**CONSULTA TEORICA de recuperación…**

**formas de almacenar datos persistentes en Android?**

En Android, hay varias formas de almacenar datos persistentes. Las opciones más comunes son:

Shared Preferences: Este es un método simple para almacenar pequeñas cantidades de datos en pares clave-valor. Es útil para almacenar preferencias de usuario como configuraciones de aplicaciones.

Almacenamiento interno (Internal Storage): Se trata de almacenar archivos directamente en la memoria interna del dispositivo. Es útil para almacenar datos privados de la aplicación que no necesitan ser compartidos con otras aplicaciones o usuarios.

Almacenamiento externo (External Storage): Se trata de almacenar archivos en una tarjeta SD u otro dispositivo de almacenamiento externo. Es útil para almacenar archivos grandes o datos que necesitan ser compartidos con otras aplicaciones o usuarios.

Base de datos SQLite: Este es un sistema de gestión de bases de datos relacionales que se puede utilizar para almacenar grandes cantidades de datos de forma estructurada.

Content Providers: Los proveedores de contenido son componentes que permiten a una aplicación compartir datos con otras aplicaciones. Pueden ser útiles para almacenar y compartir grandes cantidades de datos con otras aplicaciones.

En resumen, las opciones de almacenamiento de datos persistentes en Android son múltiples y cada una tiene sus ventajas y desventajas. La elección del método adecuado depende del tipo de datos que se quieran almacenar y de cómo se quieren utilizar en la aplicación.

**¿Qué es el archivo AndroidManifest.xml en Android?**

El archivo AndroidManifest.xml es un archivo obligatorio en cualquier aplicación de Android. Se utiliza para definir la estructura y la configuración básica de una aplicación Android. Contiene información importante acerca de la aplicación, como su nombre, el icono que se muestra en la pantalla de inicio, los permisos que necesita para funcionar y las actividades que contiene.

En resumen, el archivo AndroidManifest.xml es un archivo esencial para el desarrollo de aplicaciones de Android, ya que permite al sistema operativo Android identificar y ejecutar correctamente la aplicación, y asegura que la aplicación tenga acceso a los recursos y permisos necesarios para su correcto funcionamiento.

**¿Qué es Android?**

Android es un sistema operativo móvil altamente personalizable y adaptable que ha ganado una gran popularidad en todo el mundo, y que ofrece una amplia gama de dispositivos móviles y herramientas de desarrollo para los desarrolladores de aplicaciones móviles y juegos.

**¿Qué es el operador Elvis (?:) en Kotlin?**

El operador Elvis (?:) en Kotlin es un operador que se utiliza para proporcionar un valor predeterminado cuando se intenta acceder a un valor nulo. Es una forma abreviada de escribir una expresión condicional que comprueba si una variable es nula y, si es así, proporciona un valor predeterminado.

**¿Qué es Kotlin?**

Kotlin es un lenguaje de programación de alto nivel, orientado a objetos y funcional que se ejecuta en la máquina virtual de Java (JVM) y que fue desarrollado por JetBrains, la empresa detrás de IntelliJ IDEA, uno de los entornos de desarrollo integrado (IDE) más populares para Java y que ofrece una amplia gama de características y capacidades para el desarrollo de aplicaciones móviles, web y de escritorio.

**¿Qué es un archivo de recursos en Android?**

En Android, un archivo de recursos es un archivo que se utiliza para almacenar datos y recursos que se utilizan en una aplicación móvil, como imágenes, diseños, strings, estilos, animaciones, sonidos, valores de configuración, entre otros.

los archivos de recursos en Android son archivos que se utilizan para almacenar datos y recursos que se utilizan en una aplicación móvil, y que permiten separar los datos y recursos de la lógica de la aplicación para una gestión y mantenimiento más fácil del código.

**¿Qué es un tipo de dato en Kotlin?**

En Kotlin, un tipo de dato se refiere a la naturaleza de los datos que se pueden almacenar en una variable o utilizar en una expresión. Los tipos de datos en Kotlin pueden ser de dos tipos: tipos primitivos y tipos de objeto.

Los tipos de datos primitivos son aquellos que representan valores simples, como números enteros, números en coma flotante, caracteres y booleanos. En Kotlin, los tipos de datos primitivos incluyen:

Byte: representa un número entero de 8 bits.

