

100% Money Back
Guarantee

Vendor:Google

Exam Code:LOOKML-DEVELOPER

Exam Name:LookML Developer

Version:Demo

QUESTION 1

The code below shows a view `order_items` with its measures `total_revenue` and `user_count`

```
view: order_items {  
  measure: total_revenue {  
    type: sum  
    sql: ${TABLE}.sale_price ;;  
  }  
  
  measure: user_count {  
    type: count_distinct  
    sql: ${users.id} ;;  
  }  
}
```

11aW334953

Which code correctly represents a new measure that calculates average revenue per user?

- A. `measure: average_revenue_per_user {
type: number
sql: ${total_revenue}/${user_count} ;;
}`
- B. `measure: average_revenue_per_user {
type: average
sql: ${total_revenue}/${user_count} ;;
}`
- C. `measure: average_revenue_per_user {
type: number
sql: ${total_revenue}/${users.id};;
}`
- D. `measure: average_revenue_per_user {
type: average sql: ${total_revenue}/${users.id};;
}`

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: C

QUESTION 2

After running the Content Validator, a developer can see the error "Unknown field". Which two changes could cause this issue? (Choose two.)

- A. View name was changed from users to customers.
- B. Field type was changed from number to string.
- C. Model name was changed from e_commerce to reporting.
- D. Explore label was changed from users to customers.
- E. Field name was changed from id to user_id.

Correct Answer: BE

QUESTION 3

A developer needs to implement three persistent derived tables (PDTs) as described below.

The PDTs need to be refreshed after the daily ETL pipeline adds incremental loads to the underlying tables.

Each PDT is built off of one underlying table in the database (one PDT per table).

The underlying tables for each PDT are updated one after the other, and a new row is added to an ETL log table each time a table is updated.

Due to the unpredictable nature of the ETL pipeline, each PDT does not refresh at the same time from day to day.

Each PDT takes over an hour to build, and to save on compute costs each PDT should only be refreshed once per day.

How can the developer set up the PDTs according to these requirements?

- A. Create one datagroup tied to all three PDTs that runs when the total row count across all three tables changes.
- B. Create one datagroup tied to all three PDTs that parameterizes the view name for each PDT in the SQL trigger condition.
- C. Create three separate datagroups tied to three PDTs that run when each corresponding table's row count changes.
- D. Create three separate datagroups tied to three PDTs that run when a new row is added to the ETL log table.

Correct Answer: D

QUESTION 4

Users viewing an Explore should be able to view rows of data only where the value of the product.brand column matches the value of the user's company user attribute.

Which access filter should the developer use to meet this requirement?

