Android MCQ 2

Certainly! Here are multiple-choice questions (MCQs) from the topics of Drawables, Styles & Themes, Card and Color, and Floating Action Buttons in Android:

Drawables:

1. In Android, what is the purpose of a "drawable" resource?

   a. To define the layout and structure of an activity.

   b. To provide images, icons, and other visual assets.

   c. To specify the themes and styles for an app.

   d. To store app-specific data.

   Answer: b. To provide images, icons, and other visual assets.

2. Which folder in the Android project contains drawable resources for different screen densities?

   a. drawable-mdpi

   b. drawable-hdpi

   c. drawable-xhdpi

   d. drawable

   Answer: d. drawable

Styles & Themes:

3. In Android, what is a "style" used for?

   a. To define the layout of an activity.

   b. To specify the colors and fonts of text views.

   c. To group a set of common attributes and apply them to views.

   d. To create animations and transitions.

   Answer: c. To group a set of common attributes and apply them to views.

4. What is the primary advantage of using styles and themes in Android?

   a. They make your app smaller in size.

   b. They improve the performance of your app.

   c. They provide a consistent look and feel across the app.

   d. They add security features to the app.

   Answer: c. They provide a consistent look and feel across the app.

Card and Color:

5. What is the purpose of the "CardView" widget in Android?

   a. To create a list of items.

   b. To provide a container with a card-like appearance.

   c. To display images.

   d. To handle user input.

   Answer: b. To provide a container with a card-like appearance.

6. In Android, what is a color resource?

   a. A Java class for managing colors.

   b. An XML file that defines color values.

   c. A resource for storing images.

   d. A layout file for arranging views.

   Answer: b. An XML file that defines color values.

Floating Action Buttons:

7. What is a Floating Action Button (FAB) used for in Android?

   a. To display notifications.

   b. To provide a floating menu.

   c. To trigger a primary action in the app.

   d. To display advertisements.

   Answer: c. To trigger a primary action in the app.

8. Which library is commonly used to implement Floating Action Buttons in Android apps?

   a. Android UI Widgets Library

   b. Android Material Design Library

   c. Android Support Library

   d. Android AppCompat Library

   Answer: b. Android Material Design Library

9. How can you add an icon to a Floating Action Button in Android?

   a. By using the `setIcon()` method.

   b. By setting the `android:icon` attribute in XML.

   c. By adding an `ImageView` inside the FAB layout.

   d. FABs do not support icons.

   Answer: b. By setting the `android:icon` attribute in XML.

10. In Android, what is the recommended use case for a Mini Floating Action Button (Mini FAB)?

   a. To perform secondary actions in the app.

   b. To display app notifications.

   c. To create a floating menu.

   d. To navigate to different app screens.

   Answer: a. To perform secondary actions in the app.

Certainly! Here are more multiple-choice questions (MCQs) from the topics of Drawables, Styles & Themes, Card and Color, and Floating Action Buttons in Android:

Drawables:

11. Which drawable resource folder should you use for images that should be displayed on high-density screens?

   a. drawable-mdpi

   b. drawable-hdpi

c. drawable-xhdpi

d. drawable-xxhdpi

Answer: c. drawable-xhdpi

12. What is a 9-patch drawable in Android?

    a. A drawable with nine different color patches.

    b. A resizable bitmap image that can stretch without losing quality.

    c. A drawable with nine different layers.

    d. A drawable for displaying text.

    Answer: b. A resizable bitmap image that can stretch without losing quality.

Styles & Themes:

13. In Android, what is the purpose of a "theme"?

    a. To specify the layout of an activity.

    b. To group a set of common attributes and apply them to views.

    c. To define the colors and fonts of text views.

    d. To create animations and transitions.

    Answer: b. To group a set of common attributes and apply them to views.

14. What is the difference between a style and a theme in Android?

    a. Styles are used for views, while themes are used for activities.

b. Styles are defined in XML, while themes are defined in Java code.

c. Styles can be applied to individual views, while themes are applied to the entire app.

d. Styles define layouts, while themes define colors and fonts.

