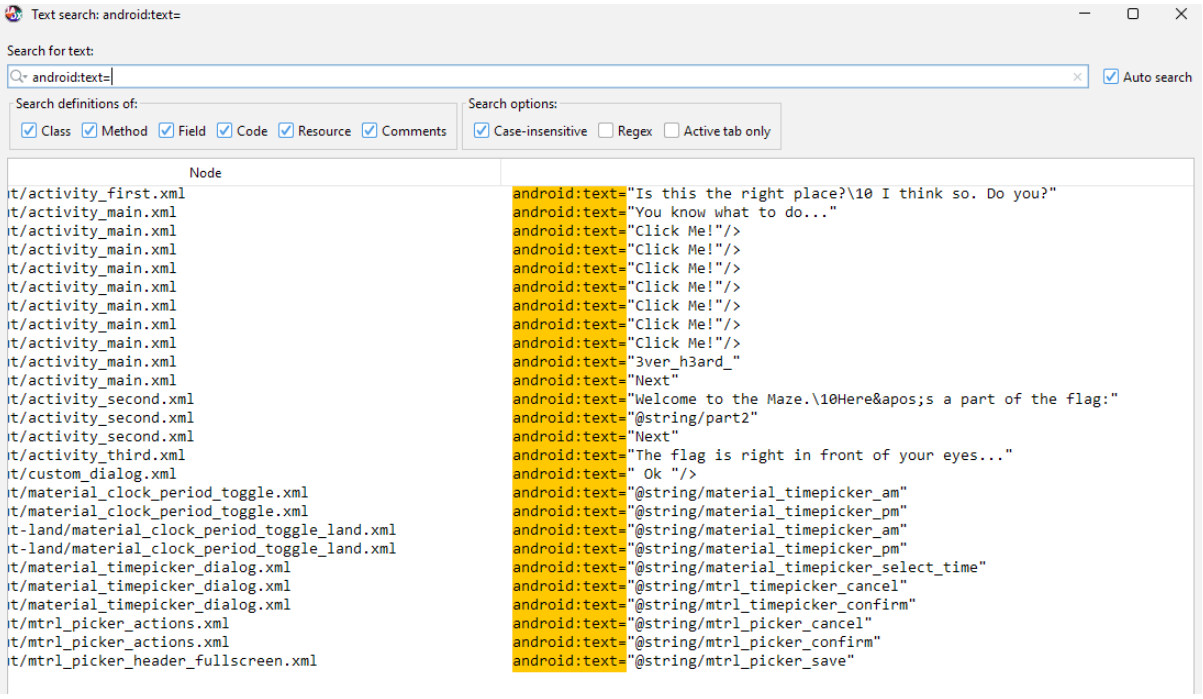
Treasure Hunt

Use JADX to reverse engineer an Android application and extract hidden information from the application's resources. Searched for the text android:text= within the app's decompile code. This search returned multiple key, but two key will be higlight:

- The string **3ver_h3ard**.
- The reference **@string/part2**, which points to another resource string within the app.

The **@string/part2** reference links to a file located at **res/values/strings.xml**.



Text search: android:text=

Search for text:

android:text=

☑ Auto search

Search definitions of: ☑ Class ☑ Method ☑ Field ☑ Code ☑ Resource ☑ Comments

Search options: ☑ Case-insensitive □ Regex □ Active tab only

```
                   Node
ıt/activity_first.xml                              android:text="Is this the right place?\10 I think so. Do you?"
ıt/activity_main.xml                               android:text="You know what to do..."
ıt/activity_main.xml                               android:text="Click Me!"/>
ıt/activity_main.xml                               android:text="Click Me!"/>
ıt/activity_main.xml                               android:text="Click Me!"/>
ıt/activity_main.xml                               android:text="Click Me!"/>
ıt/activity_main.xml                               android:text="Click Me!"/>
ıt/activity_main.xml                               android:text="Click Me!"/>
ıt/activity_main.xml                               android:text="Click Me!"/>
ıt/activity_main.xml                               android:text="3ver_h3ard_"
ıt/activity_main.xml                               android:text="Next"
ıt/activity_second.xml                             android:text="Welcome to the Maze.\10Here&apos;s a part of the flag:"
ıt/activity_second.xml                             android:text="@string/part2"
ıt/activity_second.xml                             android:text="Next"
ıt/activity_third.xml                              android:text="The flag is right in front of your eyes..."
ıt/custom_dialog.xml                               android:text=" Ok "/>
ıt/material_clock_period_toggle.xml                android:text="@string/material_timepicker_am"
ıt/material_clock_period_toggle.xml                android:text="@string/material_timepicker_pm"
ıt-land/material_clock_period_toggle_land.xml      android:text="@string/material_timepicker_am"
ıt-land/material_clock_period_toggle_land.xml      android:text="@string/material_timepicker_pm"
ıt/material_timepicker_dialog.xml                  android:text="@string/material_timepicker_select_time"
ıt/material_timepicker_dialog.xml                  android:text="@string/mtrl_timepicker_cancel"
ıt/material_timepicker_dialog.xml                  android:text="@string/mtrl_timepicker_confirm"
ıt/mtrl_picker_actions.xml                         android:text="@string/mtrl_picker_cancel"
ıt/mtrl_picker_actions.xml                         android:text="@string/mtrl_picker_confirm"
ıt/mtrl_picker_header_fullscreen.xml               android:text="@string/mtrl_picker_save"
```

Next, navigated to the file ***res/layout/activity_main.xml***, where the string ***3ver_h3ard*** is used. Finally, went to the ***res/values/strings.xml*** file, where the reference ***@string/part2*** led us to the string 0f_4ndro1d_r3v?. This string appears to be part of a hidden or encoded message related to the app's functionality or a challenge. Combining both flag will get **3ver_h3ard_0f_4ndro1d_r3v**

Flag: ironCTF{3ver_h3ard_0f_4ndro1d_r3v}