

MAJLIS ARTS AND SCIENCE COLLEGE

PG DEPARTMENT OF COMPUTER SCIENCE

(Affiliated to the University of Calicut, approved by the Government of Kerala)

Majlis Nagar, Puramannur-P.O 676552, Malappuram Dt, Kerala, 0494 2643970, 9539111174



Sixth Semester Online Study Camp.

Study Camp. 6 BCA , BSc
WhatsApp group



Scan QR Code to Join Study Camp WhatsApp Group

- * UNIT WISE REVISION
- * PREVIOUS YEAR QUESTION PAPER DISCUSSION
- * ASSIGNMENTS

masc.majliscomplex.org

MODULE 2

SYLLABUS

Understanding android resources - String resources, Layout resources, Resource reference syntax, Defining own resource IDs - Enumerating key android resources, string arrays, plurals, Colour resources, dimension resources, image resources.

Understanding content providers - Android built in providers, exploring databases on emulator, architecture of content providers, structure of android content URIs, reading data using URIs, using android cursor, working with where clause, inserting updates and deletes, implementing content.

Understanding intents - basics of intents, available intents, exploring intent composition, Rules for Resolving Intents to Their Components, ACTION PICK, GET CONTENT, pending intents.

Short Answer Questions (1 Mark)

1. What is android cursor?

Ans: A cursor object provides random read access to the rows and columns in the database.

2. What is layout resource?

Ans: The user files that define a user interface layout which is saved in res/layout/ and accessed from R.layout class.

3. Text localization is done using which resource in android?

Ans:String resources.

4. What is the use of resource ID in android?

Ans:To uniquely identify the resources

5. What is the common name to refer collection of styles in android?

Ans: styles.xml.

Short Answer Questions

6. What is resource in android? Explain plurals with example.

Ans: Static content that your code uses, such as bitmaps, colors, layout definitions, user interface

strings, animation instructions, and more are included in Resources.It is Maintained separately in various sub-directories under res/ directory of project.

Android Plurals are XML based resources which allow to handle different quantities. In the XML file you specify values for the quantities “zero”, “one”, “two”, “many”, “few”, “many”, “other” and code use the method `getQuantityString()` to get the correct value.

