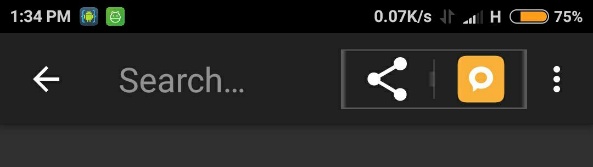
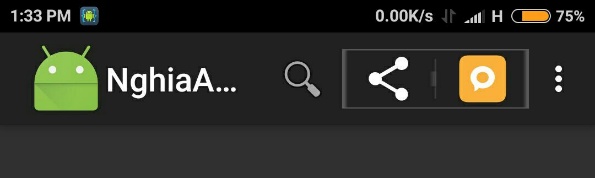
**ACTIONBAR**

**ActionBar là gì?**

Phần trên giao diện hiển thị Tên Activity, Icon Activity, Menu item, Main menu.



**Tạo menu trong ActionBar, sẽ tự hiển thị menu: menu/menu\_main.xml**

<menu xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

tools:context=".MainActivity">

<!--Search icon-->

<item

android:id="@+id/action\_search"

android:title="search"

android:icon="@android:drawable/ic\_menu\_search"

android:showAsAction="ifRoom|collapseActionView"

android:actionViewClass="android.widget.SearchView"/>

<!--Share icon-->

<item

android:id="@+id/action\_share"

android:title="Share"

android:actionProviderClass="android.widget.ShareActionProvider"

android:showAsAction="always"/>

<item

android:id="@+id/action\_settings"

android:orderInCategory="100"

android:title="@string/action\_settings"

android:showAsAction="never" />

</menu>

Thuộc tính:

* android:actionViewClass: Item sẽ hiển thị theo quy định của lớp.
  + android.widget.ShareActionProvider: Item kiểu chia sẽ.
  + android.widget.SearchView: Item kiểu tìm kiếm.
* android:showAsAction: Cách hiển thị của Item trên ActionBar
  + ifRoom: Nếu có chỗ trống.
  + collapseActionView: Đóng lại khi chọn Item khác (Search Item).
  + withText: Hiện cả Text (Title).
  + always: Luôn hiện.
  + never: Không hiện, chỉ hiện trong menu.

**Sử dụng Menu trong Activity:**

Bắt sự kiện Item Search, Item Share trong Activity:

@Override

public boolean onCreateOptionsMenu(Menu menu) {

getMenuInflater().inflate(R.menu.menu\_main, menu);

//---- bắt sự kiện của Item Search ---

MenuItem searchItem = menu.findItem(R.id.action\_search);

SearchView searchView = (SearchView) searchItem.getActionView();

searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {

@Override

public boolean onQueryTextSubmit(String query) {

//Khi ấn Enter

return false;

}

@Override

public boolean onQueryTextChange(String newText) {

//Khi đang gõ văn bản

Toast.makeText(getBaseContext(), newText, Toast.LENGTH\_SHORT).show();

return false;

}

});

//--- Bắt sự kiện Item Share ---

MenuItem item = menu.findItem(R.id.action\_share);

ShareActionProvider myShareActionProvider = (ShareActionProvider) item.getActionProvider();

Intent myIntent = new Intent();

myIntent.setAction(Intent.ACTION\_SEND);

myIntent.putExtra(Intent.EXTRA\_TEXT, "Đây là đoạn text tôi muốn chia sẻ");

myIntent.setType("text/plain");

myShareActionProvider.setShareIntent(myIntent);

return true;

}

Bắt sự kiện chọn ItemMenu trong Activity:

@Override

public boolean onOptionsItemSelected(MenuItem item) {

int id = item.getItemId();

if (id == R.id.action\_settings) {

//Chọn Item Setting trong Menu

return true;

}

return super.onOptionsItemSelected(item);

}

**Sử lý ActionBar trong Activity:**

/\*\* Gọi ActionBar để sử dụng \*/

ActionBar actionBar = getActionBar();

if (actionBar != null) {

//--- Activity Icon---

//Hiển thị Icon Home (Icon Activity)

actionBar.setDisplayOptions(ActionBar.DISPLAY\_SHOW\_HOME | ActionBar.DISPLAY\_SHOW\_TITLE);

//Thay Icon cho Icon Home

actionBar.setIcon(getResources().getDrawable(android.R.drawable.ic\_popup\_sync));

//--- Nút quay lại ----

// hiển thị Nút quay lại (<-)

actionBar.setDisplayHomeAsUpEnabled(true);

// hiển thị Nút Icon thay Nút quay lại

actionBar.setHomeAsUpIndicator(android.R.drawable.star\_off);

}

**Thay MenuItem bằng 1 Layout:**

Tai MenuItem chú ý thuộc tính actionProviderClass là tên lớp ta sẽ Inflate layout vào

<menu ... >

<item ...

app:showAsAction="always"

app:actionProviderClass="com.example.mapdemo.nghiaactionbar\_customitem.SearchActionProvider"/>

</menu>

Tạo lớp SearchActionProvider.java để Inflate Layout. Activity không làm gì cả.

public class SearchActionProvider extends ActionProvider {

private final Context mContext;

/\*\* Khởi tạo với Context \*/

public SearchActionProvider(Context context) {

super(context);

mContext = context;

}

@Override

public View onCreateActionView() {

// Gọi Layout "custom\_search" vào

LayoutInflater layoutInflater = LayoutInflater.from(mContext);

View view = layoutInflater.inflate(R.layout.custom\_search, null);

// Sử dụng Button trong Layout

Button button = (Button) view.findViewById(R.id.button);

button.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

// Do something...

Toast.makeText(mContext, "Click", Toast.LENGTH\_SHORT).show();

}

});

return view;

}

}

**Thay 1 Menu Item bằng 1 Layout, code ở Activity**

private MenuItem refreshMenuItem;

@Override

public boolean onOptionsItemSelected(MenuItem item) {

// Chọn Id

switch (item.getItemId()) {

case R.id.action\_refresh:

//--- refresh đổi MenuItem thành layout ---

refreshMenuItem = item;

new SyncData().execute();

return true;

}

}

/\*\* Chuyển icon thành hình ProgressBar \*\*/

private class SyncData extends AsyncTask<String, Void, String> {

@Override

protected void onPreExecute() {

// Thêm & hiển thị 1 Layout tại MenuItem

refreshMenuItem.setActionView(R.layout.action\_progressbar);

refreshMenuItem.expandActionView();

}

@Override

protected String doInBackground(String... params) {

// Đổi lại thành Menu Item cũ sau 3s

try {

Thread.sleep(3000);

} catch (InterruptedException e) {

e.printStackTrace();

}

return null;

}

@Override

protected void onPostExecute(String result) {

refreshMenuItem.collapseActionView();

// remove the progress bar view

refreshMenuItem.setActionView(null);

}

};