What are shared preferences? When do we use Shared Preferences?

Shared Preferences allow you to save and retrieve data in the form of key,value pair. When we want to store simple data like primitive types.

How are the values stored in shared preferences?

They get written to the preference file and a key is used to find the location in the file.

How do you get the associated file of shared preferences?

In order to use shared preferences, you have to call a method getSharedPreferences() that returns a SharedPreference instance pointing to the file that contains the values of preferences.

Can you have multiple shared preferences file?

Yes by using the getSharedPreferences() you can declare the name of the file used and you can make multiple files

How do you write a file to internal/external storage?

context.openFileOutput(filename, Context.MODE\_PRIVATE).use {  
it.write(fileContents.toByteArray())  
}

Or use MODE\_APPEND to not overwrite the file every time

What permission(s) are needed for the above?

For Internal none for External you need WRITE\_EXTERNAL\_STORAGE

How do you create a SQL database in Android?

First extend the class with SQLLiteOpenHelper, next override onCreate and onUpgrade. Make a constructor for the database, then make a SQL statement to make the table in onCreate and use execSQL to make the table.

What are methods you need to implement when extend the “native database” class?

onCreate and onUpgrade. Optionally onRead.

How to do you update the schema of the database?

You use onUpgrade

How do you execute a query in database helper class?

You use rawQuery() and a cursor to use the result

Name the methods in the database helper class used for CRUD operations?

onCreate, onUpgrade, rawQuery, and execSQL

How do you have an image in the SQL database?

You use a blob and convert the image from a bit map to a byte array to send to the database

Write a query to create a table

CREATE TABLE table

Write a query to select everything from a table

SELECT \* FROM table

What is a recyclerView?

RecyclerView is flexible and efficient version of ListView. It is an container for rendering larger data set of views that can be recycled and scrolled very efficiently. RecyclerView is like traditional [ListView](http://stacktips.com/tutorials/android/android-listview-tutorial) widget, but with more flexibility to customizes and optimized to work with larger datasets. It uses a subclass of RecyclerView.Adapter for providing views that represent items in a data set.

What is a list view?

Android ListView is a view which groups several items and display them in vertical scrollable list. The list items are automatically inserted to the list using an Adapter that pulls content from a source such as an array or database.

 Which is better and why?

RecyclerView. Direct horizontal scrolling is not supported in ListView. Whereas in RecyclerView you can easily implement horizontal and vertical scrolling.ListView supports only simple linear list view with vertical scrolling. Whereas RecyclerView supports three types of lists using RecyclerView.LayoutManager. 1) StaggeredGridLayoutManager 2) GridLayoutManager, 3) LinearLayoutManager. ListView gives you an option to add divider using dividerHeight parameter whereas RecyclerView enable you to customize divider (spacing) between two elements using RecyclerView.ItemDecoration class. Also ListView continually creates the views by using findbyid which increases the overhead but RecyclerView uses the findbyid once and then uses bindtoView to cycle out which elements are needed instead of recreating them all the time.

How would you implement the RecyclerView.Adapter class? Write a sample class explaining what each method is responsible for.

Class SampleAdaptor (pass in list to view): RecyclerView.Adaptor<SampleAdaptor.MyviewHolder>(){

Override fun onCreateView{ We create the view to display here. We use the layout inflater to get the id of the layout to use. Then we return the view we created to use.

Class MyviewHolder{ view = Layoutinflater.does stuff to get R.id.layout

Then return MyviewHolder(view)}}

Override fun onBindView(holder: MyviewHolder){

We bind the elements in the list we are using to the view we created to populate the views

holder.textview = list[position]

}

Override fun getItemCount(){

Returns the sixe of the list for layout manager to use.  
 return list.size}

How can you change the orientation of the recyclerView?

LinearLayoutManager layoutManager

= new LinearLayoutManager(this, LinearLayoutManager.HORIZONTAL, false);

Name some layout managers you can use with the recyclerView?

Linear, grid, and staggered grid