To differentiate between plural and singular strings, you can define a plural in your *strings.xml* file and list the different quantities, as shown in the example below:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <plurals name="hello_people">
        <item quantity="one">Hello to %d person</item>
        <item quantity="other">Hello to %d people</item>
    </plurals>
</resources>
```

7. Explain any two types of drawable resources?

Ans: A *Drawable* resource is a general concept for a graphic which can be drawn. The simplest case is a graphical file (bitmap), which would be represented in Android via a *BitmapDrawable* class.

Every Drawable is stored as individual files in one of the *res/drawable* folders. Typically you would store bitmaps for different resolutions in the *-mdpi*, *-hdpi*, *-xhdpi*, *-xxhdpi* subfolders of *res/drawable*.

Bitmap File

A bitmap graphic file (.png, .jpg, or .gif). Creates a [BitmapDrawable](#).

Shape Drawable

An XML file that defines a geometric shape, including colors and gradients. Creates a [GradientDrawable](#).

8. What is a content provider?

Ans: A content provider presents data to external applications as one or more tables that are similar to the tables found in a relational database. A row represents an instance of some type of data the provider collects, and each column in the row represents an individual piece of data collected for an instance. Manages access to central repository of data. Provider is a part of an Android application, which provides its own UI for working with data.

9. What are android cursors?

Ans: A cursor object provides random read access to the rows and columnist contains.

Using cursor methods, you can iterate over the rows in the results, determine the data type of each column, get the data out of a column, and examine other properties of the results.

Some cursor implementations automatically update the object when the provider's data changes, or trigger methods in an observer object when the Cursor changes, or both.

10. What is an intent?

Ans: A cursor object provides random read access to the rows and columnist contains.

Using cursor methods, you can iterate over the rows in the results, determine the data type of each column, get the data out of a column, and examine other properties of the results.

Some cursor implementations automatically update the object when the provider's data changes, or trigger methods in an observer object when the Cursor changes, or both.

Paragraph Questions(5 Marks)

11.Describe any 4 type of resources in android.

Ans: **Animation Resources**

Define pre-determined animations.Tween animations are saved in res/anim/ and accessed from the R.anim class.Frame animations are saved in res/drawable/ and accessed from the R.drawable class.

Color State List Resource

Define a color resources that changes based on the View state.Saved in res/color/ and accessed from the R.color class.

Drawable Resources

Define various graphics with bitmaps or XML.Saved in res/drawable/ and accessed from the R.drawable class.

Layout Resource

Define the layout for your application UI.Saved in res/layout/ and accessed from the R.layout class.

Menu Resource

Define the contents of your application menus.Saved in res/menu/ and accessed from the R.menu class.

String Resources

Define strings, string arrays, and plurals (and include string formatting and styling). Saved in res/values/ and accessed from the R.string, R.array, and R.plurals classes.

Style Resource

Define the look and format for UI elements.Saved in res/values/ and accessed from the R.style class.

Font Resources

Define font families and include custom fonts in XML.Saved in res/font/ and accessed from the R.font class.

More Resource Types

Define other primitive values as static resources, including the following:

Bool-XML resource that carries a boolean value.

Color-XML resource that carries a color value (a hexadecimal color).

Dimension-XML resource that carries a dimension value (with a unit of measure).

ID-XML resource that provides a unique identifier for application resources and components.

Integer

XML resource that carries an integer value.

Integer Array

XML resource that provides an array of integers.

Typed Array

XML resource that provides a [TypedArray](#) (which you can use for an array of drawables).

12. What is dimension resource? Which all the units used to specify dimensions? Give XML syntax for defining dimension resource.

Ans: A dimension value defined in XML. A dimension is specified with a number followed by a unit of measure. For example: 10px, 2in, 5sp. The following units of measure are supported by Android:

Dp: Density-independent Pixels - An abstract unit that is based on the physical density of the screen. These units are relative to a 160 dpi (dots per inch) screen, on which 1dp is roughly equal to 1px..

Sp: Scale-independent Pixels - This is like the dp unit, but it is also scaled by the user's font size preference. It is recommended you use this unit when specifying font sizes, so they will be adjusted for both the screen density and the user's preference.

Pt: Points - 1/72 of an inch based on the physical size of the screen, assuming a 72dpi density screen.

Px: Pixels - Corresponds to actual pixels on the screen. This unit of measure is not recommended because the actual representation can vary across devices; each device may have a different number of pixels per inch and may have more or fewer total pixels available on the screen.

Mm: Millimeters - Based on the physical size of the screen.

In: Inches - Based on the physical size of the screen.

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<resources>
```

```
<dimen
```

```
    name="dimension_name"
```

```
>dimension</dimen>
```

```
</resources>
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<resources>
```

```
    <dimen name="textview_height">25dp</dimen>
```

```
    <dimen name="textview_width">150dp</dimen>
```

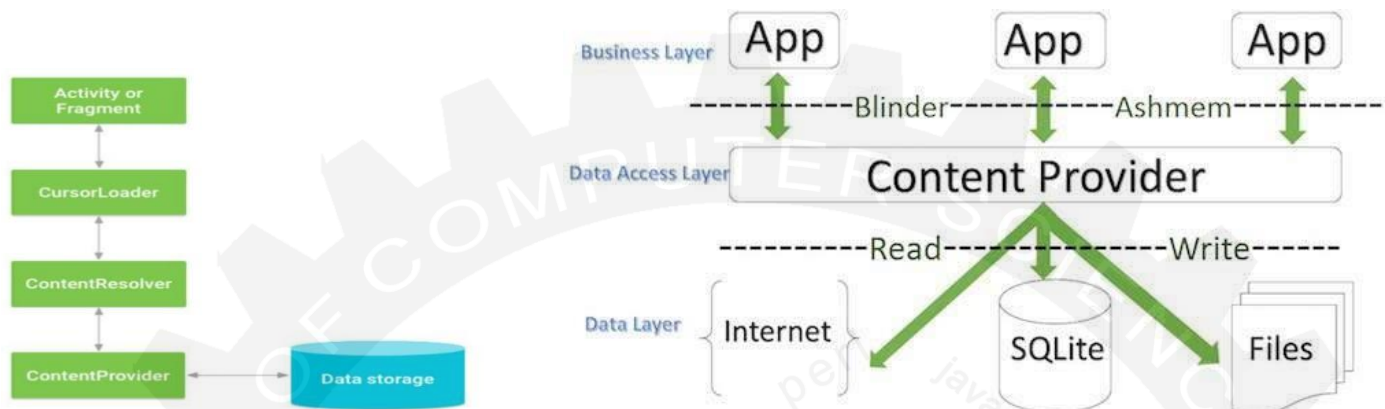
```
    <dimen name="ball_radius">30dp</dimen>
```

```
<dimen name="font_size">16sp</dimen>
```

```
</resources>
```

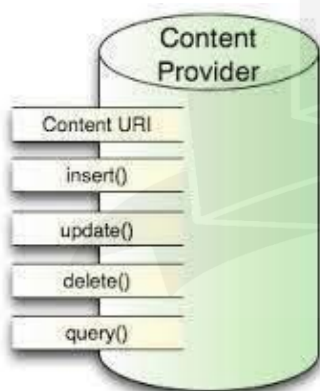
13.Explain about structure of content providers.

