

Android Applications UI/UX Design and Monetization Techniques

Android AND-803

Version Demo

Total Demo Questions: 6

Total Premium Questions: 66

Buy Premium PDF

<https://passqueen.com>

support@passqueen.com

QUESTION NO: 1

You should add the following XML code illustrated in the following image to AdView widget in order to add an Ad to your activity:

Which two choices of the following are correct about adUnitId ? (Select two)

- A. An ad unit ID is a unique identifier assigned to each of your ad units when they are created in AdMob.
- B. The ad unit Id is added to your app's code and is used to identify ad requests from the ad unit.
- C. The ad unit Id is created by Android Studio, and it must be used for all ads in the same app.
- D. The ad unit Id is used as a user name to login to AdMob web site to check your revenue of this ad.

ANSWER: A B

Explanation:

An ad unit ID is a unique identifier assigned to an individual ad unit when that unit is created in the AdMob console. An app can contain multiple ad units—for example, separate units for banner, interstitial, rewarded, or native ads—and each unit has its own ID. This identifier lets AdMob associate an ad request with the specific configured ad unit, including its ad format, settings, and reporting.

The ad unit ID is included in the app configuration or code, such as the `adUnitId` attribute of an `AdView` in XML or the corresponding property when an ad view is configured programmatically. When the app loads an ad, the Google Mobile Ads SDK uses that value to identify which ad unit is making the request. For production releases, developers must use the ad unit ID created for their own app in AdMob. During development, Google recommends using test ad unit IDs or registering test devices so that test impressions and clicks are not treated as invalid activity. See the [Google Mobile Ads SDK banner guide](#) and the [AdMob guidance on ad units](#).

QUESTION NO: 2

An Android drawing app sells a permanent premium brush collection through Google Play. A user should buy the collection once and retain access after reinstalling the app on another device. After a successful purchase, which handling is most appropriate?

- A. Consume the purchase after the user selects a brush for the first time.
- B. Grant and retain the entitlement as a non-consumable one-time purchase, then acknowledge the purchase.
- C. Create a monthly subscription for the brush collection after the first purchase.
- D. Store a local Boolean value and never query previous purchases again.

ANSWER: B

Explanation:

Grant and retain the entitlement as a non-consumable one-time purchase, then acknowledge the purchase. is correct. The brush collection is a durable entitlement: it is not depleted each time it is used and should be available again when the user restores purchases on another device. The app should grant access only after handling the completed purchase correctly and should acknowledge an eligible purchase.

Consuming the purchase would be appropriate for an item that can be bought repeatedly and depleted, such as virtual currency. Treating the purchase as a subscription would incorrectly imply recurring access and renewal. Saving only a local flag is also insufficient because it cannot reliably restore entitlement after reinstalling or changing devices.

QUESTION NO: 3

A team is adding analytics to understand whether users complete a new account-recovery flow. Which three choices represent useful and responsible analytics practices? (Select three)

- A. Record events for starting, completing, and abandoning the recovery flow.
- B. Measure the percentage of users who complete recovery after starting it.
- C. Exclude passwords, verification codes, and other personally identifiable information from event parameters.
- D. Send every character entered in each text field as an analytics event.

ANSWER: A B C

Explanation:

Recording meaningful recovery-flow events, measuring the completion and abandonment rate, and avoiding personally identifiable information in event parameters are correct. Events should represent meaningful user actions, such as starting recovery, receiving a verification step, completing recovery, or leaving the flow. These events can be used to identify where users experience friction.

Analytics should collect only data needed to improve the product. Passwords, verification codes, and similar sensitive values must never be included in analytics events. Recording every character typed produces unnecessary data and does not provide an appropriate measure of usability.

QUESTION NO: 4

A developer has completed a major redesign and wants to identify installation, navigation, and payment problems before exposing the update to all users. Which Google Play release approach is the best first step?

- A. Release the update to a limited testing track before a production rollout.
- B. Upload the update directly to production and rely on user reviews for testing.
- C. Change the app from free to paid before uploading the redesign.
- D. Create a new application package name for every interface update.

ANSWER: A

Explanation:

Release the update to a limited testing track before a production rollout. is correct because testing tracks allow a developer to distribute a release to a controlled audience and collect feedback before broad availability. This reduces the impact of usability, compatibility, and monetization defects discovered after real-device use. Uploading directly to production provides less control, while changing the app price or creating a new package name does not validate the redesigned experience.

QUESTION NO: 5

A UX designer is reviewing a prototype for an app that lets users transfer money between their own accounts. Which three choices are appropriate before the user confirms the transfer? (Select three)

- A. Show the source account, destination account, and transfer amount.
- B. Use a clear confirmation action, such as "Confirm transfer."
- C. Allow the user to return and edit the transfer details before confirmation.
- D. Use an unlabeled checkmark as the only confirmation control.

E. Hide the account details and require the user to remember them from the previous screen.

ANSWER: A B C

Explanation:

Showing the source account, destination account, and amount, using clear action wording, and providing a way to return and edit details are correct. A confirmation screen should help users detect mistakes before an important action is committed. Clear information and predictable navigation support user trust and reduce accidental transfers.

Making the confirmation action ambiguous or concealing key details creates avoidable risk. Requiring users to memorize account details from a prior screen adds unnecessary cognitive load and makes error prevention more difficult.

QUESTION NO: 6

Which of the following Android files will be released to Google Play?

- A. APK file.
- B. MainActivity.java or MainActivity.Kt.
- C. Manifest.xml.
- D. All app files.
- E. Android App Bundle (AAB) file.

ANSWER: E

Explanation:

An Android App Bundle (AAB) file is the standard publishing artifact for releasing a new Android app to Google Play. An AAB contains the app's compiled code and resources, organized so that Google Play can generate and deliver optimized APKs tailored to each device configuration, such as its CPU architecture, screen density, and language. This reduces the amount of code and resources a user needs to download compared with distributing one universal APK. Developers create a signed release bundle, normally with the `bundleRelease` Gradle task, and upload the resulting `.aab` file through Play Console. Google Play then uses that bundle to produce the installable APKs served to users; the source files and project configuration files are inputs to the build rather than the publishing artifact. Google's Android Developers documentation states that new apps submitted to Google Play must publish with Android App Bundles, subject to limited exceptions for specific distribution scenarios. See the [Android App Bundle overview](#) and [Google Play Console guidance on app bundles](#).