Answer: c. Styles can be applied to individual views, while themes are applied to the entire app.

Card and Color:

15. What is the purpose of the "elevation" property in a CardView?

a. To specify the background color of the card.

b. To control the elevation (shadow) of the card relative to other views.

c. To set the corner radius of the card.

d. To define the card's layout.

Answer: b. To control the elevation (shadow) of the card relative to other views.

16. In Android, how can you define a custom color in XML?

a. By using the `<color>` element with a hex color code.

b. By using the `<customColor>` element with a name.

c. By adding a new color resource folder.

d. By using the `setCustomColor()` method in Java code.

Answer: a. By using the `<color>` element with a hex color code.

Floating Action Buttons:

17. What is the primary use case for a Speed Dial Floating Action Button (Speed Dial FAB) in Android?

    a. To display notifications.

    b. To create a floating menu with multiple actions.

    c. To trigger a primary action in the app.

    d. To navigate to different app screens.

    Answer: b. To create a floating menu with multiple actions.

18. How can you add a label to a Floating Action Button in Android?

    a. By using the `setLabel()` method.

    b. By setting the `android:label` attribute in XML.

    c. By adding a `TextView` below the FAB.

    d. FABs do not support labels.

    Answer: c. By adding a `TextView` below the FAB.

19. What is the purpose of the "mini" attribute in a Floating Action Button (FAB)?

    a. It sets the FAB's size to be larger than the standard size.

    b. It makes the FAB smaller than the standard size.

    c. It changes the color of the FAB.

    d. It adds a shadow to the FAB.

    Answer: b. It makes the FAB smaller than the standard size.

20. Which library provides support for Material Design components like Floating Action Buttons in Android?

    a. Android UI Widgets Library

    b. Android Jetpack Library

    c. Android Material Design Library

    d. Android Support Library

    Answer: c. Android Material Design Library

Of course! Here are more multiple-choice questions (MCQs) related to Drawables, Styles & Themes, Card and Color, and Floating Action Buttons in Android:

Drawables:

21. Which Android resource folder is used to store vector drawables?

    a. drawable-mdpi

    b. drawable-hdpi

    c. drawable-xhdpi

    d. drawable-vectordrawable

    Answer: d. drawable-vectordrawable

22. In Android, what is the purpose of a "Layer List" drawable?

    a. To define a list of image layers.

    b. To create complex shapes.

    c. To specify the colors and fonts of text views.

d. To store app-specific data.

Answer: a. To define a list of image layers.

Styles & Themes:

23. In Android, what is a theme attribute used for?

    a. To specify a color.

    b. To define a style.

    c. To reference a theme.

    d. To customize the appearance of UI elements.

    Answer: d. To customize the appearance of UI elements.

24. What is the purpose of a parent theme in Android?

    a. It defines the primary color of the app.

    b. It groups together a set of common attributes.

    c. It specifies the layout of an activity.

    d. It defines the version of Android to use.

    Answer: b. It groups together a set of common attributes.

Card and Color:

25. What is the primary advantage of using a CardView in Android?

    a. To simplify the layout of an app.

b. To provide a card-like appearance with shadows and rounded corners.

c. To store app data.

d. To display images in a grid layout.

Answer: b. To provide a card-like appearance with shadows and rounded corners.

26. In Android, what is the primary purpose of a color state list?

a. To define a custom color.

b. To specify the background color of a view.

c. To define different colors for different states of a view (e.g., normal, pressed).

d. To create gradients.

Answer: c. To define different colors for different states of a view (e.g., normal, pressed).

Floating Action Buttons:

27. In Android, what is the purpose of the "extended" Floating Action Button (FAB)?

a. To provide a larger FAB.

b. To add labels to the FAB.

c. To display additional actions.

d. To create a floating menu.

Answer: c. To display additional actions.

28. What is the recommended action to associate with the main Floating Action Button (FAB) in Android?

   a. Display a menu.

   b. Delete an item.

   c. Open a navigation drawer.

   d. Play a video.

