



# LẬP TRÌNH DI ĐỘNG

---

## Bài 5: menu và một số loại menu thường dùng



# Nội dung

---

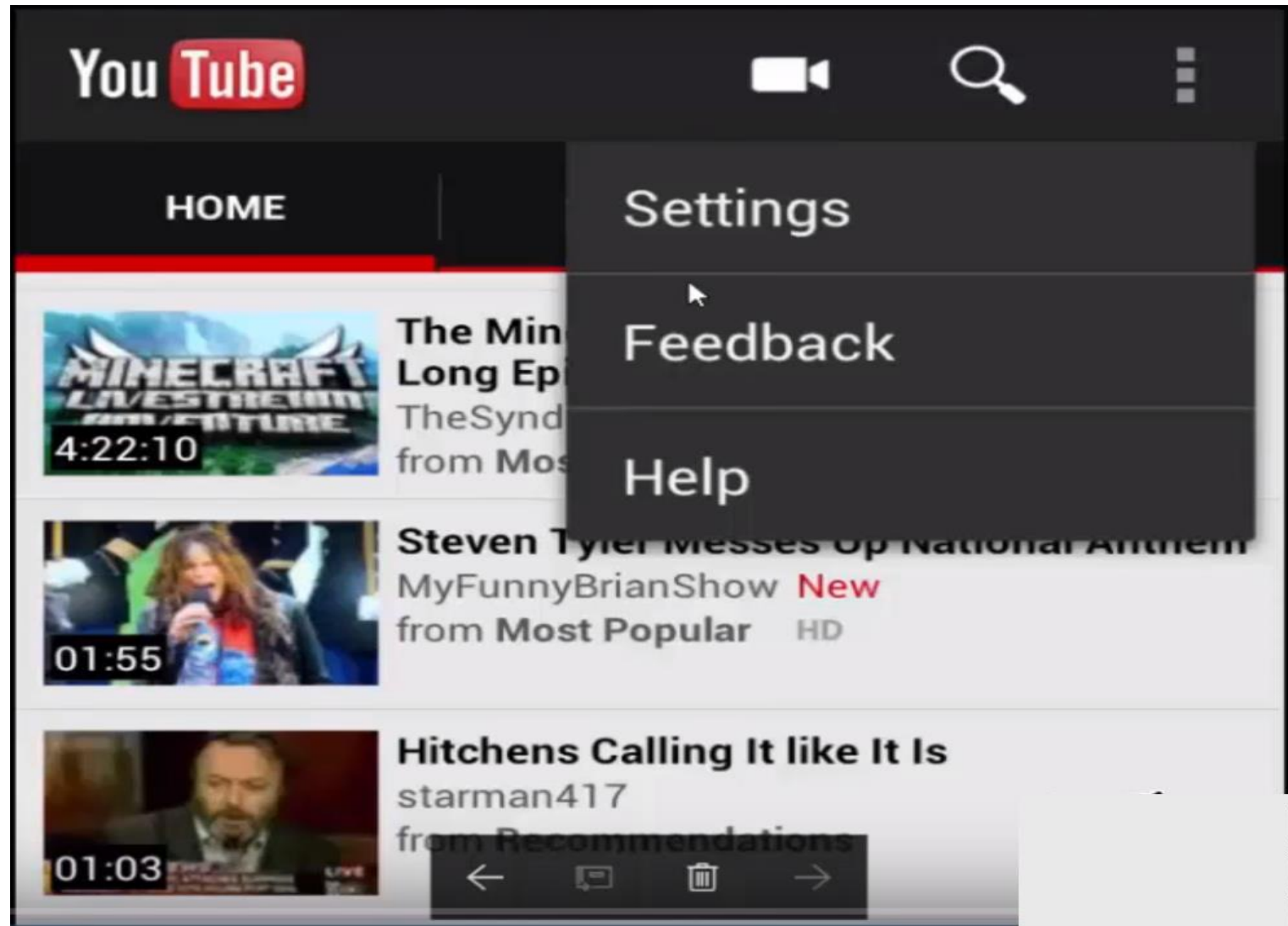
1. Menu
2. Popup menu
3. Context menu
4. Alert dialog
5. Dialog custom



Phần 1

# Khái niệm Menu

# Menu





# Khởi tạo menu

---

- Tạo một layout để chứa menu
  - Tạo thư mục (new directory) menu
  - Right click menu tạo Menu resource file
  - Trong menu resource file chứa tầng cặp thẻ <item>
- Các thuộc tính cơ bản của menu
  - Title: hiển thị nhãn menu(bắt buộc)
  - Icon="**@android:drawable/ic\_menu\_add**":đưa menu ra ngoài làm biểu tượng
  - showAsAction="**always**": luôn hiển thị biểu tượng



# Khởi tạo menu

---

- Các thuộc tính cơ bản của menu
  - `orderInCategory` = **"1"**: hiện thị icon theo thứ tự 1(mặc định 0)
  - Nếu không để `orderInCategory` thì hiện thị theo thứ tự khai báo



# Khởi tạo menu

---

- `<menu>`
- `<item android:title="add" android:id="@+id/menuAdd"`
- `android:icon="@android:drawable/ic_menu_add"`
- `app:showAsAction="always">`
- `</item>`
- `<item`
- `android:id="@+id/menuSetting"`
- `android:orderInCategory="1"`
- `android:title="Settings">`
- `</item>`



# Khởi tạo menu

---

```
■      <item
■          android:orderInCategory="4"
■          android:id="@+id/menuContact"
■          android:title="Contact">
■      <menu>
■          <item
■              android:id="@+id/menuPhone"
■              android:title="phone"></item>
■          <item>...</item>
■      </menu>
■  </item>
■  </menu>
```



