


# Cómo programar tareas con WorkManager

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

[Volver a la ruta de aprendizaje](https://developer.android.com/courses/pathways/android-basics-compose-unit-7-pathway-1) (https://developer.android.com/courses/pathways/android-basics-compose-unit-7-pathway-1)

1. ¿Qué herramienta te permite visualizar, supervisar y depurar los workers de tu app?

☐ Profiler

☒ Inspector de tareas en segundo plano  ¡Correcto!

☐ Logcat

☐ Administrador de dispositivos

2. ¿Cuáles de las siguientes opciones son estados de trabajo terminal válidos?

*Selecciona todas las respuestas que consideres correctas.*

☒ CANCELADA  ¡Correcto!


☐ BORRADA

☒ ERROR  ¡Correcto!

☒ COMPLETA  ¡Correcto!


3. ¿Cuáles de las siguientes opciones son tipos válidos de solicitudes de trabajo?

Selecciona todas las respuestas que consideres correctas.

☒ OneTimeWorkRequest  ¡Correcto!

☐ SingleWorkRequest

☐ RepeatingWorkRequest

☒ PeriodicWorkRequest  ¡Correcto!

4. La creación y puesta en cola de varias tareas dependientes y el orden en el que se deben ejecutar se denomina vinculación.


☐ Verdadero

☒ Falso  ¡Correcto!

5. ¿En cuál de las siguientes situaciones son útiles las restricciones de trabajo?

☐ Verificar que se guarde una forma de pago válida en el dispositivo del usuario antes de que se ejecute el trabajo

☐ Comprobar la hora antes de que se ejecute el trabajo

☒ Verificar que el dispositivo esté conectado a una red Wi-Fi antes de descargar una gran cantidad de datos de la app  ¡Correcto!

☐ Comprobar que la app se haya abierto una cantidad determinada de veces antes de que se ejecute el trabajo

6. ¿Cuál de las siguientes opciones es una manera de pasar datos de entrada a un worker?

☐ Pasa los datos como un argumento cuando llames a la función `doWork()`.

- ☒ Usar un objeto de datos para pasar pares clave-valor ☒ ¡Correcto!
- ☐ Pasar datos como una string, pero debe tener menos de 140 caracteres
- ☐ Asígnalo a la variable `worker.inputData`.

7. Después de poner el trabajo en cola, puedes comprobar su estado antes de \_\_\_\_.

*Selecciona todas las respuestas que consideres correctas.*

- ☒ El nombre ☒ ¡Correcto!
- ☒ El ID ☒ ¡Correcto!
- ☒ La etiqueta ☒ ¡Correcto!
- ☐ Tipo de trabajo

8. El Inspector de tareas en segundo plano te permite detener a los trabajadores durante su ejecución.

- ☐ Verdadero
- ☒ Falso ☒ ¡Correcto!

9. ¿Qué compilador de Worker se recomienda para probar elementos `CoroutineWorker`?

- ☐ `OneTimeWorkRequestBuilder`
- ☐ `PeriodicWorkRequestBuilder`
- ☐ `TestWorkerBuilder`
- ☒ `TestListenableWorkerBuilder` ☒ ¡Correcto!

10. Cuando pruebas las implementaciones de Worker, puedes llamar a los trabajadores directamente con `doWork()` en lugar de ponerlos en cola.

☒ Verdadero ☒ ¡Correcto!

☐ Falso

## Results

Tu puntuación es **10 de 10**. ¡Felicitaciones! Aprobaste el cuestionario.

¡Obtuviste la insignia **Cómo programar tareas con WorkManager!**

Se agregó la insignia a tu perfil.



Compartir



(https://developer.android.com/courses/quizzes/android-basics-compose-unit-7-pathway-1/android-basics-compose-unit-7-pathway-1?authuser=0&hl=es-419&continue=https%3A

**Volver a la ruta de aprendizaje** (https://developer.android.com/courses/

**Ver perfil** (https://developers.google.com/profile/u/11083234338959505892