   Answer: b. Delete an item.

29. How can you change the background color of a Floating Action Button in Android?

   a. By using the `setBackgroundColor()` method.

   b. By setting the `android:background` attribute in XML.

   c. By adding a background image.

   d. FABs do not support background color changes.

   Answer: a. By using the `setBackgroundColor()` method.

30. What is the purpose of the "app:srcCompat" attribute when setting the icon of a Floating Action Button?

   a. To specify the source code for the icon.

   b. To provide compatibility with older Android versions for vector drawables.

   c. To define the shape of the FAB.

   d. To set the FAB's background color.

Answer: b. To provide compatibility with older Android versions for vector drawables.

These additional MCQs cover various aspects of Drawables, Styles & Themes, Card and Color, and Floating Action Buttons in Android app development.

Background Processes : Asyn Task, Services, Broadcast Receiver

Certainly! Here are multiple-choice questions (MCQs) related to Background Processes in Android, including AsyncTask, Services, and Broadcast Receivers:
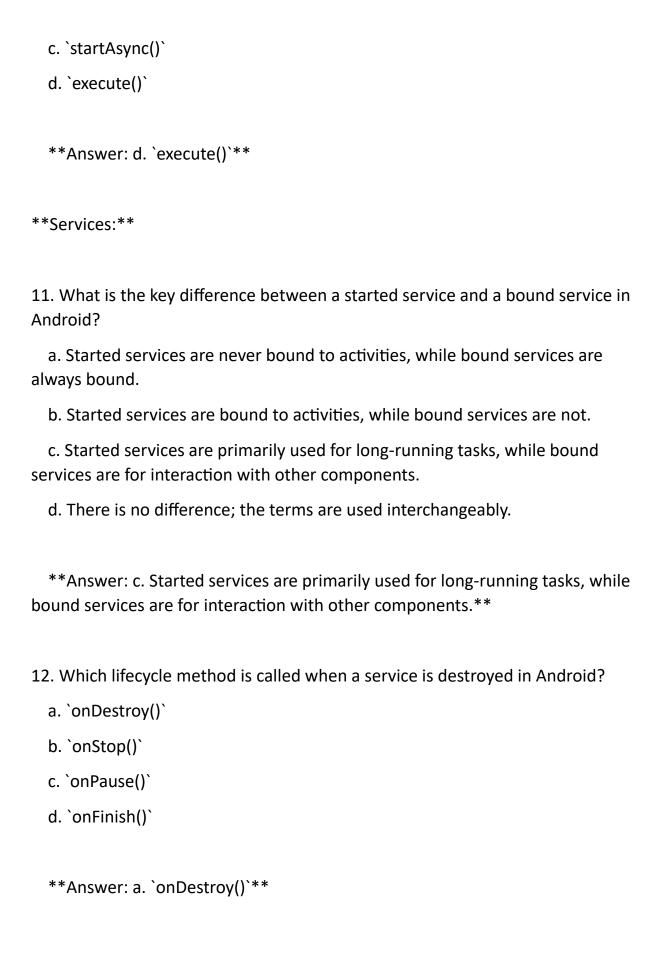
AsyncTask:

1. What is AsyncTask primarily used for in Android?

   a. Performing long-running operations on the main thread.

   b. Downloading and caching large data files.

   c. Running tasks in the background and updating the UI on the main thread.

   d. Managing services in the background.

   Answer: c. Running tasks in the background and updating the UI on the main thread.

2. In an AsyncTask, which method is executed on the main thread?

   a. `onPostExecute()`

   b. `doInBackground()`

c. `onPreExecute()`

d. `onProgressUpdate()`

Answer: a. `onPostExecute()`

Services:

3. What is the primary purpose of a bound service in Android?

   a. To perform tasks in the background without user interaction.

   b. To interact with other components, such as activities or fragments.

   c. To broadcast system-wide events.

   d. To display notifications to the user.

   Answer: b. To interact with other components, such as activities or fragments.

