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.pass2lead.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

Only users with department attributes of Finance and Executive should be able to access the revenue view. Only users with the value of Executive for the department user attribute should be able to view the total_revenue field.

Given the code snippet below: How should the required access grants be structured to set up this system of access?

A. required_access_grants: [grant_b] in the revenue view, required_access_grants: [grant_a] in the total_revenue field

B. required_access_grants: [grant_a] in the revenue view, required_access_grants: [grant_a, grant_b] in the total_revenue field

C. required_access_grants: [grant_b] in the financial_data Explore, required_access_grants: [grant_a] in the total_revenue field

D. required_access_grants: [grant_a, grant_b] in the revenue view, required_access_grants: [grant_a] in the total_revenue field

Correct Answer: B

QUESTION 2

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 3

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 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 developer creates a derived table and wants to add persistence to it. Because the table is not used on a frequent basis, the developer wants the table to be cached for 12 hours, but only when a user has queried it.

Which persistence parameter should be added to the derived table\\'s definition in order to satisfy this use case?

A. persist_with: “12 hours”

B. datagroup: 12_hours { max_cache_age: “12 hours” }

C. persist_for: “12 hours”

D. sql_trigger_value: SELECT FLOOR{UNIX_TIMESTAMP{} / {6*60*60}} ;;

Correct Answer: A