LOOKML-DEVELOPER Q&As

LookML Developer

Pass Google LOOKML-DEVELOPER Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.pass4itsure.com/lookml-developer.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Google Official Exam Center

QUESTION 1

A developer would like to add a new dimension of type: yesno for the enabled column in their users table. The column is of type: string in the database and returns yes and no values.

How should the developer define the yesno dimension?

A.
```
dimension: is_enabled {

type: yesno

sql: $(TABLE).enabled IS NOT NULL ;;

}
```

B.
```
dimension: is_enabled {

type: yesno

sql: CASE WHEN $(TABLE).enabled = ""yes"" then ""Yes"" ELSE ""No""
END;;

}
```

C.
```
dimension: is_enabled {

type: yesno

sql: $(TABLE).enabled ;;

}
```

D.
```
dimension: is_enabled {

type: yesno

sql: $(TABLE).enabled = ""yes"" ;;

}
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: A

QUESTION 2

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 3

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 4

A developer wants to calculate the ratio of total sales from the orders view and total users from the users view.

Which two methods can be used to create a measure that meets these requirements? (Choose two.)

A.

```
view: users{

measure: total_users{

type: count

}

measure: total_sales_per_user {

type: sum

sql: 1.0*${orders.total_sales)/${total_users};;

value_format_name: usd

}
}

view: orders{

dimension: sale_price{

type: number

sql: ${TABLE}.sale_price;;

}

measure: total_sales{

type: sum

sql: ${sale_price};;

}
}
```

B.

```
view: users{

measure: total_users{

type: count

}

measure: total_sales_per_user {

type: number

sql: 1.0*${orders.total_sales)/${total_users};;

value_format_name: usd

 }
}
view: orders{

dimension: sale_price{

type: number

sql: ${TABLE}.sale_price;;

}

measure: total_sales{

type: sum

sql: ${sale_price};;

}
}
```

C.

```
view: users{

measure: total_users{

type: count

}
}

view: orders{

dimension: sale_price{

type: number

sql: ${TABLE}.sale_price;;

}

measure: total_sales{

type: sum

sql: ${sale_price};;

}

measure: total_sales_per_user {

type: number

sql: 1.0*${total_sales)/users.${total_users};;

value_format_name: usd

}
}
```

D.

```
view: users{

measure: total_users{

type: count

}
}

view: orders{

dimension: sale_price{

type: number

sql: ${TABLE}.sale_price;;

}

measure: total_sales{

type: sum

sql: ${sale_price};;

}

measure: total_sales_per_user {

type: number

sql: 1.0*${total_sales)/${users.total_users};;

value_format_name: usd

}
}
```

E.

```
view: users{

measure: total_users{

type: count

}

measure: total_sales_per_user {

type: number

sql: 1.0*${total_sales)/${total_users};;

value_format_name: usd

}
}

view: orders{

dimension: sale_price{

type: number

sql: ${TABLE}.sale_price;;