4. Which method is called when a service is first created in Android?

   a. `onStartCommand()`

   b. `onCreate()`

   c. `onBind()`

   d. `onStart()`

   Answer: b. `onCreate()`

Broadcast Receiver:

5. What is the primary purpose of a broadcast receiver in Android?

a. To send data between activities.

b. To perform background tasks.

c. To respond to system-wide or application-specific broadcast messages.

d. To manage background services.

Answer: c. To respond to system-wide or application-specific broadcast messages.

6. Which component is responsible for registering and unregistering broadcast receivers?

a. Activity

b. Service

c. BroadcastReceiver

d. Context

Answer: d. Context

7. What is the intent filter used for in a broadcast receiver?

a. To filter incoming broadcast messages based on action strings.

b. To define the layout of the broadcast receiver's UI.

c. To specify the permissions required for the receiver.

d. To determine the priority of the receiver.

Answer: a. To filter incoming broadcast messages based on action strings.

8. Which method is invoked when a broadcast is received by a BroadcastReceiver?

  a. `onCreate()`

  b. `onReceive()`

  c. `onStart()`

  d. `onHandleIntent()`

  Answer: b. `onReceive()`

Certainly! Here are more multiple-choice questions (MCQs) related to Background Processes in Android, including AsyncTask, Services, and Broadcast Receivers:

AsyncTask:

9. What is the main limitation of using AsyncTask for background tasks in Android?

  a. AsyncTask cannot be used for background tasks.

  b. AsyncTask can only run for a maximum of 10 seconds.

  c. AsyncTask may lead to memory leaks if not handled properly.

  d. AsyncTask cannot be executed in parallel.

  Answer: c. AsyncTask may lead to memory leaks if not handled properly.

10. Which method is used to execute an AsyncTask in Android?

  a. `executeTask()`

  b. `runInBackground()`

c. `startAsync()`

d. `execute()`

Answer: d. `execute()`

Services:

11. What is the key difference between a started service and a bound service in Android?

    a. Started services are never bound to activities, while bound services are always bound.

    b. Started services are bound to activities, while bound services are not.

    c. Started services are primarily used for long-running tasks, while bound services are for interaction with other components.

    d. There is no difference; the terms are used interchangeably.

    Answer: c. Started services are primarily used for long-running tasks, while bound services are for interaction with other components.

12. Which lifecycle method is called when a service is destroyed in Android?

    a. `onDestroy()`

    b. `onStop()`

    c. `onPause()`

    d. `onFinish()`

    Answer: a. `onDestroy()`

Broadcast Receiver:

13. In Android, what is the purpose of the "LocalBroadcastManager"?

    a. To send broadcasts between different applications.

    b. To send broadcasts within the same application only.

    c. To receive broadcasts from remote servers.

    d. To handle system-wide broadcasts.

    Answer: b. To send broadcasts within the same application only.

14. Which permission is required to receive broadcasts with a permission-level protection level?

    a. `RECEIVE_BROADCAST`

    b. `SEND_BROADCAST`

    c. `BROADCAST_PERMISSION`

    d. `RECEIVE_SMS`

    Answer: c. `BROADCAST_PERMISSION`

15. What is the purpose of the "abortBroadcast()" method in a broadcast receiver?

    a. To stop a broadcast from being sent to other receivers.

    b. To abort the current application and start a new one.

    c. To unregister the broadcast receiver.

    d. To prevent a service from running.

    Answer: a. To stop a broadcast from being sent to other receivers.

16. When using a BroadcastReceiver, what is the recommended approach for performing long-running tasks?

    a. Run the task in the main thread.

    b. Use an IntentService.

    c. Execute the task in a bound service.

    d. Use AsyncTask within the BroadcastReceiver.

    Answer: b. Use an IntentService.

Certainly! Here are more multiple-choice questions (MCQs) related to Background Processes in Android, including AsyncTask, Services, and Broadcast Receivers:

AsyncTask:

17. In an AsyncTask, what is the purpose of the `onProgressUpdate()` method?

    a. To perform background tasks.

    b. To execute code on the main thread before starting the background task.

    c. To publish progress updates to the UI thread.

    d. To execute code when the AsyncTask is canceled.

