



**GOBIERNO DE  
MÉXICO**

**EDUCACIÓN**  
SECRETARÍA DE EDUCACIÓN PÚBLICA



**TECNOLÓGICO  
NACIONAL DE MÉXICO**



# **TECNOLOGICO NACIONAL DE MEXICO**

## **INSTITUTO TECNOLOGICO DE CIUDAD MADERO**

**Carrera: Ingeniería en Sistemas Computacionales.**

**Materia: Programación Nativa para Móviles.**

**Maestro: Jorge Peralta Escobar.**

**Integrantes del equipo:**

**Karla Denisse Cruz Solís #21070310**

**Yahir Osvaldo Valero Hernández #21070330**

**Grupo: A**

**Hora: 09:00 – 10:00**

**Semestre: Enero - Junio 2025.**

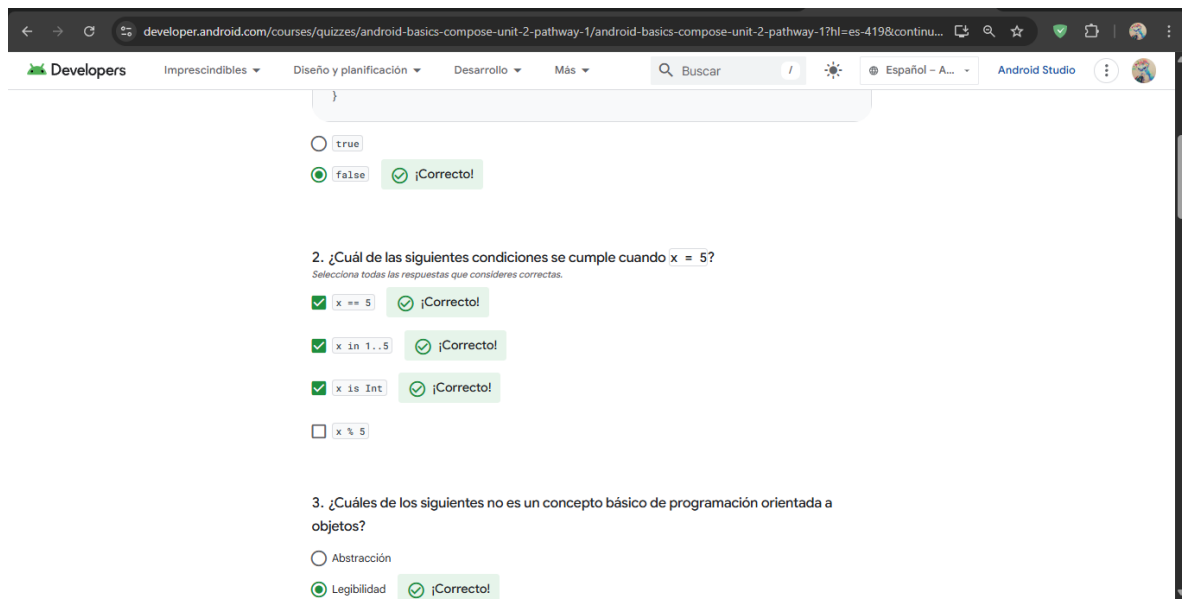
## Cuestionario U2- Conceptos básicos de Kotlin

Este cuestionario evalúa conocimientos fundamentales del lenguaje, incluyendo estructuras condicionales, operadores, tipos anulables, funciones y principios de programación orientada a objetos. También abarca temas como modificadores de visibilidad, uso de palabras clave como super, definición de interfaces, y el manejo de expresiones lambda.



