DUMPSBOSS.

HCIP-HarmonyOS Application Developer V1.0

Huawei H14-231_V1.0

Version Demo

Total Demo Questions: 2

Total Premium Questions: 23

Buy Premium PDF

https://dumpsboss.co

support@dumpsboss.co

support@dumpsboss.co
dumpsboss.co

Topic Break Down

Topic	No. of Questions
Topic 1, HarmonyOS Application Development Foundation	2
Topic 2, ArkTS Application Development	2
Topic 3, ArkUI Application Development	6
Topic 4, Application Model	9
Topic 5, Application Data Management	2
Topic 6, Application Security	1
Topic 7, Application Performance Optimization	1
Total	23

QUESTION NO: 1

An ArkUI page renders a list of orders with `ForEach`. Orders can be inserted, removed, and reordered, and each order item contains state that must remain associated with the correct order. Which key strategy is most appropriate?

A. Use the current array index as the key

B. Use each order's stable unique identifier as the key

C. Use the total number of orders as the key

D. Generate a new random value for every render as the key

ANSWER: B

Explanation:

Use each order's stable unique identifier as the key is correct. A stable unique key enables ArkUI to identify the same logical item when list positions change, preserving the correct item-component association. The array index changes when items are inserted or reordered, so it is not suitable for this requirement. A generated value that changes on every render likewise prevents stable identification.

QUESTION NO: 2

An application needs access to a user-sensitive permission before it reads protected data. Select three actions that the application should perform.

A. Declare the required permission in module.json5.

B. Request the permission from the user at runtime.

C. Handle a denied permission result before reading the protected data.

D. Set the UIAbility exported attribute to true.

E. Assume that declaration automatically grants the permission.

ANSWER: A B C

Explanation:

The application must declare the required permission in **module.json5**, request the permission from the user at runtime when it is needed, and handle the authorization result before attempting the protected operation. Declaring a permission does not automatically grant it. The exported attribute controls whether an ability can be invoked by other applications and is unrelated to user authorization.