

1- Adding the action bar



Adding action buttons:

1. Create new project "My ApplicationLab2"
2. In res : Create a new android resource directory, resource type (menu).
3. In res/menu/ : Create a new menu resource file.
4. In res/menu/main_menu.xml : Add buttons you want

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">

// to add search button in action bar
```

```
<item android:id="@+id/action_search"
      android:icon="@drawable/ic_action_search"
      android:title="@string/action_search"
      app:showAsAction="ifRoom" />
```

```
// to add setting button in action bar
```

```
<item android:id="@+id/action_settings"
      android:title="@string/action_settings"
      app:showAsAction="never" />
```

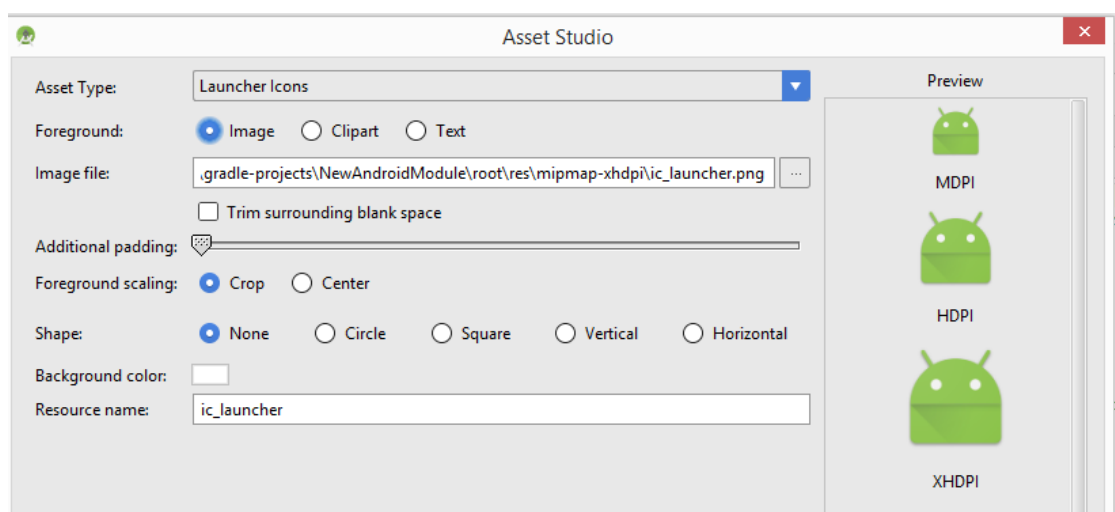
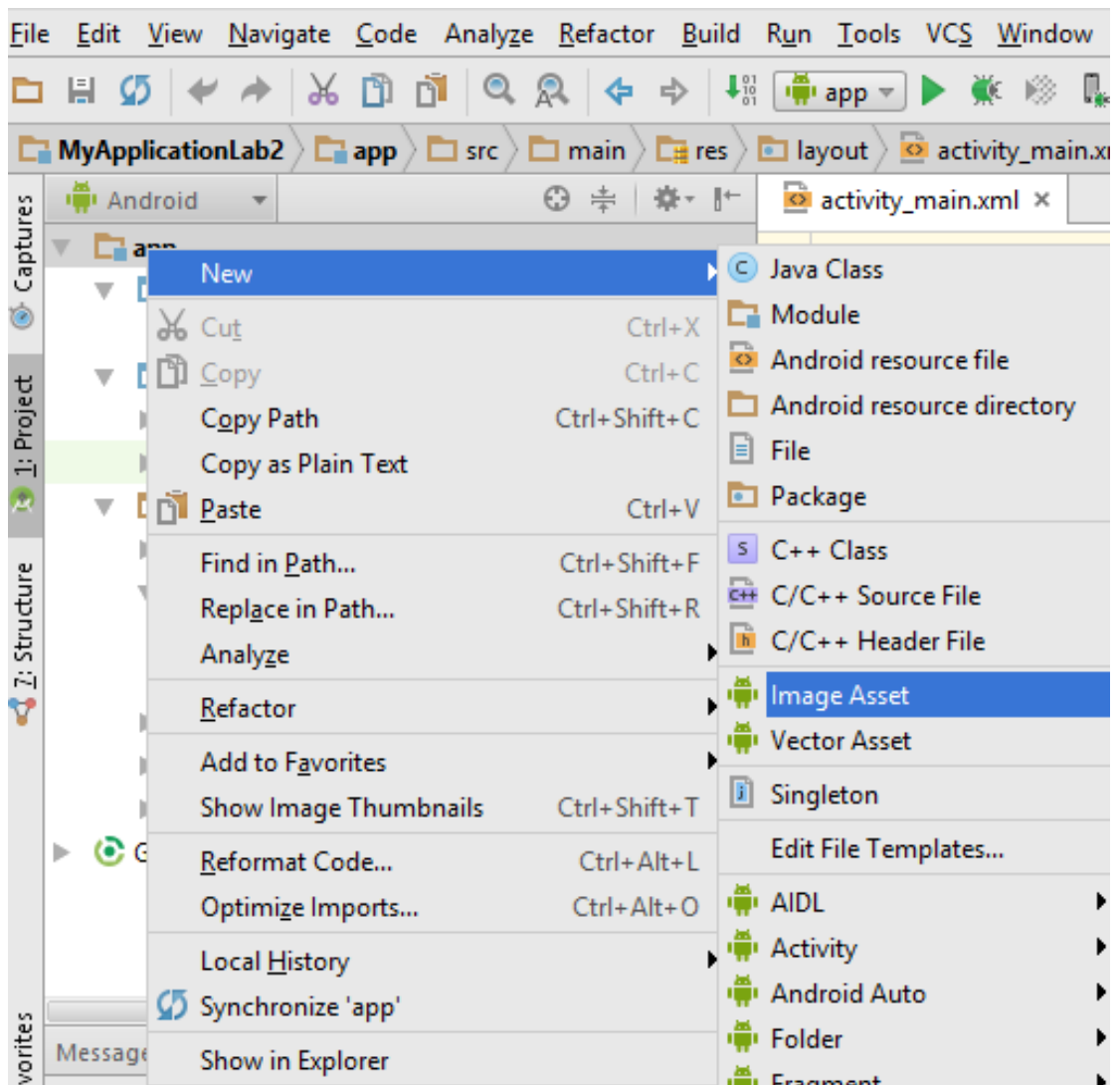
```
</menu>
```

5. In res/values/strings.xml :

