DAM 2 – M08: Programación multimedia y dispositivos móviles

Name:

Date:

1. **What does SDK mean?**
   1. Software Development Kit.
   2. Source Development Kit.
   3. SDK Development Kit.
   4. Soft Development Kit.
2. **What window or view will we use to see the debug output of our terminals or emulators?**
   1. Devices.
   2. Logcat.
   3. Output.
   4. Debug.
3. **What window or view will we use to see the messages derived from the operation of the emulator (not from the applications)?**
   1. Devices.
   2. Logcat.
   3. Output.
   4. Debug.
4. **Where can we test our applications?**
   1. Only on the emulator.
   2. Only on a terminal connected to the computer.
   3. In the emulator or in a terminal connected to the computer.
   4. In a terminal.
5. **What is the name of the Android application configuration file?**
   1. android.conf.
   2. AndroidConf.xml.
   3. AndroidManifest.xml.
   4. AndroidManifest.conf.
6. **If a reference to R.strings.app\_name is found in the java code,to which it refers?**
   1. To the string called app\_name inside ./res/values/strings.xml.
   2. To the file ./res/strings/app\_name.
   3. To the file ./res/values/app\_name.
   4. To the file ./asserts/app\_name.
7. **In which folder would you place your application's screen definition?**
   1. ./beef/.
   2. ./res/screen/.
   3. ./res/layout/.
   4. ./assets/.
8. **In which folder is the source code of our classes placed?**
   1. In anyone other than the root.
   2. ./beef/.
   3. ./source/.
   4. ./src/.
9. **Where are the image animations placed?**
   1. ./res/xml/.
   2. ./res/drawable/.
   3. ./res/layout/.
   4. ./res/values/.
10. **Where are the colors in colors.xml defined?**
    1. ./res/xml/.
    2. ./res/drawable/.
    3. ./res/values/.
    4. ./res/layout/.
11. **If we want to make an interface with a series of objects ordered from top to bottom:**
    1. We will use a TableLayout tag with TableRow
    2. We will use a vertical LinearLayout tag.
    3. We will use a ConstraintLayout tag.
    4. They are all true.
12. **Where are the texts placed and defined?**
    1. ./res/values/.
    2. ./res/drawable/.
    3. ./res/layout/.
    4. ./res/xml/.
13. **Where are the images used in the design of the applications placed?**
    1. ./res/values/.
    2. ./res/raw/.
    3. ./res/drawable/.
    4. ./res/images/.
14. **What extension do Android source code files have to have?**
    1. .and.
    2. .xml.
    3. .Java.
    4. .src.
15. **What is the R resource file?**
    1. A class that facilitates access to all files placed within the ./res/ folder.
    2. A file with the file names used throughout the project.
    3. A file generated by the compiler with the list of sources.
    4. On Android there is no R resource file.
16. **Once a java view object with an xml resource in the interface has been identified, what methods are those that provide us with its attributes?**
    1. set’s (setText(), setTag(), etc)
    2. make’s (makeText(), makeTag(), etc).
    3. get’s (getText(), getTag(), etc)
    4. do’s (doText(), doTag(), etc)
17. **How many ways do we have to make a button?**
    1. Define a java button object and execute the onClick method of the Button class.
    2. Define void's with a View parameter and define the buttons with the onClick.
    3. A listener with onClickListener from a class that implements it.
    4. b and c are true.
18. **What class do the classes responsible for displaying the screens that interact with the user have to inherit from?**
    1. MainActivity.
    2. Activity.
    3. Screen.
    4. Gui.
19. **Which statement is correct?**
    1. Each Android application runs in its own Linux process.
    2. An Android application can have multiple Linux processes.
    3. An Android application has as many Linux processes as there are threads.
    4. Android is a single-process system.
20. **What Android:layout parameters are responsible for the area occupied by the objects of the .xml interfaces?**
    1. height and width.
    2. weight.
    3. marginTop, marginBottom, marginStart, marginEnd .
    4. They are all true.
21. **What does a FragmentActivity and its modern derivatives allow?**
    1. Show activities by flowing the life cycle in the Fragment.
    2. Being able to select the different Fragments in the same Activity.
    3. Fragment the interface of the activity to place the interface elements.
    4. Facilitate communication between Activities.
22. **What method is called when creating an Activity?**
    1. onCreate().
    2. onStart().
    3. onResume().
    4. onPreStart().
23. **If I already have an application with an activity started. What is the sequence that will be executed if I lock and unlock the screen?**
    1. onCreate()-onStart()-onResume()-onPause()-onStop()-onDestroy().