}

measure: total_sales{

type: sum

sql: ${sale_price};;
```

A. Option A

B. Option B

C. Option C

D. Option D

E. Option E

Correct Answer: AC

QUESTION 5

A developer needs to model out LookML to convert existing reports into Looker. The existing reports are:

Report 1: A report with order and order_items data, which finds the order with the largest total value of the order_item prices. Report 2: A report with order and order_items data, which finds the order with the largest total number of

products ordered.

Report 3: A report with data on every product, whether or not it has been ordered.

Each database table used is updated in real time as orders are made.

How should the developer construct an Explore using the order_items view as the base view?

A. Create one persistent derived table to calculate Report 1, create one persistent derived table to calculate Report 2, and join in the products view with a full_outer join.

B. Create one persistent derived table to calculate Reports 1 and 2, and join in the products view with a full_outer join.

C. Create one ephemeral derived table to calculate Report 1, create one ephemeral derived table to calculate Report 2, and join in the products view with a left_outer join.

D. Create one ephemeral derived table to calculate Reports 1 and 2, and join in the products view with a full_outer join.

Correct Answer: A

QUESTION 6

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 7

Users have built a popular dashboard and want to have change management built in for any edits made to the dashboard. The developer sets up version control for the model on which the dashboard is based.

What should the developer build to meet the business requirement?

A. A native derived table based on the dashboard.

B. A dashboard LookML file included in the project.

C. A link to the dashboard included in the project.

D. An Explore LookML file based on the dashboard.

Correct Answer: B

QUESTION 8

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} ;;
}
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: A

QUESTION 9

A developer wants to create a new Explore based on the order_items view. The developer creates an

Explore in the ecommerce model file with the following definition:

explore: order_items {}

After saving and validations, the developer receives this LookML validator error:

Inaccessible view “inventory_items”, “inventory_items” is not accessible in explore” “order_items”. Check

for typos and missing joins in explore “order_items”.

What caused this error to appear?

A. A field in the order_items view references a field in the inventory_items view.

B. A field in the inventory_items view references a field in the order_items view.

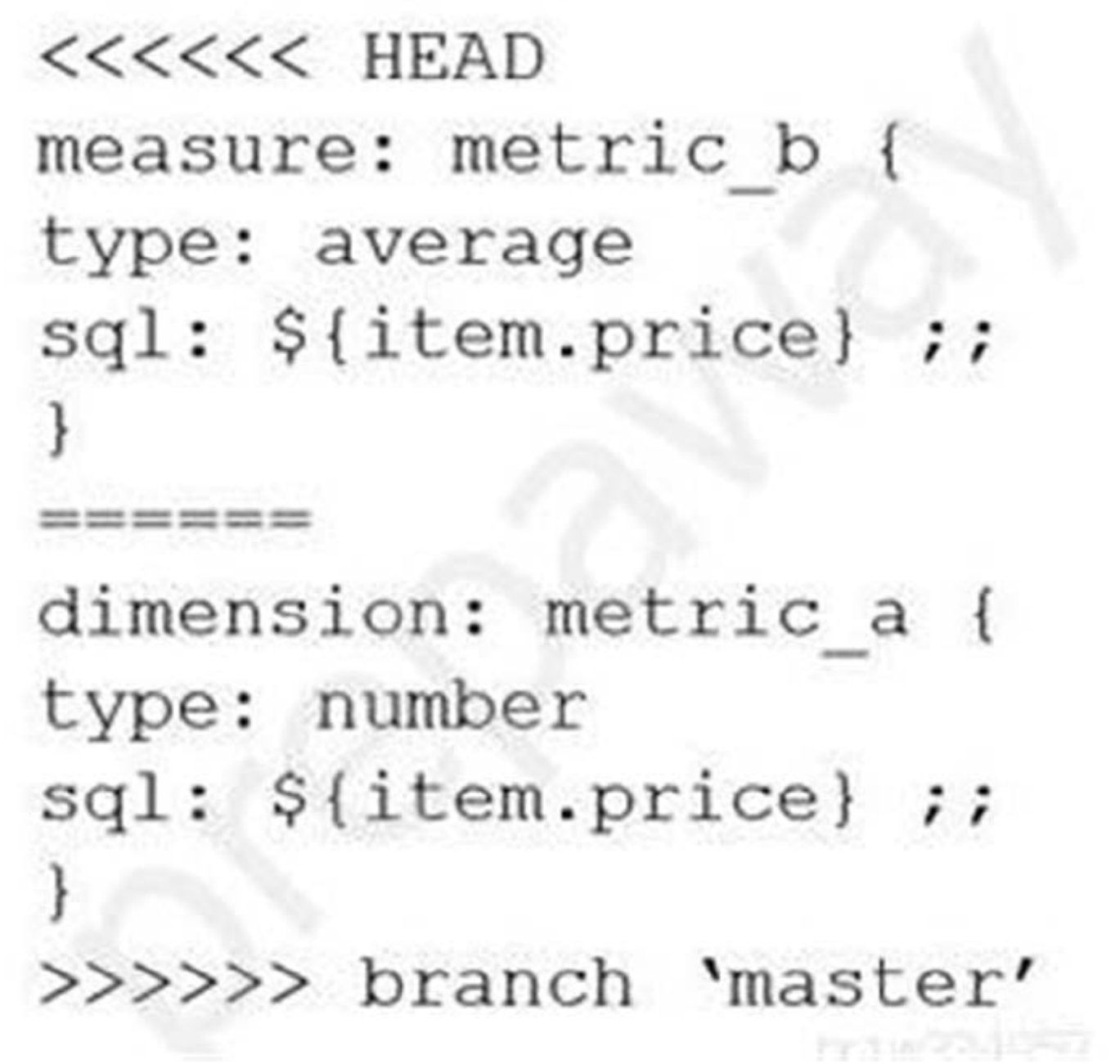
C. There is an Explore named inventory_items which references the order_items view.

D. There is another Explore named order_items which references the inventory_items view.

Correct Answer: A

QUESTION 10

A developer commits changes after adding LookML for a new measure. Upon pulling from production, the developer notices the following lines in the LookML:



A. Remove “andlt;> branch ‘master’”

B. Remove “andlt;