



VCE & PDF

PassApply.com

<https://www.passapply.com/lookml-developer.html>

2024 Latest passapply LOOKML-DEVELOPER PDF and VCE dumps

Download

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.passapply.com/lookml-developer.html>

100% Passing Guarantee
100% Money Back Assurance

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

-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers





QUESTION 1

A LookML developer finishes some LookML work and commits changes in their personal development branch. The developer is asked to Pull and Merge Other Changes.

What does this indicate?

- A. A change has been deployed to a shared branch.
- B. A change has been committed in another developer's personal branch.
- C. A change has been committed in another shared branch.
- D. A change has been deployed to production.

Correct Answer: B

QUESTION 2

Users report that every time they change the filter on their Explore, the filters take a very long time to populate.

How can the developer improve the filtering experience with this Explore?

- A. Limit the filter suggestions using the suggestions parameter.
- B. Add an `always_filter` parameter to restrict the filter suggestions.
- C. Use an `access_filter` parameter to automatically apply filters.
- D. Add persistence to the base view of the Explore.

Correct Answer: A

QUESTION 3

A developer needs to build a new dimension that offers an age-based cohort representation of users. Which LookML code should the developer use to meet the requirement?



- A.

```
dimension: age_field {  
  type: bins  
  bins_size: 30  
  style: classic  
  sql: ${age} ;;  
}
```
- B.

```
dimension: age_field {  
  type: groups  
  groups: [<30, 30-60, >60]  
  sql: ${age} ;;  
}
```
- C.

```
dimension: age_field {  
  type: string_tiers: [0 to 30, 30 to 60, 60 and above]  
  style: classic  
  sql: ${age} ;;  
}
```
- D.

```
dimension: age_field {  
  type: tier_tiers: [0, 30, 60]  
  style: classic A sql: ${age} ;;  
}
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: B

QUESTION 4

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?



```
explore: financial_data {  
  view_name: base_table
```

```
  join: revenue {  
  }  
}
```

```
view: revenue {  
  measure: total_revenue {}  
}
```

```
access_grant: grant_a {  
  user_attribute: department  
  allowed_values: ["executive"]  
}
```

```
access_grant: grant_b {  
  user_attribute: department  
  allowed_values: ["finance", "executive"]  
}
```

- 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 5

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  
}
```

11/26/2024

A. Option A

B. Option B

C. Option C



D. Option D

Correct Answer: C

[LOOKML-DEVELOPER
PDF Dumps](#)

[LOOKML-DEVELOPER
Practice Test](#)

[LOOKML-DEVELOPER
Braindumps](#)