    2. onPause()-onStop()-onRestart()-onStart()-onResume()
    3. onDestroy()-onPause()-onRestart()-onStart-onResume().
    4. onPause()-onStop()-onDestroy()-onCreate()-onStart()-onResume().
24. **Within the life cycle of an activity, what is the last method that is called?**
    1. onStop().
    2. onPause().
    3. onDestroy().
    4. onFinish().
25. **How are View objects organized in Android?**
    1. In linear form.
    2. In a sequential manner.
    3. In the shape of a tree.
    4. They do not have a fixed order.
26. **How can we define our user interface?**
    1. Through an XML file.
    2. With Android code (Java/Kotlin).
    3. Through an XML file and/or Android code (Java/Kotlin).
    4. Using a png type file.
27. **Which method of our activity should we call to set our main View object?**
    1. setMainView.
    2. setContentView.
    3. setRootView.
    4. setView.
28. **To identify each of the view components, what attribute do we use?**
    1. id
    2. yam.
    3. key.
    4. view.
29. **What ways are there to define an intention?**
    1. Intent(Intent.ACTION\_X) with X in {VIEW, DIAL, CALL, IMAGE\_CAPTURE, SEND, etc}
    2. checkPermission\_Intent and Intent.ACTION\_CALL.
    3. Implicit with Intent context and explicit with specific context.
    4. None is correct.
30. **Which of the following Intent.ACTION opens the android system phone app?**
    1. Intent.ACTION\_VIEW
    2. Intent.ACTION\_CALL
    3. Intent.ACTION\_IMAGE\_CAPTURE
    4. Intent.ACTION\_DIAL
31. **Where are permissions defined?**
    1. Build.gredle.kts (Module:App)
    2. Permission.xml
    3. AndroidManifest.xml
    4. MainActivity.java
32. **The configuration of the Android builder with the dependencies and plugins in which file is it?**
    1. Build.gredle.kts (Module:App)
    2. Permission.xml
    3. AndroidManifest.xml
    4. MainActivity.java
33. **How do we add additional information to an intention?**
    1. intent.putExtra(Context, resource)
    2. intent.addKey(key, resource)
    3. intent.setInfo(key, retry)
    4. None of the above.
34. **Identifiers defined at compile time are saved in a generated class in which one?**
    1. In a resource class called Interface.
    2. In a resource class called R.
    3. In a resource class called Resources.
    4. In a resource class called Android.
35. **Which of these ways of accessing context is incorrect?**
    1. getMyContext().
    2. getContext().
    3. this.
    4. getApplicationContext().
36. **The resources are saved in the “res” folder, but where is this folder placed within our project?**
    1. Wherever we want.
    2. Wherever we want, but by convention ./res/ is used.
    3. In res.
    4. In the package, next to the classes that will use them.
37. **In which subfolder are view layouts saved?**
    1. res/layout.
    2. res/view.
    3. res/drawable.
    4. res/xml.
38. **If I want to translate my application into English, in which folder will I place its texts?**
    1. res/values/strings-en.xml.
    2. res/values-en/strings.xml.
    3. res-en/values/strings.xml.
    4. It doesn't matter as long as you define it in the manifesto.
39. **The file “res/drawable/example.png” can be accessed from our java code with the R class. What will be its path?**
    1. R.drawable.example.
    2. R.drawable.example\_png.
    3. R.resources.drawable.example.
    4. R.getDrawable(“example.png”).
40. **The Bundle class is used to share information…**
    1. To set application preferences generally for all applications
    2. To create or restore activities with data from other activities.
    3. To communicate the objects of different Fragments.
    4. They are all true.
41. **What class allows us to save data associated with a key persistently and independently of the application's activities?**
    1. SharedPreferences.
    2. SharedData.
    3. KeyPreferences.
    4. KeyData.
42. **What types of data can the SharedPreferences class save?**
    1. Any type of data.
    2. Only the data that implements its interface.
    3. Data of primitive type (boolean, float, int, long, string).
    4. None of the above.
43. **What type of Activity can we use to implement a preferences editor?**
    1. Activity.
    2. ListActivity.
    3. MapActivity.
    4. PreferenceActivity.
44. **What does the PreferenceActivity activity need?**
    1. A layout with the configuration of preferences.
    2. An xml file with the preferences configuration.
    3. A layout that shows the preferences form.
    4. Nothing special.
45. **How can I retrieve a saved preference using the PreferenceActivity class?**
    1. With the getSharedPreferences method of Context.
    2. With the getSharedPreferences method of the Activity.
    3. With the getSharedPrefrerences method of the Application class.
    4. Creating a SharedPreferences object.