The screenshot shows the introduction to the 'Conceptos básicos de Kotlin' quiz. The page header includes the Android Developers logo and navigation links. The main heading is 'Conceptos básicos de Kotlin'. Below it, a subtext says 'Hagamos una prueba rápida. Debes responder al menos 7 preguntas de manera correcta para aprobar el cuestionario.' There is a button labeled 'Volver a la ruta de aprendizaje'. The first question is: '1. El siguiente código imprimirá "Divisible by 5" si number es igual a 25.' Below the question is a code block containing Kotlin code: 

```
if (number % 10 == 0) {  
    println("Divisible by 10")  
} else if (number == 5) {  
    println("Divisible by 5")  
}
```



The screenshot shows the results for the first question. The question is: '1. El siguiente código imprimirá "Divisible by 5" si number es igual a 25.' The code block is repeated. Below the code, there are two radio button options: 'true' and 'false'. The 'false' option is selected, and a green checkmark with the text '¡Correcto!' is displayed next to it. Below this, the second question is: '2. ¿Cuál de las siguientes condiciones se cumple cuando x = 5?'. The subtext says 'Selecciona todas las respuestas que consideres correctas.' There are four checkbox options: 'x == 5', 'x in 1..5', 'x is Int', and 'x % 5'. The first three are checked, and each has a green checkmark with the text '¡Correcto!' next to it. The fourth option is not checked. Below this, the third question is: '3. ¿Cuáles de los siguientes no es un concepto básico de programación orientada a objetos?'. There are two radio button options: 'Abstracción' and 'Legibilidad'. The 'Legibilidad' option is selected, and a green checkmark with the text '¡Correcto!' is displayed next to it.

← → ↺ developer.android.com/courses/quizzes/android-basics-compose-unit-2-pathway-1/android-basics-compose-unit-2-pathway-1?hl=es-419&continuo... 🔍 ☆ 📁 📄 📄 📄

🐾 Developers Imprescindibles ▾ Diseño y planificación ▾ Desarrollo ▾ Google Play Comunidad 🔍 Buscar / 🌞 ● Español - A... ▾ Android Studio ⋮

3. ¿Cuáles de los siguientes no es un concepto básico de programación orientada a objetos?

- ☐ Abstracción
- ☒ Legibilidad ✅ ¡Correcto!
- ☐ Inheritance
- ☐ Polimorfismo

4. ¿Cuáles son los cuatro modificadores de visibilidad en Kotlin?

- ☐ public, private, protected y abstract
- ☐ static, override, internal y external
- ☒ private, protected, public y internal ✅ ¡Correcto!
- ☐ public, protected, static y internal

5. La palabra clave \_\_\_ se usa para llamar a un método de la clase superior.

- ☐ this
- ☒ super ✅ ¡Correcto!
- ☐ parent

← → ↺ developer.android.com/courses/quizzes/android-basics-compose-unit-2-pathway-1/android-basics-compose-unit-2-pathway-1?hl=es-419&continuo... 🔍 ☆ 📁 📄 📄 📄

🐾 Developers Imprescindibles ▾ Diseño y planificación ▾ Desarrollo ▾ Google Play Comunidad 🔍 Buscar / 🌞 ● Español - A... ▾ Android Studio ⋮