```
<resources>
  <string name="app_name">My ApplicationLab2</string>
  <string name="action_search">Search</string>
  <string name="action_settings">Setting</string>
</resources>
```

6. Add ic action search image in res/drawable/

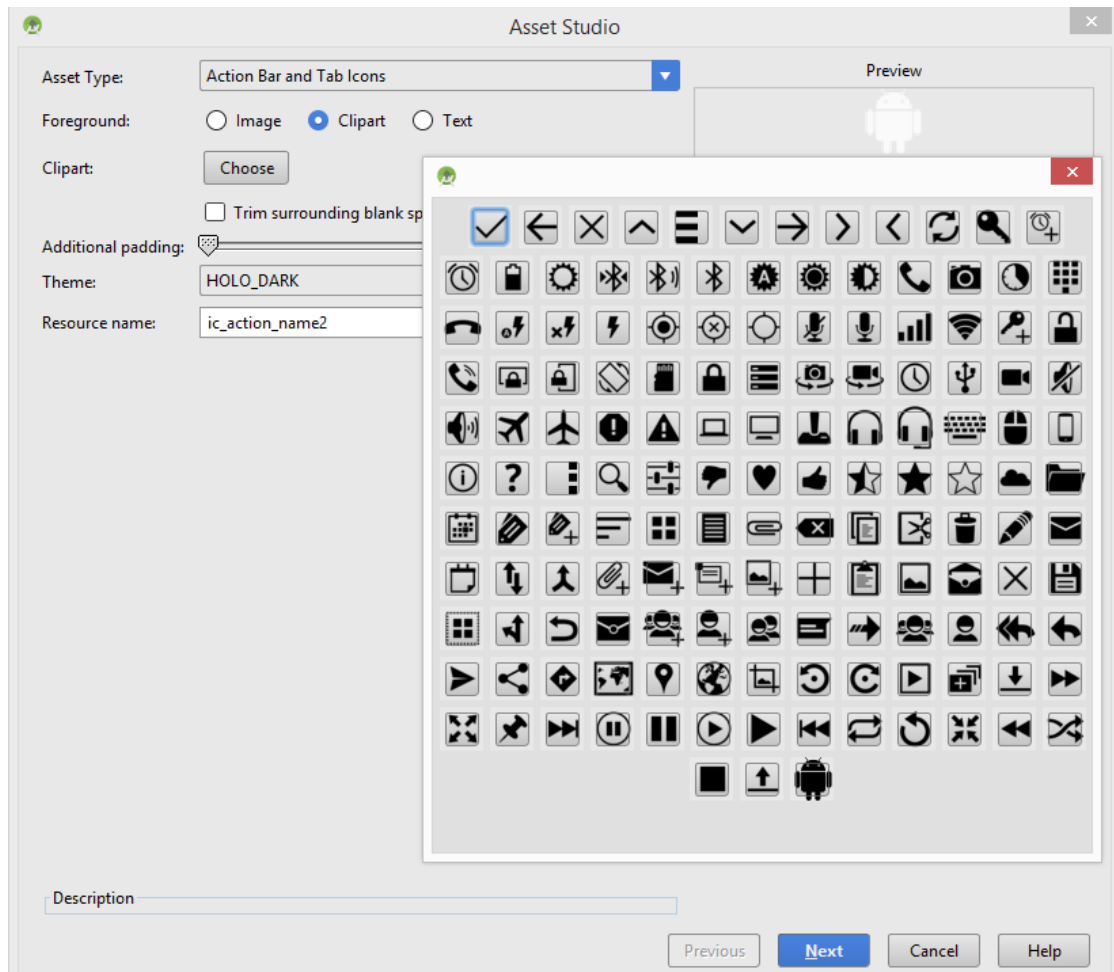
Or click right on app, new, Image Asset



From Asset type => choose **Action Bar and Tab icons**

From Foreground => choose **Clipart**

From Clipart => Click **Choose**



Then choose the image you want and click next then finished

7. In `mainactivity.java` : need to override `onCreateOptionsMenu` , To add buttons to the action bar

