. Product Variant Pricing**

* **Hint:** Pricing information is stored as an **Array of Objects** in product variants. Study the Product Variants documentation.

**12. Published Product Data Retrieval**

* **Hint:** Avoid using Product query for published product data. Instead, use Product projection search or Product projection query. Refer to Product Projections documentation.

**13. Custom Line Items in Cart**

* **Hint:** Custom Line Items' prices and taxes are **not handled by price selection** like Line Items. See Custom Line Items documentation.

# Prateek Arora

**1. Prices Field in Product Variant**

* **Hint:** The prices field in a product variant contains **Embedded Prices only**. Standalone prices are not part of this field. Review the Product Variant documentation.

**2. Gift Items in Cart Discount**

* **Hint:** The quantity of a gift item added via a Cart Discount **cannot be changed** once added. Refer to the Cart Discount documentation.

**3. Removing Line Items from a Cart**

* **Hint:** Use the CartRemoveLineItemActionBuilder to remove a line item by its ID in the Java SDK. Study the Cart Update documentation.

**4. Unsupported HTTP Methods**

* **Hint:** The commercetools platform does **not support PUT** as an HTTP method. Learn more about commercetools API standards.

**5. Orders and Order Edits**

* **Hint:** New line items can be added through Order Edits but not directly to Orders. Explore the Order Edits documentation.

**6. Custom Type for Cart**

* **Hint:** To associate a Custom Type with a Cart, use resourceTypeIds with value ["order"]. Check the Custom Types documentation.

**7. Standalone Prices and Products**

* **Hint:** Standalone Prices are associated with products using their **SKU field**. Study the Standalone Prices documentation.

**8. State Names in Zones**

* **Hint:** State names in zones **do not need to comply with ISO 3166 standards**. Refer to the Zone documentation.

**9. Deleting Product Attributes**

* **Hint:** Deleting an attribute from a product type removes it from all product variants. Learn about Product Types documentation.

**10. Updating Product Keys**

* **Hint:** Use ProductSetKeyActionBuilder to update a product key in the Java SDK. Review the Product Update documentation.

**11. Java SDK Version Support**

* **Hint:** commercetools Java SDK supports **Java versions 8 and 9**. Visit the Java SDK documentation.

**12. Custom Line Item Price and Tax**

* **Hint:** Custom Line Items' price and tax are **not handled by the price selection** like Line Items. Check the Cart documentation.

**13. Administrators' Access in Organizations**

* **Hint:** Members of the Administrators team have **unrestricted access** to all projects and teams in the organization. Explore Organizations documentation.

**14. External Tax Mode**

* **Hint:** Use ExternalAmount as the taxMode when tax calculations are handled externally. Learn about Tax Mode documentation.

**15. Inventory Mode for Limited Items**

* **Hint:** Use ReserveOnOrder for limited supply items that cannot be restocked. Refer to the Inventory documentation.

**16. Composable Commerce Resources with Customer Data**

* **Hint:** The resources Customer, Order, and DiscountCode can contain or refer to customer data. Review the Customer Data documentation.

**17. Cart Update Actions for Multiple Shipping Modes**

* **Hint:** For a cart with multiple shipping modes, use:
  + addItemShippingAddress
  + addShippingMethod
  + setLineItemShippingDetails
  + addCustomShippingMethod
  + setCustomLineItemShippingDetails.  
    See Cart Update Actions documentation.

**18. Facet Results Query Parameters**

* **Hint:** Use filter.query and filter.facets to affect facet results. Learn more about Facet Search documentation.

# Kamesh Vasamsetti

**1. Product Description Requirement**

* **Hint:** A product description is **not required** to create a product. Only essential fields like name, slug, and product type are mandatory. Review the Product creation documentation.

**2. Cart Discount Scenarios**

* **Hint:** Discounts applied via **discount codes** (e.g., $15 off for subscribers) are implemented as Cart Discounts. Explore the Cart Discount API documentation.