5. La palabra clave \_\_\_ se usa para llamar a un método de la clase superior.

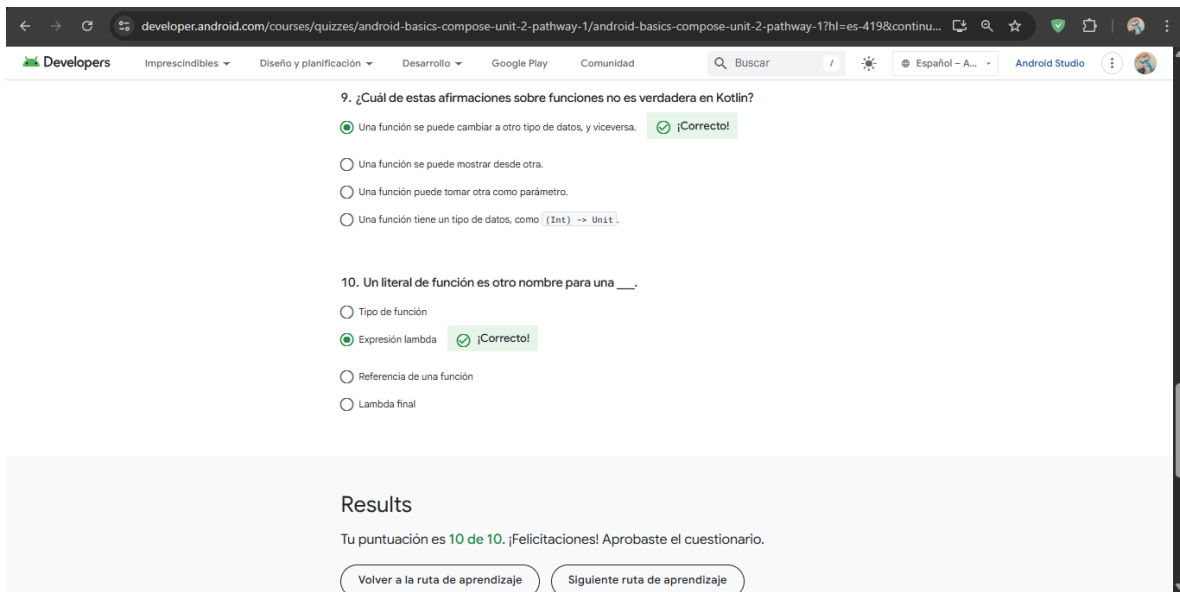
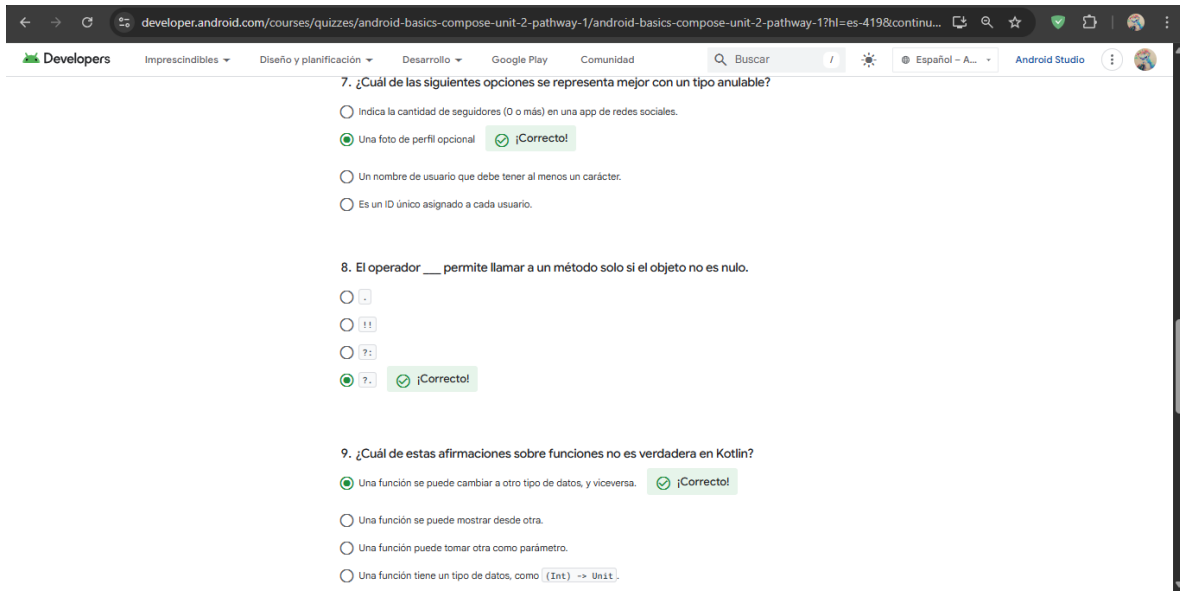
- ☐ this
- ☒ super ✅ ¡Correcto!
- ☐ parent
- ☐ self

6. Un(a) \_\_\_ define las propiedades o los métodos que una clase debe implementar.

- ☐ Delegar
- ☐ Tipo genérico
- ☒ interfaz ✅ ¡Correcto!
- ☐ Subclase

7. ¿Cuál de las siguientes opciones se representa mejor con un tipo anulable?