- A.

```
access_filter: {  
  field: company  
  user_attribute: ${product.brand}  
}
```
- B.

```
access_filter: {  
  field: product.brand  
  user_attribute: company  
}
```
- C.

```
access_filter: {  
  field: user.company  
  user_attribute: brand  
}
```
- D.

```
access_filter: {  
  field: product.brand  
  user_attribute: {{ _user_attributes['company'] }}  
}
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: B

QUESTION 5

A LookML developer creates an Explore that joins two views. The base view has information about users' interactions with the support team. The joined view contains data about the users. The support team using this Explore feels overwhelmed by the amount of data this Explore shows them and decides to just look at open tickets.

What should the developer add to the Explore in the model to achieve these requirements?

- A. A filtered measure
- B. The hidden parameter
- C. The sql_always_where parameter
- D. A relationship definition

Correct Answer: D

QUESTION 6

A user reports that, when a date dimension is filtered to “before now” results are returned that consistently include tomorrow. Dimension fill has been ruled out as a cause of the issue.

Which LookML parameter should be used to resolve this issue?

- A. Week_start_day
- B. Convert_tz
- C. Datatype
- D. Fiscal_month_offset

Correct Answer: D

QUESTION 7

The developer has moved the orders Explore (shown below) from model_a to model_b, where both models are in the same project, and all users have access to both models.

Connection: “demo” include: “.view” explore: orders {}

What will happen after making this change?

- A. Dashboard tiles and Looks will be automatically pointed to the orders Explore in model_b.
- B. Dashboard tiles and Looks will redirect to the new database connection.
- C. Dashboard tiles and Looks that rely on this Explore will be deleted.
- D. Dashboard tiles and Looks that rely on this Explore will return an error.

Correct Answer: C

QUESTION 8

A LookML developer has a transactions view with several measures that each perform complex calculations involving

multiple fields. The LookML developer creates an Explore based on the transactions view. The product team wants to perform further functions on these measures, such as SUM, AVG, MIN, MAX, and RANK. The team wants these further functions to be performed at different levels of detail: weekly, monthly, and yearly.

How can the LookML developer model these requirements and minimize the amount of code rewriting?

- A. Add measures to the transactions view of type: number to apply the required functions.
- B. Change the existing measures in the transactions view to dimensions, and add measures of the different required types.
- C. Create a constant for each measure so it can be reused across other areas of the LookML project.
- D. Create native derived tables using transactions as the explore_source.

Correct Answer: D

QUESTION 9

A developer is defining the users table within a view file in Looker. The users table will be available as an individual Explore and it will also be joined into other Explores, such as the products Explore. The developer needs to limit the fields visible in the products Explore without affecting the visibility of the fields in the users Explore.

How should the developer meet this requirement?

- A. Use the fields parameter at the join level for the products Explore to specify which fields should be included and leave the users Explore as is.
- B. Create duplicate dimensions and measures, one for the users Explore and one for the products Explore, and use the hidden parameter to modify the visibility of the fields.
- C. Create two view files for the users table. One view file will have all possible fields for the users Explore, and the other will have only the fields required for the products Explore.
- D. Use the hidden parameter in the users view file for the fields that should not come over to the products Explore and leave the users Explore as is.

Correct Answer: A

QUESTION 10

A developer is connecting a LookML project to a remote Git repository. The developer wants to track which users are committing code changes, creating pull requests, or deploying to production when the different Git commands are initiated from within Looker.

Which type of Git connection should be utilized to meet this business requirement?

- A. A bare Git repository
- B. Multiple account HTTPS
- C. Single account HTTPS

D. SSH

Correct Answer: D

QUESTION 11

A developer needs to add an Explore built off of the orders view, which surfaces only completed orders. An orders Explore exists that contains all order information. Fields from the orders view are also referenced in other existing views such as `${orders.fieldname}`.

How should developer define a new Explore for completed orders and keep all field references working correctly?

- A.

```
explore: completed_orders {  
  sql_always_where: ${orders.status} = "complete" ;;  
  view_name: orders  
}
```
- B.

```
explore: completed_orders {  
  sql_always_where: ${orders.status} = "complete" ;;  
  from: orders  
}
```
- C.

```
explore: completed_orders {  
  always_filter: {  
    A field: orders.status  
    A value: "complete"  
  }  
  from: orders  
  view_name: orders  
}
```
- D.

```
explore: completed_orders {  
  always_filter: {  
    A field: orders.status  
    A value: "complete"  
  }  
  from: completed_orders  
  view_name: orders  
}
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: C

QUESTION 12

A LookML Developer is working with denormalized tables and needs to create a measure adding up the Order Shipping column in the table below:

Order Item ID	Order ID	Order Shipping
1	1	10.00
2	1	10.00
3	2	20.00
4	2	20.00
5	2	20.00

- A.

```
measure: total_shipping {  
  type: sum  
  sql: ${order_shipping} ;;  
}
```
- B.

```
measure: total_shipping {  
  type: sum_distinct  
  sql: ${order_shipping} ;;  
}
```
- C.

```
measure: total_shipping {  
  type: sum_distinct  
  sql_distinct_key: ${order_id} ;;  
  sql: ${order_shipping} ;;  
}
```
- D.

```
measure: total_shipping {  
  type: sum  
  sql_distinct_key: ${order_id} ;;  
  sql: ${order_shipping} ;;  
}
```

raw34953

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Correct Answer: A