

UF1:

1. ¿En que directorio de un proyecto Android crearemos la animaciones de vistas?
 1. res/animator
 2. res/drawable/anim
 3. **res/anim**
 4. res/animations
2. ¿Qué significan las siglas AVD?
 1. **Android Virtual Device**
 2. Android Visual Device
 3. Android Virtual Debug
 4. Android Virtual Debugger
3. El tipo de conexiones que hace referencia a las tecnologías wireless para iniciar comunicaciones a pocos centímetros de distancia se conoce como...:
 1. Bluetooth
 2. Telecom
 3. **NFC**
 4. WLAN
4. En Android Studio existe una funcionalidad que controla todo el mecanismo de compilación mediante unos ficheros de configuración. Se trata de...:
 1. APK
 2. **Gradle**
 3. Logcat
 4. AndroidManifest
5. Una de las herramientas que Android Studio usa para solucionar e identificar errores podemos encontrarla en las pestañas inferiores y muestra los errores del sistema y los mensajes que configuremos en nuestra APP. Esta herramienta se conoce como:
 1. Gradle
 2. error.xml
 3. AndroidManifest.xml
 4. **Logcat**
6. ¿En qué directorio podemos encontrar los ficheros de diseño de los layouts?
 1. /drawable/layout
 2. /java/layout
 3. /kotlin/layout
 4. **/res/layout**
7. ¿Qué permiso debemos declarar dentro del Android Manifest para escribir en el almacenamiento interno del dispositivo?
 1. WRITE_STORAGE
 2. WRITE_INTERNAL_STORAGE
 3. **No necesitamos ningún permiso para acceder a la memoria interna del dispositivo.**
 4. WRITE_EXTERNAL_STORAGE
8. ¿Qué es un servicio?
 1. Es un proceso ejecutado de forma invisible para el usuario,

- existe un único tipo en las aplicaciones Android.
2. Es un proceso ejecutado de forma invisible para el usuario, pero puede comunicarse con el.
 3. Es un proceso ejecutado de forma visible para el usuario, existen dos tipos diferentes.
 4. **Es un proceso ejecutado de forma invisible para el usuario, existen tres tipos diferentes**
9. ¿Qué dos métodos son obligatorios en la clase AsyncTask?
1. start() y onFinishExecute()
 2. onExecute() y onPostExecute()
 3. doInBackground() y start()
 4. **doInBackground() y onPostExecute()**
10. ¿Qué hace el código que se muestra a continuación?
- ```
val i = Intent (this,
Activity2::class.java);
i.putExtra("mensaje", "Esto es otro
mensaje para la Actividad 2");
i.putExtra("ID", 85);
startActivity(i);
```
1. La actividad principal que lo ejecuta recibe dos mensajes de la Actividad B, que tiene un identificador de proceso con ID 85
  2. **La actividad que lo invoca envía a la Actividad 2 un mensaje de texto y un dato numérico**
3. Ninguna de las opciones es correcta
  4. Envía dos mensajes de texto al proceso 85 que cargará la Actividad B

## UF2:

1. Para enviar y recibir señal de audio e imagen en tiempo real como en videollamadas o contenido en streaming como en radios online podemos usar el protocolo...
  1. RTP
  2. FTP
  3. MediaPlayer
  4. ExoPlayer
2. Indica cuál de los siguientes no es un servicio del sistema que puede ser usado por Android
  1. Search Service
  2. MediaPlayer Service
  3. Window Manager
  4. Activity Manager
3. Dentro de las clases que podemos usar en MediaPlayer indica cuál es de bajo nivel que destaca en la reproducción de fuentes de audio que ya han sido decodificadas
  1. SoundPool
  2. AudioTrack
  3. MediaService
  4. ToneGenerator
4. ¿En qué directorios podemos almacenar ficheros de sonido y vídeo para poder ser reproducidos desde el código de una activity?
  1. /layout/raw
  2. /res/raw
  3. /src/raw
  4. /res/raw
5. ¿Qué permiso debemos definir en Android Manifest para poder grabar una conversación telefónica?
  1. `<uses-permission android:name="android.permission.RECORD_CALL"/>`
  2. `<uses-permission android:name="android.permission.MICROPHONE"/>`
  3. `<uses-permission android:name="android.permission.RECORD_AUDIO"/>`
  4. `<uses-permission android:name="android.permission.RECORD"/>`
6. En Android podemos usar un elemento para compartir archivos de forma segura entre distintas aplicaciones. Se trata de...:
  1. Android Manifest
  2. Listener
  3. File Provider
  4. Intent
7. Las clases utilizadas para el uso de sensores se encuentran en el paquete android.hardware y son:
  1. Sensor, SensorEvent, SensorManager y SensorEventListener.