- ☐ Indica la cantidad de seguidores (0 o más) en una app de redes sociales.
- ☒ Una foto de perfil opcional ✅ ¡Correcto!
- ☐ Un nombre de usuario que debe tener al menos un carácter.
- ☐ Es un ID único asignado a cada usuario.



## Cuestionario U2- Agrega un botón a una app

Este cuestionario evalúa conocimientos fundamentales sobre Jetpack Compose y el uso del depurador en Android Studio. Aborda el uso de componentes componibles como Imagen y Columna, el manejo de estados con remember y mutableStateOf, y la alineación de elementos con Alignment.Center. También incluye preguntas sobre el funcionamiento del depurador, como la inspección de variables y la navegación en la pila de llamadas mediante la opción "Salir".


← → ↺ developer.android.com/courses/quizzes/android-basics-compose-unit-2-pathway-2/android-basics-compose-unit-2-pathway-2?continu... 📄 🔍 ☆ 🍏 📁 📄 📄 📄

Developers Imprescindibles ▾ Diseño y planificación ▾ Desarrollo ▾ Más ▾ 🔍 Buscar / 🌞 Español - A... ▾ Android Studio Acceder

## Cómo agregar un botón a una app

Hagamos una prueba rápida. Debes responder al menos 5 preguntas de manera correcta para aprobar el cuestionario.

Volver a la ruta de aprendizaje



1. Usa un elemento \_\_\_ componible para mostrar una imagen.

☐ Botón

☐ Texto

☒ Imagen ¡Correcto!

☐ Ícono

2. Alignment.Center centra los componentes de la IU de manera horizontal y vertical.

← → ↺ developer.android.com/courses/quizzes/android-basics-compose-unit-2-pathway-2/android-basics-compose-unit-2-pathway-2?continu... 📄 🔍 ☆ 🍏 📁 📄 📄 📄

Developers Imprescindibles ▾ Diseño y planificación ▾ Desarrollo ▾ Más ▾ 🔍 Buscar / 🌞 Español - A... ▾ Android Studio Acceder

2. Alignment.Center centra los componentes de la IU de manera horizontal y vertical.

☒ Verdadero ¡Correcto!

☐ Falso

3. Las funciones componibles pueden almacenar un objeto en la memoria con el elemento componible \_\_\_.

☒ recordar ¡Correcto!