**3. API Extension Trigger Actions**

* **Hint:** Multiple trigger actions can be configured per API Extension, enabling flexibility. Check the API Extensions documentation.

**4. Standalone Price Creation**

* **Hint:** Creating a Standalone Price with a non-existent SKU results in **a new Standalone Price**, as it does not require an associated SKU upfront. Refer to the Standalone Price documentation.

**5. Cart Discount Predicate**

* **Hint:** Use lineItemCount(1 = 1) >= 5 to target Carts with at least 5 line items. Study the Cart Discount predicate documentation.

**6. Filters and Facets**

* **Hint:** Filters **do not return terms** created based on attribute values; facets handle that function. Understand the Facets and Filters documentation.

**7. Freezing a Cart**

* **Hint:** Discounts that expire after a cart freeze are **still applied**. Check the Cart Freeze documentation.

**8. Shopping List Usage**

* **Hint:** A Shopping List is not used as **a product selection for stores**. Its purpose is for "save-for-later" and sharing lists. Explore the Shopping List documentation.

**9. Facet Query Parameters**

* **Hint:** Only filter.query and filter.facets affect facets. Avoid over-selecting. Study the Facet Query documentation.

**10. Subscriptions and API Extensions**

* **Hint:** Subscriptions trigger actions **after** an API call succeeds, while API Extensions trigger actions **before**. Learn more in the Subscriptions and API Extensions documentation.

**11. Strong Consistency in Product Projection**

* **Hint:** Product projection updates are **eventually consistent**, not strongly consistent. Review the Product Projection documentation.

**12. CloudEvents Subscription Payloads**

* **Hint:** CloudEvents payloads include a resource field but are **not always triggered by resource events**. Confirm details in the Subscription documentation.

**13. API Client Scope Best Practices**

* **Hint:** Define precise, narrow scopes to avoid unnecessary permissions. Broad scopes are not recommended. See API Client setup documentation.

**14. Product Discount Predicate**

* **Hint:** Use Price - CustomerGroup for applying targeted discounts in Product Discount predicates. Learn about Product Discount documentation.

**15. Orders and Order Edits**

* **Hint:** New line items can be added via **Order Edits** but not directly to Orders. Explore Order Edit documentation.

**16. Category Best Practices**

* **Hint:** Keep front-end navigation in mind while creating categories. Avoid deep nesting or multiple trees unnecessarily. Study Category best practices.

**17. Freezing Carts for Static Discounts**

* **Hint:** Carts can be frozen to maintain current prices and discounts, preventing further modifications. Check Cart Freeze documentation.

**18. Duplicate Emails in Stores**

* **Hint:** The same email address can be registered for multiple customers in different stores within the same project. Learn about Customer Accounts documentation.

**19. Query and Facets Relationship**

* **Hint:** filter.query is applied to query results **before** facets are calculated, influencing the output. Explore Facet Query documentation.

**20. External Authorization Tokens**

* **Hint:** External authorization tokens can be integrated into commercetools for secure access. Confirm in Authorization documentation.

**21. Deleting Line Items**

* **Hint:** Use CartRemoveLineItemActionBuilder to delete a line item. Study Cart Update Actions.

**22. Zone Configuration**

* **Hint:** Zones always include countries and can optionally include states. Refer to the Zone documentation.

**23. Updating Product Keys**

* **Hint:** Use ProductSetKeyActionBuilder for updating product keys in Java SDK. Learn from Product Update Actions.

# Arun Suresh

**1. Global Customers with Same Email**

* **Hint:** Customers in the same project cannot share the same email, even with different keys. Emails are globally unique in commercetools projects. Refer to Customer Accounts Documentation.

**2. Cart Freezing and Discounts**

* **Hint:** Freezing a cart retains applicable discounts even after expiration. Discounts are still considered unless manually removed. Cart Freezing Documentation.