2. Sensor, SensorRotation, SensorOutputEvent y SensorEventListener.
3. SensorData, SensorEvent, SensorManager y SensorEventListener.
4. Sensor, SensorRotation, SensorManager y SensorEventListener.
8. Esta clase permite solapar múltiples fuentes de sonido que se inician al mismo tiempo, decidiendo en cada caso cuál se reproduce y cuál no
  1. FileProvider
  2. AsyncTask
  3. SoundPool
  4. AudioTrack
9. ¿Qué propiedad deberemos declarar en el Android Manifest para hacer uso de la cámara?
  1. `<uses-permission android:name="android.permission.CAM" />`
  2. `<permissions android:name="android.permission.CAMERA" />`
  3. `<uses-permission android:name="android.permission.CAMARA" />`
  4. `<uses-permission android:name="android.permission.CAMERA" />`
10. La herramienta ExoPlayer sirve para...:
  1. Reproducir sonidos que se encuentran en la memoria del dispositivo y han sido decodificados en PCM
  2. Reproducir tonos de pitido de bajo nivel
  3. Reproducir video y configurar los parámetros de la reproducción
  4. Reproducir audio ya codificado como MP3

### UF3:

1. Se conoce como OpenGL
  1. Es una de las librerías para gráficos de alto rendimiento 2D y 3D más importantes. Permite cargar una serie de recursos Drawable para crear una animación.
  2. Es la plantilla o lienzo que permite definir en las aplicaciones un control a nivel de interfaz de usuario.
  3. Es la propiedad que permite añadir a cualquier objeto una determinada animación mediante el uso de propiedades o estilos de programación.
2. Los componentes principales de un motor de juego son...
  1. Librerías, motor físico y motor de renderizado.
  2. Objetos, motor físico y motor gráfico.
  3. Funciones, motor físico y motor de renderizado.
  4. Atributos, motor físico y motor de renderizado.
3. Un collider son áreas que envuelven al GameObject, utilizados para:
  1. Añadirle movimiento por toda la escena a nuestros diferentes objetos del juego
  2. Detectar las colisiones entre los diferentes objetos del juego
  3. Añadirle físicas a los elementos del juego.
  4. Añadirle animación a nuestros elementos del juego.
4. ¿Cuáles son los tipos de colisionadores que existen?
  1. Character collider, Mesh collider y Sphere collider.
  2. Character collider, Box collider y Capsule Collider.
  3. Box collider, Capsule Collider, Mesh Collider y Sphere Collider
  4. Character collider, Cube collider, Sphere collider y Mesh Collider
5. La lógica del juego...
  1. Proporciona un framework potente que permite animar una gran cantidad de objetos y representar dichos objetos de una multitud de formas.
  2. Se encargará de procesar todos los eventos del usuario y dibujar continuamente la escena del juego
  3. Es la representación gráfica de una serie de rutinas de programación, que ofrecen al usuario un diseño en un escenario gráfico 2D o 3D
  4. Es el punto de interacción entre el usuario y el videojuego. Su

- objetivo es brindar la información necesaria para que el usuario interactúe con fluidez durante el juego.
6. Se conoce como Animadores:
    1. es la propiedad que permite añadir a cualquier objeto una determinada animación mediante el uso de propiedades o estilos de programación.
    2. es la plantilla o el lienzo que permite definir en las aplicaciones un control a nivel de interfaz de usuario
    3. es una de las librerías para gráficos de alto rendimiento 2D y 3D más importantes
    4. permite cargar una serie de recursos Drawable para crear una animación
  7. La Interfaz de usuario...
    1. proporciona un framework potente que permite animar una gran cantidad de objetos y representar dichos objetos de una multitud de formas
    2. es la representación gráfica de una serie de rutinas de programación, que ofrecen al usuario un diseño en un escenario gráfico 2D o 3D.
    3. se encargará de procesar todos los eventos del usuario y dibujar continuamente la escena del juego.
  4. es el punto de interacción entre el usuario y el videojuego. Su objetivo es brindar la información necesaria para que el usuario interactúe con fluidez durante el juego.
  8. El canvas ...
    1. Permite cargar una serie de recursos Drawable para crear una animación en Unity
    2. Es la plantilla o el lienzo que permite definir en las aplicaciones de Unity un control a nivel de interfaz de usuario
    3. Es una de las librerías para gráficos de alto rendimiento 2D y 3D más importantes
    4. Es la propiedad que permite añadir a cualquier objeto en Unity una determinada animación mediante el uso de propiedades o estilos de programación
  9. El motor de renderizado...
    1. Son todas aquellas librerías de las que se hace uso para el desarrollo de figuras, polígonos, luces y sombras, etc.
    2. Es el encargado de gestionar las colisiones, scripts de programación, sonidos.

3. Es el encargado de renderizar las animaciones de nuestro juego.
  4. Es el encargado de renderizar todas las texturas de un mapa, todos los relieves, suavizado de objetos, etc.
10. La interfaz que proporciona Unity posee varias ventanas, una de ellas es la ventana del inspector, la cual...
1. Permite visualizar y modificar las propiedades de un objeto
  2. se encuentran las funciones donde poder trabajar. En el centro los controles de play, pausa y pasos.
  3. permite la visualización de la escena.
  4. Permite visualizar assets