☐ Columna

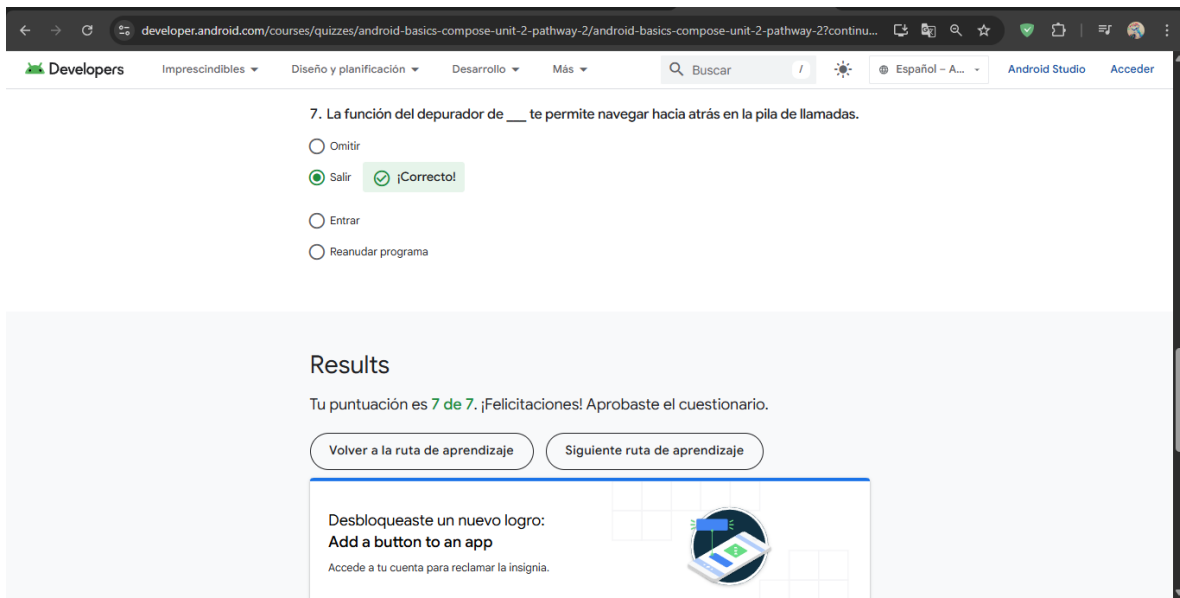
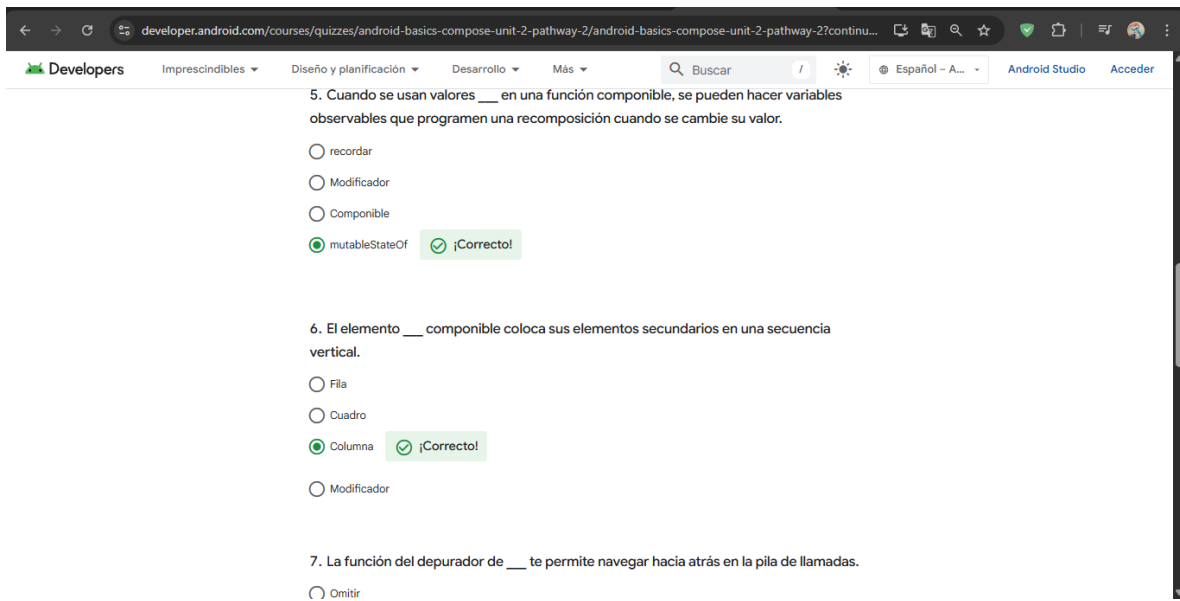
☐ Modificador

☐ Componible

4. El depurador te permite inspeccionar variables cuando se suspende la ejecución del código.

☒ Verdadero ¡Correcto!

☐ Falso



## Cuestionario U2- Interactúa con la IU y el estado

Este cuestionario trata sobre conceptos clave de Jetpack Compose relacionados con la composición, el estado y las pruebas. Explica cómo Compose ejecuta por primera vez los elementos componibles durante la composición inicial y cómo responde a los cambios mediante la recomposición. También aborda el estado, entendido como cualquier valor que puede cambiar con el tiempo, y el concepto de elevación de estado, que implica mover el estado hacia arriba para que un componente se vuelva sin estado. También incluye detalles prácticos como el uso de `onNext` para navegar entre campos de entrada, la función `kotlin.math.ceil()` para redondear valores, el uso del Inspector de diseño para analizar la UI en tiempo real, y las ubicaciones y anotaciones correctas para las pruebas de UI en Android.

developer.android.com/courses/quizzes/android-basics-compose-unit-2-pathway-3/android-basics-compose-unit-2-pathway-3?hl=es-419&c...