    Answer: c. To publish progress updates to the UI thread.

18. What is the recommended way to handle configuration changes (e.g., screen rotation) when using AsyncTask in Android?

    a. Use a Service instead of an AsyncTask.

b. Use a ViewModel to store and manage AsyncTask results.

c. Re-run the AsyncTask from scratch in `onCreate()`.

d. Implement `onRetainNonConfigurationInstance()`.

Answer: b. Use a ViewModel to store and manage AsyncTask results.

Services:

19. Which type of Android service runs in the same process as the application and runs indefinitely until explicitly stopped?

a. Foreground Service

b. Bound Service

c. Intent Service

d. Started Service

Answer: a. Foreground Service

20. What is the primary use case for an Intent Service in Android?

a. Performing long-running tasks on the main thread.

b. Handling incoming SMS messages.

c. Running tasks sequentially in a background thread.

d. Binding to an activity.

Answer: c. Running tasks sequentially in a background thread.

Broadcast Receiver:

21. In Android, what is the purpose of the "ordered broadcast" feature?

    a. To broadcast system events.

    b. To send broadcasts between different applications.

    c. To specify a broadcast's priority and order of receivers.

    d. To restrict broadcasts to specific receivers.

    Answer: c. To specify a broadcast's priority and order of receivers.

22. When should you register a BroadcastReceiver dynamically using the `registerReceiver()` method?

    a. Only when the application is first installed.

    b. In the AndroidManifest.xml file.

    c. In the `onCreate()` method of an Activity.

    d. When you want to receive broadcasts only once.

    Answer: c. In the `onCreate()` method of an Activity.

23. What is the purpose of the "android:exported" attribute in the `<receiver>` element of the AndroidManifest.xml file?

    a. To specify the receiver's priority.

    b. To indicate whether the receiver is accessible to other apps.

    c. To define the receiver's permissions.

    d. To set the receiver's label.

    Answer: b. To indicate whether the receiver is accessible to other apps.

24. In Android, which broadcast action is commonly used to trigger a BroadcastReceiver when the device's network connectivity changes?

    a. `android.net.wifi.STATE_CHANGE`

    b. `android.intent.action.BATTERY_CHANGED`

    c. `android.intent.action.SCREEN_ON`

    d. `android.intent.action.AIRPLANE_MODE`

    Answer: a. `android.net.wifi.STATE_CHANGE`

These additional questions cover various aspects of AsyncTask, Services, and Broadcast Receivers in Android background processing.

Certainly! Here are more multiple-choice questions (MCQs) related to Background Processes in Android, including AsyncTask, Services, and Broadcast Receivers:

AsyncTask:

25. What is the purpose of the `onPostExecute()` method in an AsyncTask?

    a. To execute code on the main thread before starting the background task.

    b. To publish progress updates to the UI thread.

    c. To perform background tasks.

    d. To execute code on the main thread after the background task is complete.

    Answer: d. To execute code on the main thread after the background task is complete.

26. In an AsyncTask, which method should be used to perform background tasks off the main thread?

    a. `onPreExecute()`

    b. `onPostExecute()`

    c. `doInBackground()`

    d. `onProgressUpdate()`

    Answer: c. `doInBackground()`

Services:

27. Which method should be implemented in a Service to handle incoming intents?

    a. `onStartCommand()`

    b. `onCreate()`

    c. `onBind()`

    d. `onHandleIntent()`

    Answer: a. `onStartCommand()`

28. What is the primary difference between a foreground service and a background service in Android?

    a. Foreground services run with higher priority.

    b. Foreground services can't display notifications.

    c. Background services can't interact with other components.

    d. Background services run on a separate thread.

Answer: a. Foreground services run with higher priority.

