

## Your grade: 86.66%

[Next item →](#)

Your latest: 86.66% • Your highest: 86.66% • To pass you need at least 80%. We keep your highest score.

1. True or False: Android does not provide a way for the 3rd party app to start up activities belonging to other apps.

1 / 1 point

- True  
 False

**Correct**

Correct! You can start up activities belonging to other apps. For example, you can run the camera to take a photo that will be added directly to your application.

2. How is a Compose UI element represented?

1 / 1 point

- AppCompatActivity**  
 **View** Object  
 Tag in the **XML** file  
 Special function

**Correct**

Correct! The function is called **composable** and it is annotated with an **@Composable** annotation.

3. Which of the following **composable** layouts can be used in Jetpack Compose? Select all that apply.

1 / 1 point

- Box**  
 **Row**  
 **Column**  
 **Stack**

1 / 1 point

4. Which statements are true about modifiers? Select all that apply.

- A **modifier** is responsible for modifying **Activity** behavior.  
 Modifiers allow you to impart specific interactivity or behavior to the elements.

**Correct**

Correct! You can, for example, make a button or a text clickable using modifiers.

- When augmenting appearance, modifiers can act as decorators.

**Correct**

Correct! Modifiers decorate or style the UI elements or composables. Modifiers can, for example, modify the size, position, spacing and color.

5. In the code below, what is the reason behind usage of the **remember** function and **mutableStateOf** observable?

1 / 1 point

```
val value = remember { mutableStateOf(/*some value*/) }
```

- Reset of the variable value during recomposition.  
 Prevent reset of the variable value during recomposition.  
 Reset of the variable value during function call.

**Correct**

Correct! To retain the value, you have to use the `remember` function and `mutableStateOf` observable. Otherwise, the function would reset the variable to its initial value.

6. Which of the following are part of Jetpack navigation?

1 / 1 point

Select all that apply.

**NavGraph**

**Correct**

Correct! Every screen display in an Android application is a destination in navigation.

**NavCompose**

**NavController**

**Correct**

Correct! `NavController` is a container for displaying the current destination.

**NavController**

**Correct**

Correct! `NavController` is responsible for navigating between destinations while keeping track and manipulating the back state.

7. Each item in a Grid is known as a:

1 / 1 point

Column

Image

Cell

Row

**Correct**

Correct! In Android, a Grid is a scrollable layout that displays the items in two dimensions. A grid consists of both Rows and Columns. Each item in a Grid is known as a Cell.

8. What do the last two characters of the hexadecimal ARGB colors represent?

0 / 1 point

0x 00 00 00 00  
? ?

`alt=0x00000000`

`0x00000000`

- The value of blue color.
- The value of Black color.
- The value of red color.
- The value of alpha channel.
- The value of the green color.

**Incorrect**

Not quite. Please review the video [Defining colors ↗](#).

9. True or False: The main difference between `dp` (density independent pixel) and `sp` (scalable pixel) is that `sp` takes the text size setting of the user device into consideration when deciding what text size should be displayed.

1 / 1 point

- True  
 False

 **Correct**

Correct! The user may modify this setting in the Android system preferences and the text size is scaled accordingly in the application.

10. What will the output of this code be?

0 / 1 point

```
1 var visible by remember { mutableStateOf(true)}
2 } Column {
3     if (visible) {
4         Text(text = "Hello")
5     } Button(onClick = { visible = !visible }) {
6         Text("My Button")
7     }
8 }
```

- The first `Button` click will instantly show the `Text` and the next `Button` click will have no effect.  
 The first `Button` click will instantly hide the `Text` and the following `Button` click will instantly show the `Text`.  
 The first `Button` click will start the fade in animation of the `Text` and the following `Button` click will start the fade out animation of the `Text`.  
 The first `Button` click will start the fade out animation of the `Text` and following `Button` click will start fade in animation of the `Text`

 **Incorrect**

Not quite. Please review the video [Animated visibility](#).

11. Which of the following best describes what an `Activity` is?

1 / 1 point

- The first screen that appears after an app has launched.  
 Information registered in the manifest.  
 The app component that contains the user interface.  
 The business logic present within the `main()` method.

 **Correct**

Correct! An `Activity` contains part of an app's user interface and represents the app screen.

12. Which of the following is a container element that enables the grouping of elements

1 / 1 point

- `ViewGroup`  
 `GroupView`  
 `View`  
 `Group`

 **Correct**

Correct! A `ViewGroup` is a special view that does not have an appearance of its own.

13. Which of the following refers to the states an `Activity` goes through from the moment an app starts until it exits?

1 / 1 point

- state  
 lifecycle methods  
 activity lifecycle  
 callbacks

 **Correct**

Correct! The `Activity` class provides a number of callback functions which are called during different stages of the lifecycle.