Developers Imprescindibles ▾ Diseño y planificación ▾ Desarrollo ▾ Más ▾

Buscar /

Español - A... ▾ Android Studio Acceder

TRAINING > ANDROID BASICS WITH COMPOSE > BUILDING APP UI > UI AND STATE

## Interactúa con la IU y el estado

Hagamos una prueba rápida. Debes responder al menos 7 preguntas de manera correcta para aprobar el cuestionario.

[Volver a la ruta de aprendizaje](#)

1. Jetpack Compose ejecuta tus elementos componibles por primera vez durante \_\_\_\_, y realizará un seguimiento de esos elementos que llamas para describir la IU.

☒ Composición inicial ☒ ¡Correcto!

☐ Recomposición

☐ Cambio de estado

☐ Cierre inesperado de la app

developer.android.com/courses/quizzes/android-basics-compose-unit-2-pathway-3/android-basics-compose-unit-2-pathway-3?hl=es-419&c...

Developers Imprescindibles ▾ Diseño y planificación ▾ Desarrollo ▾ Más ▾

Buscar /

Español - A... ▾ Android Studio Acceder

1. Jetpack Compose ejecuta tus elementos componibles por primera vez durante \_\_\_\_, y realizará un seguimiento de esos elementos que llamas para describir la IU.

☒ Composición inicial ☒ ¡Correcto!

☐ Recomposición

☐ Cambio de estado

☐ Cierre inesperado de la app

2. La única forma de modificar un objeto Composition es a través de la recomposición.

☒ Verdadero ☒ ¡Correcto!

☐ Falso

3. El/la \_\_ se genera cuando Jetpack Compose vuelve a ejecutar los elementos componibles que pueden haberse modificado en respuesta a los cambios de datos.

☐ Composición inicial

☒ Recomposición ☒ ¡Correcto!

developer.android.com/courses/quizzes/android-basics-compose-unit-2-pathway-3/android-basics-compose-unit-2-pathway-3?hl=es-419&c...

Developers Imprescindibles ▾ Diseño y planificación ▾ Desarrollo ▾ Más ▾ Buscar / Español - A... Android Studio Acceder

3. El/la \_\_\_ se genera cuando Jetpack Compose vuelve a ejecutar los elementos componibles que pueden haberse modificado en respuesta a los cambios de datos.

- ☐ Composición inicial
- ☒ **Recomposición** ¡Correcto!
- ☐ Cambio de estado
- ☐ Cierre inesperado de la app

4. \_\_\_ en una aplicación es cualquier valor que puede cambiar con el tiempo.

- ☒ **Estado** ¡Correcto!
- ☐ valor
- ☐ valueChange
- ☐ StateValue

5. \_\_\_ es un patrón en el que el estado se mueve hacia arriba para dejar a un componente sin estado.

- ☐ Cambio de estado

developer.android.com/courses/quizzes/android-basics-compose-unit-2-pathway-3/android-basics-compose-unit-2-pathway-3?hl=es-419&c...

Developers Imprescindibles ▾ Diseño y planificación ▾ Desarrollo ▾ Más ▾ Buscar / Español - A... Android Studio Acceder

5. \_\_\_ es un patrón en el que el estado se mueve hacia arriba para dejar a un componente sin estado.

- ☐ Cambio de estado
- ☒ **Elevación de estado** ¡Correcto!
- ☐ Composición elevada
- ☐ Recomendación

6. ¿Qué propiedad `KeyboardAction` se usa para mover el foco al siguiente elemento que admite composición?

- ☐ `onDone`
- ☒ **`onNext`** ¡Correcto!
- ☐ `onGo`
- ☐ `onSend`

7. ¿Cuál de las siguientes funciones de Kotlin se usa para redondear un double o float?

- ☐ `kotlin.math.ceilUp()`



developer.android.com/courses/quizzes/android-basics-compose-unit-2-pathway-3/android-basics-compose-unit-2-pathway-3?hl=es-419&c...