Ans: A content provider component supplies data from one application to others on request. Such requests are handled by the methods of the `ContentResolver` class. A content provider can use different ways to store its data and the data can be stored in a database, in files, or even over a network. This involves number of simple steps to create your own content provider.



- First of all you need to create a Content Provider class that extends the `ContentProviderbase` class.
- Second, you need to define your content provider URI address which will be used to access the content.
- Next you will need to create your own database to keep the content. Usually, Android uses SQLite database and framework needs to override `onCreate()` method which will use SQLite Open Helper method to create or open the provider's database. When your application is launched, the `onCreate()` handler of each of its Content Providers is called on the main application thread.
- Next you will have to implement Content Provider queries to perform different database specific operations.
- Finally register your Content Provider in your activity file using `<provider>` tag.

Here is the list of methods which you need to override in Content Provider class to have your Content Provider working –



- **onCreate()** This method is called when the provider is started.
- **query()** This method receives a request from a client. The result is returned as a Cursor object.
- **insert()** This method inserts a new record into the content provider.
- **delete()** This method deletes an existing record from the content provider.
- **update()** This method updates an existing record from the content provider.
- **getType()** This method returns the MIME type of the data at the given URI

Essay Question(10 Marks)

14. Briefly explain the following :-

- Colors.
- Strings
- Emulator
- Images.

Ans:

a. Colors

The color IDs are available under the **your-package.R.color** namespace. Android defines a base set of colors, which are accessible through the Android **android.R.color** namespace.

Example

The following shows some examples of specifying color in an XML resource file.

```
<resources>
  <color name="red">#f00</color>
  <color name="blue">#0000ff</color>
  <color name="green">#f0f0</color>
  <color name="main_back_ground_color">#ffffff00</color>
</resources>
```

The entries above need to be in a file residing in the `/res/values` subdirectory. The name of the file can be anything you choose. The following code shows an example of using a color resource in Java code.

```
int mainBackGroundColor = activity.getResources.getColor(R.color.main_back_ground_color);
```

The following code shows how you can use a color resource in a view definition.

```
<resources>
  <color name="red">#f00</color>
  <color name="blue">#0000ff</color>
  <color name="green">#f0f0</color>
```

```
<color name="main_back_ground_color">#ffffff00</color>
</resources>
```

The following code shows how to use defined color in layout resource.

```
<TextView android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:textColor="@color/red"
    android:text="Sample Text to Show Red Color"/>
```

Android allows you to define strings in one or more XML resource files. These XML files reside in the **/res/values** subdirectory. You can name the XML files the way you like, although you commonly see the file name as **strings.xml**.

Example

The following code shows an example of a string-resource file.

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="hello">hello</string>
    <string name="app_name">hello appname</string>
</resources>
```

c.Android Emulator

The **Android emulator** is an **Android Virtual Device (AVD)**, which represents a specific Android device. We can use the Android emulator as a target device to execute and test our Android application on our PC. The Android emulator provides almost all the functionality of a real device. We can get the incoming phone calls and text messages. It also gives the location of the device and simulates different network speeds. Android emulator simulates rotation and other hardware sensors.

We can run an Android app from the Android Studio project, or we can run an app which is installed on the Android Emulator as we run any app on a device.

To start the Android Emulator and run an application in our project:

In **Android Studio**, we need to create an Android Virtual Device (AVD) that the emulator can use to install and run your app. To create a new AVD:-

Creating AVD

If you want to emulate a real device, first create an AVD with the same device configurations as real device, then launch this AVD from AVD manager.

Changing Orientation

Usually by default when you launch the emulator, its orientation is vertical, but you can change its orientation by pressing Ctrl+F11 key from keyboard.

d. Image Resources

Android provides us with the ability to use image resources in our applications. We can put image files in res/drawable directory and access them from the xml layout or by the generated ID from R.java class file.

You can use image files of the following formats:

- PNG
- JPEG
- GIF

Notice that there are two restrictions:

- If you use two files with the same base name you will receive an error. You cannot use two files like Hello.jpeg and Hello.png.
- If you create any subdirectory under res/drawable directory any image files in it will be ignored.

```
<imageview android:layout_height="wrap_content"  
android:layout_width="wrap_content"  
android:src="@drawable/android" />
```