14. Which of the following are examples of view groups? Select all that apply.

1 / 1 point

**LinearLayout**

**Correct**

Correct! **LinearLayout** organizes contained views horizontally or vertically as a single row or column.

**FrameLayout**

**Correct**

Correct! **FrameLayout** is designed to block out an area of the screen to display a single item.

**RelativeLayout**

**Correct**

Correct! **RelativeLayout** specifies the position of contained views relative to one another.

**ImageViewLayout**

15. Which of the following is the approach used by Jetpack Compose to build a user interface?

1 / 1 point

declarative

traditional

hierarchical

imperative

**Correct**

Correct! The declarative approach allows developers to focus more on the application logic rather than XML and visual design.

16. When writing composable functions, which of the following do you add before the start of the function definition?

1 / 1 point

**@annotation**

**@function**

**composable**

**@Composable**

**Correct**

Correct! Using composable functions means you no longer need to define any XML.

17. Which of the following is an annotation that allows developers to easily create components that are modular and reusable?

1 / 1 point

**@Annotation**

**@Composable**

**@Component**

**@Kotlin**

**Correct**

Correct! Jetpack Compose can be used to build a hierarchical UI consisting of composable functions.

18. Which of the following built-in layouts in Compose places composables vertically?

1 / 1 point

row

column

layout

box

**Correct**

Correct! For a row the main axis is the vertical axis, so you can set the vertical alignment.

19. Which of the following built-in layouts in Compose places composables vertically?

1 / 1 point

- box
- layout
- column
- row

 **Correct**

Correct! For a row the main axis is the vertical axis, so you can set the vertical alignment.

20. Which of the following is the correct way to use a modifier to give a Column a background color of black and fill the available space?

1 / 1 point

```
1 Column(  
2 | Modifier.fillMaxSize().background(Color(color: 0xFF000000))  
3 )
```

```
1 Column(  
2 | Modifier.fillMaxSize().backgroundColor(color: 0xFF000000)  
3 )  
4
```

```
1 Column(  
2 | modifier = Modifier.fillMaxSize().background(Color(color: 0xFF000000))  
3 )
```

 **Correct**

Correct! Modifiers can be chained and are applied from left to right.

21. Which of the following modifiers is used to set the background color of a composable?

1 / 1 point

- color**
- background**
- layoutColor**
- backgroundColor**

 **Correct**

Correct! The **background** modifier accepts a color object as an argument.

22. Which of the following is the default event handler added to the **Button** composable by Jetpack Compose?

1 / 1 point

- onInput**
- onPress**
- onClick**
- onStart**

 **Correct**

Correct! The **onClick** event handler will execute the specified code in response to the **Button** being clicked.

23. Which of the following can be used to implement interaction handling in Jetpack Compose?

1 / 1 point

- clicks
- event modifiers
- events
-

⌚ long presses

**Correct**

Correct! Event modifiers are a powerful tool for enhancing the user experience and providing a more dynamic and interactive UI.

24. Which of the following refers to a current snapshot or condition of a program that changed over time?

1 / 1 point

- events
- user interfaces
- activities
- state

**Correct**

Correct! State is generally referred to as the data of some piece of information of a program that changed over time.

25. What is a Scaffold in Jetpack Compose?

1 / 1 point

- A state management library.
- A fixed layout.
- A customizable layout.
- A collection of presets.

**Correct**

Correct! A Scaffold is a convenient way to combine UI patterns and Material Components.

26. Which of the following keywords is used for creating an object without creating a class?

1 / 1 point

- `override`
- `route`
- `object`
- `interface`

**Correct**

Correct! When implementing navigation in Jetpack Compose, destinations can be defined as objects.

27. Which of the following Jetpack Compose layout arrangements adds equal space to all children?

1 / 1 point

- start
- center
- space around
- space evenly

**Correct**

Correct! The space evenly arrangement adds equal space to the children of a `List` including space at the start and end.

28. When would you use a lazy layout?

1 / 1 point

- When you have performance issues.
- When you have a small number of items to display as a list.
- When designing a responsive layout.
- When you have a large number of items to display as a list.

**Correct**

Correct! A large or unknown number of items can lead to performance issues.

29. Which of the following allows you to separate application design details from the UI structure and behavior?

0 / 1 point

- composable
- themes
- design systems
- material components

 **Incorrect**

Not quite. Please review the video [The Theme function](#).

30. Which of the following APIs can be used to animate a value in Jetpack Compose?

0 / 1 point

- remember API
- animate\*AsState API
- animate API
- animatedVisibility API

 **Incorrect**

Not quite. Please review the video [Value animation](#).