Developers Imprescindibles ▾ Diseño y planificación ▾ Desarrollo ▾ Más ▾ Buscar / Español - A... Android Studio Acceder

7. ¿Cuál de las siguientes funciones de Kotlin se usa para redondear un double o float?

☐ kotlin.math.ceilUp()

☒ kotlin.math.ceil() ¡Correcto!

☐ kotlin.math.roundDown()

☐ kotlin.math.roundUp()

8. El Inspector de diseño es una herramienta de Jetpack Compose que te permite inspeccionar un diseño de Compose dentro de una app en ejecución en un emulador o dispositivo físico.

☒ Verdadero ¡Correcto!

☐ Falso

9. Las pruebas de IU se almacenan en el directorio \_\_\_\_.

☐ principal

☒ androidTest ¡Correcto!

developer.android.com/courses/quizzes/android-basics-compose-unit-2-pathway-3/android-basics-compose-unit-2-pathway-3?hl=es-419&c...

Developers Imprescindibles ▾ Diseño y planificación ▾ Desarrollo ▾ Más ▾ Buscar / Español - A... Android Studio Acceder

9. Las pruebas de IU se almacenan en el directorio \_\_\_\_.

☐ principal

☒ androidTest ¡Correcto!

☐ prueba

☐ res

10. Las pruebas locales y de la IU deben anotarse con la anotación \_\_\_\_.

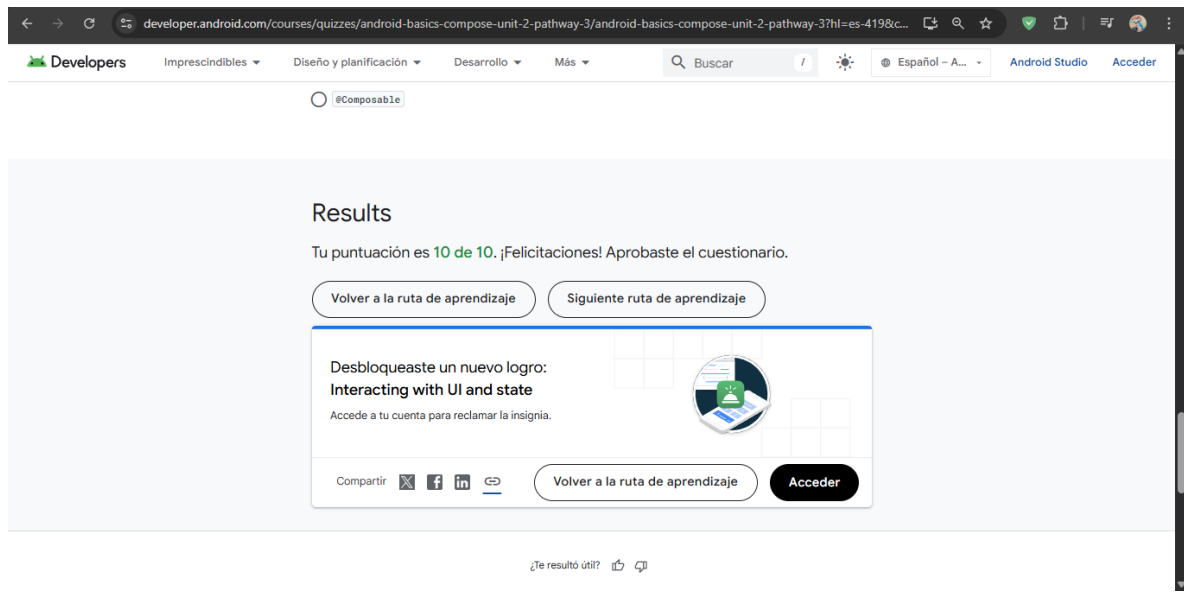
☐ @VisibleForTesting

☐ @Preview

☒ @Test ¡Correcto!

☐ @Composable

Results



**Nota:** Esta tarea ya había sido entregada anteriormente con tiempo, pero se modificó.