

# Bloque 3. Ejercicio Propuesto Fragment Dialogo Personalizado

[Descargar este ejercicio](#)

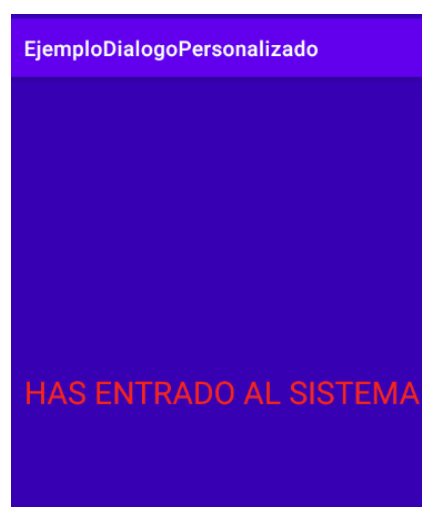
Crea una aplicación que al iniciarse muestre el Dialogo que ya creamos con anterioridad en el Bloque 6, pero ahora lo haremos mediante **DialogFragment**.



En este caso vamos a dar funcionalidad a los dos botones.

👉 Se tendrá que controlar que el usuario y la contraseña no están vacíos para poder cambiar de fragment.

Al pulsar Entrar se cargará un **FragmentEntrar** con el siguiente aspecto:



Y al pulsar Crear Cuenta se cargará el **FragmentCrearCuenta** con el siguiente aspecto.

Tendremos que tener la clase pojo Usuario que tendrá las tres propiedades *nombre*, *contraseña* (Strings) y *guardar* (Boolean). La salida que se muestra en el texto del fragment, es la llamada al

toString de esta clase (la contraseña se genera con la cantidad de caracteres introducidos en el password del dialogo, el mensaje se mostrará controlando si se ha pulsado o no el CheckBox)

#### EjemploDialogoPersonalizado

Login: Pedro  
Password: \*\*\*\*\*  
Activado recuerdo de datos



👉 Todos los fragments se tendrán que cargar desde la main, por lo que **para controlar los eventos** producidos en el Dialogo, lo más conveniente es usar **comunicación por Interfaces** (Se puede crear una enumeración con los nombres de los fragments y usar esta como parámetro de la interface, de esta forma podremos controlar a que fragment queremos ir). Por otro lado, **para pasar la información** desde el dialogo al FragmentCrearCuenta, se deberá usar un **ViewModel** con un LiveData de tipo Usuario.

💡 Puedes cargar el fragment **DialogoPersonalizado** cuando se pulsa el **BackPressed**, de esta forma podrás pasar de un fragment a otro sin tener que ejecutar cada vez la app.