Broadcast Receiver:

29. In Android, which component can send a broadcast?

a. Only the system itself.

b. Any installed app or the system.

c. Only apps with special permissions.

d. Only the app that registered the receiver.

Answer: b. Any installed app or the system.

30. When should you register a BroadcastReceiver in the AndroidManifest.xml file?

a. When the receiver needs to receive broadcasts dynamically.

b. When the receiver should be explicitly triggered by an activity.

c. When the receiver is part of an exported content provider.

d. When the receiver is not intended to receive broadcasts.

Answer: d. When the receiver is not intended to receive broadcasts.

31. What happens if you call `abortBroadcast()` in a BroadcastReceiver?

a. The broadcast is stopped, and no other receivers receive it.

b. The broadcast is delayed by a specified time.

c. The broadcast is forwarded to all available receivers.

d. The broadcast is logged in the system log.

Answer: a. The broadcast is stopped, and no other receivers receive it.

32. What is the purpose of the "sticky broadcast" in Android?

    a. It sticks to the screen until the user dismisses it.

    b. It persists even after the broadcast receiver unregisters.

    c. It sends a broadcast repeatedly at regular intervals.

    d. It's a type of broadcast with high priority.

    Answer: b. It persists even after the broadcast receiver unregisters.

These questions cover various aspects of AsyncTask, Services, and Broadcast Receivers in Android background processing.

Certainly! Here are more multiple-choice questions (MCQs) related to Background Processes in Android, including AsyncTask, Services, and Broadcast Receivers:

AsyncTask:

33. In an AsyncTask, which method is used to pass data from the background thread to the main UI thread?

    a. `onPostExecute()`

    b. `doInBackground()`

    c. `onPreExecute()`

    d. `onProgressUpdate()`

    Answer: d. `onProgressUpdate()`

34. What is the purpose of the `onPreExecute()` method in an AsyncTask?

    a. To execute code on the main thread before starting the background task.

    b. To perform background tasks.

    c. To publish progress updates to the UI thread.

    d. To execute code on the main thread after the background task is complete.

    Answer: a. To execute code on the main thread before starting the background task.

Services:

35. In Android, what is the primary use case for a bound service?

    a. To perform long-running tasks in the background.

    b. To interact with other components and share data.

    c. To display notifications.

    d. To handle incoming SMS messages.

    Answer: b. To interact with other components and share data.

36. Which lifecycle method of a service is called when it is unbound from all clients?

    a. `onDestroy()`

    b. `onStop()`

    c. `onUnbind()`

    d. `onDisconnected()`

Answer: c. `onUnbind()`

Broadcast Receiver:

37. In Android, what is the purpose of the "sendOrderedBroadcast()" method?

   a. To send a broadcast to all available receivers simultaneously.

   b. To send a broadcast with a specified delay.

   c. To send a broadcast with a specified order of receivers.

   d. To send a broadcast to only the foreground activities.

   Answer: c. To send a broadcast with a specified order of receivers.

38. What is the purpose of the "LocalBroadcastManager" in Android?

   a. To send broadcasts between different applications.

   b. To send broadcasts within the same application only.

   c. To restrict access to the broadcast receiver.

   d. To schedule broadcasts for future delivery.

   Answer: b. To send broadcasts within the same application only.

39. When should you register a BroadcastReceiver using the `<receiver>` element in the AndroidManifest.xml file?

   a. When the receiver needs to receive broadcasts dynamically.

   b. When the receiver should be explicitly triggered by an activity.

   c. When the receiver is part of an exported content provider.

   d. When the receiver is not intended to receive broadcasts.

Answer: c. When the receiver is part of an exported content provider.

40. What is the primary use case for a "sticky" broadcast in Android?

    a. To send a broadcast with high priority.

    b. To send a broadcast that sticks to the screen.

    c. To send a broadcast that is available for retrieval by future broadcasts.

    d. To send a broadcast to a specific package.

    Answer: c. To send a broadcast that is available for retrieval by future broadcasts.

These additional questions cover various aspects of AsyncTask, Services, and Broadcast Receivers in Android background processing.