**3. Product Type Attribute Reference**

* **Hint:** Use key-value-document for referencing custom objects in Product Types. Product Type Documentation.

**4. Filters and Facets**

* **Hint:** Filters **do not return terms** from attribute values; facets handle this functionality. Filters refine search results instead. Facet and Filter Documentation.

**5. Preventing Cart Modifications**

* **Hint:** Freezing a cart locks prices, taxes, and discounts to maintain consistency during order processing. Cart Documentation.

**6. Product Projections API**

* **Hint:** The API returns **either current or staged projections**, not both simultaneously. Product Projections API Documentation.

**7. State Machine Transitions**

* **Hint:** Define transitions to establish workflows between states in a state machine. State Machine Documentation.

**8. Inventory Mode for Limited Items**

* **Hint:** Use ReserveOnOrder for items with limited supply that cannot be restocked. Inventory Documentation.

**9. Pricing in Product Variants**

* **Hint:** Prices are stored as an array of objects representing price tiers. Pricing Documentation.

**10. API Limits**

* **Hint:** Refer to the API Reference Documentation for comprehensive limits and constraints. API Limits Documentation.

**11. API Extension Triggers**

* **Hint:** Triggers activate for specified resource actions such as "Create" or "Update" under defined conditions. API Extensions Documentation.

**12. HTTP Requests in Java SDK**

* **Hint:** The SDK supports both blocking and non-blocking HTTP requests. Java SDK Documentation.

**13. Multi-Buy Discount**

* **Hint:** The number of free items depends on the maxOccurrence setting in the discount configuration. Cart Discount Documentation.

**14. Cart Quantity Discounts**

* **Hint:** For "buy 3, get 1 free" discounts, the total discount for 10 items is **3 free items**. Discount Rules Documentation.

**15. Active Product Discounts**

* **Hint:** A maximum of **500 active discounts** is allowed by default. Product Discounts Documentation.

**16. Cart Discount Predicate**

* **Hint:** Use lineItemCount(1 = 1) >= 5 for discounts targeting carts with at least five line items. Cart Discount Predicate Documentation.

**17. Standalone Price Association**

* **Hint:** Standalone Prices are associated via the SKU field of products. Standalone Price Documentation.

**18. Fields in Resource vs Draft**

* **Hint:** Resources typically have more fields than their drafts due to additional system-generated fields. Resource Structure Documentation.

**19. Category Information**

* **Hint:** Categories contain information about their **parent and ancestors**, not just the parent. Category Documentation.

**20. Custom Line Items in Carts**

* **Hint:** Prices and taxes for custom line items are manually handled, not managed by price selection. Custom Line Item Documentation.

**21. Beta Features**

* **Hint:** Beta features are not under the API contract for backward compatibility and are subject to change. Beta Features Documentation.

**22. Best Practices for Categories**

* **Hint:** Keep front-end navigation structures in mind when creating categories to optimize usability. Category Best Practices.

**23. Cart Version Updates**

* **Hint:** Adding line items to a cart increments its version to **>=101**. Cart Versioning Documentation.

**24. Group Attributes**

* **Hint:** Use Set of Set of Text for attributes requiring grouped text sets. Attribute Types Documentation.

**25. Matching Shipping Methods**

* **Hint:** Use the Shipping Methods endpoint to fetch methods matching a cart. Shipping Methods Documentation.

**26. Java SDK Versions**

* **Hint:** Retrieve the latest Java SDK version from Maven Central. Java SDK Documentation.

**27. "In Production" Flag**

* **Hint:** Projects flagged as "In Production" cannot be deleted. Project Settings Documentation.

**28. Cart Merging with Different Currencies**

* **Hint:** Carts with different currencies **cannot** be merged. Cart Merging Documentation.

**29. Cart Shipping Mode Updates**

* **Hint:** Use addItemShippingAddress, addShippingMethod, and similar actions for multi-shipping mode. Cart Shipping Mode Documentation.