Short: representa un número entero de 16 bits.

Int: representa un número entero de 32 bits.

Long: representa un número entero de 64 bits.

Float: representa un número en coma flotante de 32 bits.

Double: representa un número en coma flotante de 64 bits.

Char: representa un carácter Unicode de 16 bits.

Boolean: representa un valor booleano (verdadero o falso).

Los tipos de datos de objeto, por otro lado, son aquellos que representan objetos y se utilizan para almacenar valores más complejos, como cadenas de caracteres, arreglos y listas. En Kotlin, los tipos de datos de objeto incluyen:

String: representa una cadena de caracteres.

Array: representa un arreglo de elementos de un tipo de datos determinado.

List: representa una lista de elementos de un tipo de datos determinado.

Map: representa un mapa de valores asociados con claves.

Kotlin también proporciona la capacidad de definir tipos de datos personalizados mediante la creación de clases y enumeraciones.

En resumen, los tipos de datos en Kotlin se refieren a la naturaleza de los datos que se pueden almacenar en una variable o utilizar en una expresión, y pueden ser de dos tipos: tipos primitivos y tipos de objeto. Los tipos de datos primitivos representan valores simples, mientras que los tipos de datos de objeto representan objetos y se utilizan para almacenar valores más complejos.

**¿Qué es un RecyclerView en Android?**

el RecyclerView es un componente de la interfaz de usuario en Android que se utiliza para mostrar grandes conjuntos de datos de una manera eficiente. El RecyclerView utiliza la reciclaje de vistas para mejorar el rendimiento de la aplicación y reduce la cantidad de memoria necesaria para mostrar grandes conjuntos de datos. Además, el RecyclerView ofrece una amplia gama de opciones de personalización y es uno de los componentes más populares y útiles en el desarrollo de aplicaciones Android.

**¿Qué es un Intent en Android?**

un Intent es un objeto que se utiliza en Android para comunicar información entre componentes de una aplicación y entre diferentes aplicaciones. Los Intents se componen de una acción y datos, y se utilizan para realizar tareas como abrir una nueva actividad, compartir información con otras aplicaciones y realizar transmisiones de difusión. Los Intents son una parte importante del modelo de componentes de Android y permiten que las aplicaciones se comuniquen entre sí de manera efectiva.

**¿Cuál es el componente principal de una aplicación de Android?**

El componente principal de una aplicación de Android es la actividad (Activity). Una actividad es un componente de la interfaz de usuario que representa una sola pantalla con la que el usuario puede interactuar. Las actividades se utilizan para crear la interfaz de usuario de una aplicación de Android y para gestionar las interacciones del usuario con la aplicación.

**¿Qué es una variable en Kotlin?**

una variable en Kotlin es un espacio en memoria que se utiliza para almacenar un valor. Las variables tienen un nombre, un tipo de datos y un valor, y se pueden declarar como variables mutables o inmutables utilizando las palabras clave "var" y "val".

**¿Qué es el nullsafety en Kotlin?**

el null safety es una característica de Kotlin que ayuda a prevenir los errores de referencia nula de variables. En Kotlin, se debe especificar explícitamente si una variable puede ser nula o no, lo que ayuda a garantizar que las variables se inicialicen correctamente y se eviten errores comunes de referencia nula en tiempo de ejecución. Además, Kotlin proporciona una sintaxis concisa y segura para trabajar con variables que pueden ser nulas.

**¿Qué es un Bundle en Android?**

un Bundle en Android es un objeto que se utiliza para pasar datos entre componentes de una aplicación. Un Bundle es una estructura de datos que contiene un conjunto de pares clave-valor, donde la clave es una cadena y el valor puede ser cualquier objeto serializable. Los Bundles se utilizan con frecuencia para pasar información entre actividades o fragmentos en Android.

**¿Qué significa "SDK" en Android?**

el SDK en Android es un conjunto de herramientas de desarrollo que incluye bibliotecas, herramientas de compilación, documentación y ejemplos de código que se utilizan para desarrollar aplicaciones para la plataforma Android. El SDK de Android es una parte fundamental del desarrollo de aplicaciones para dispositivos Android y proporciona a los desarrolladores todo lo que necesitan para crear, depurar y probar aplicaciones de manera eficaz.