```
@Override
public boolean onCreateOptionsMenu(Menu menu)
{
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.main_menu, menu);
    return super.onCreateOptionsMenu(menu);
}
```

8. In `mainactivity.java` : need to override `onOptionsItemSelected` , To respond to action buttons

```
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId())
    {
        case R.id.action_search:
            openSearch();
            return true;
        case R.id.action_settings:
            openSettings();
            return true;
        default:

```

```

        return super.onOptionsItemSelected(item);
    }

    public void openSearch()
    {
        Toast.makeText(MainActivity.this, "search", Toast.LENGTH_LONG).show();
    }

    public void openSettings()
    {
        //
    }

```

2- Adding Up Button for Low-level Activities

- 1- Create new activity "ChildActivity "
- 2- In mainactivity.java :

```

    public void openSettings()
    {
        Intent intent= new Intent(this,ChildActivity.class);
        startActivity(intent);
    }

```

- 3- In Androidmanifest.xml

```

<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.hendpc.myapplicationlab2">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".ChildActivity"
            android:parentActivityName=".MainActivity">
        </activity>
    </application>

</manifest>

```

- 4- In ChildActivity : Enable the up button

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_child);
    getSupportActionBar().setDisplayHomeAsUpEnabled(true);
}

```

3- Supporting different languages

Make copy from strings.xml in res/values/

For example:

- English (default locale), /values/strings.xml:

```
<resources>
    <string name="app_name">My ApplicationLab2</string>
    <string name="Hello_World">Hello World</string>
</resources>
```

- Spanish, /values /strings-es.xml:

```
<resources>
    <string name="app_name"> Mi Aplicación2 </string>
    <string name="Hello_World">Hola Mundo </string>
</resources>
```

- French, /values /strings-fr.xml:

```
<resources>
    <string name="app_name"> Mon Application</ </string>
    <string name="Hello_World"> Bonjour le monde </string>
</resources>
```

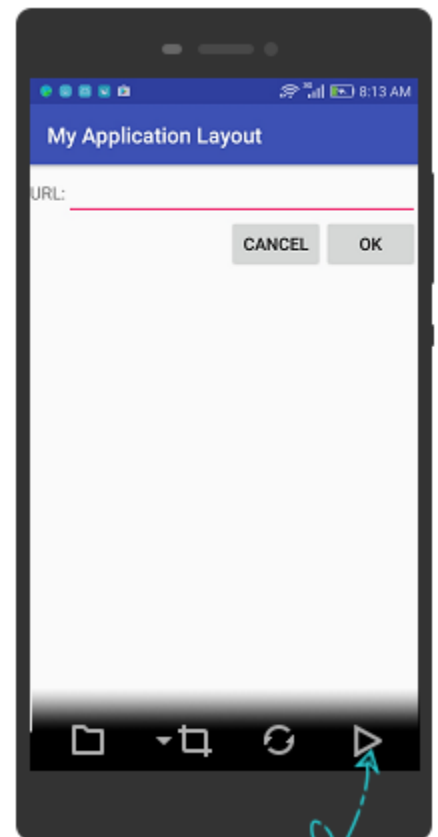
4- Examples of Relative / Linear layout

1-

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout

xmlns:android="http://schemas.android.com/apk/res/an
droid"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content">

    <TextView android:id="@+id/label"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="URL: "
        android:layout_alignBaseline="@+id/entry"
        android:layout_alignParentLeft="true"/>
    <EditText
        android:id="@+id/entry"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_toRightOf="@+id/label"
        android:layout_alignParentTop="true"/>
    <Button
        android:id="@+id/ok"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/entry"
        android:layout_alignRight="@+id/entry"
        android:text="OK" />
    <Button
        android:id="@+id/cancel"
```



```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_toLeftOf="@id/ok"
        android:layout_alignTop="@id/ok"
        android:text="Cancel" />
    </RelativeLayout>

```

2-

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout

xmlns:android="http://schemas.android.com/a
pk/res/android"

xmlns:tools="http://schemas.android.com/too
ls"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
    <Button
        android:text="I AM BIG"
        android:textSize="120dip"
        android:textStyle="bold"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
    />
    <Button
        android:text="I am small"
        android:layout_width="wrap_content"

    android:layout_height="wrap_content"

    android:layout_centerInParent="true" />
</RelativeLayout>

```

