

@id v.s. @+id in Android Studio

Intro

In xml file, id (stands for identity) is unique so that it can identifies the node. For example, in Android Studio, one wants access a node where id = `list_entry` in `list_example.xml`. One can create `list_example.xml` in `.../res/layout/list_example.xml`. Then create a node with attribute android:id as “@+id/list_entry”.

A simple example here.

In `.../res/layout/list_example.xml`,

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content" >
    <ListView
        android:id="@+id/list_entry"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content" />
</LinearLayout>
```

What is R.java.

The file R.java contains all id that defined by yourself, or built-in id (i.e. id that is defined by Android Studio developers). Resource ID that can be accessed through R.id is generated by latest R.java file, according to comment in line 24

```
// Resources ID generated in the latest R.java for framework.
```

in this file

AndroidMain/androidx/compose/ui/graphics/vector/compat/AndroidVectorResources.android.kt

Use case of plus sign +

The plus sign + will define a new entry for framework. For precisely said, the plus sign + will create a new entry in R.java. One can access id programmatically where id is in R.java through (for example R.id.list_entry) .

@android:id	@id	@+id
Android resources, which are already defined in Android system, will be referred.	resources that you have defined/created in your project.	Create a new entry in resource that has NOT been defined yet.
none	none	Uniqueness: It can be defined twice.
Only used for id defined by Android System	Only used for built-in id.	Can NOT defined that built-in id.

See Also

Concept of R.java:

[android - What is the concept behind R.java? - Stack Overflow](#)

Ref

[Difference between "@id/" and "@+id/" in Android - Stack Overflow](#)