# Đưa menu ra giao diện(tạo menu)

---



## ■ Cài đặt hàm onCreateOptionsMenu

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



# Bắt sự kiện cho menu

---

- Khai báo id cho từng item menu
  - **android:id="@+id/menuAdd"**
  - Sử dụng switch case hoặc if-else



# Bắt sự kiện cho menu

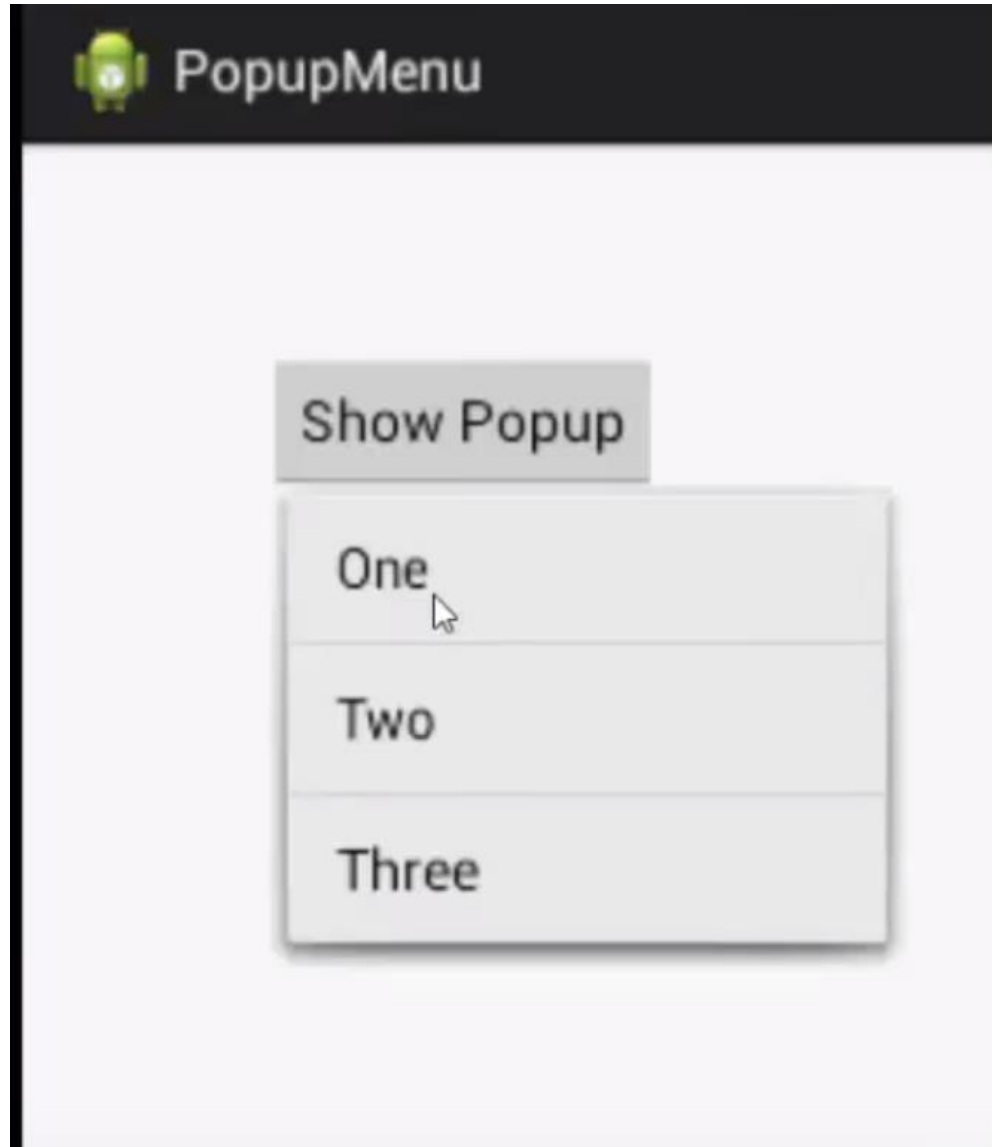
## ■ Cài đặt hàm onOptionsItemSelected

```
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case R.id.menuAdd:
            Toast.makeText(this, "Bạn kick vào add",
                Toast.LENGTH_SHORT).show();
            break;
        case R.id.menuSetting:
            Toast.makeText(this, "Bạn kick vào
                menuSetting", Toast.LENGTH_SHORT).show();
            break;
        case ...
    }
    return super.onOptionsItemSelected(item);
}
```



# Popup Menu

---





# Popup Menu

---

- Tạo một layout để chứa menu
  - Tạo thư mục (new directory) menu
  - Right click menu tạo Menu resource file
  - Trong menu resource file chứa tầng cặp thẻ <item>
- Các thuộc tính cơ bản của menu
  - Title: hiện thị nhãn menu(bắt buộc)
  - Icon="**@android:drawable/ic\_menu\_add**":đưa menu ra ngoài làm biểu tượng
  - showAsAction="**always**": luôn hiện thị biểu tượng



# Menu popup recoure file

---

```
<menu xmlns:android="">
    <item
        android:title="Thêm"
        android:id="@+id/menuThem"></item>
    <item
        android:id="@+id/menuXoa"
        android:title="Xóa"></item>
    <item
        android:id="@+id/menuSua"
        android:title="Sửa"></item>
</menu>
```



# Hiện thị popup menu

---

```
private void showMenu() {  
    PopupMenu popupMenu=new PopupMenu(this, btnPopup);  
  
    popupMenu.getMenuInflater().inflate(R.menu.menu_popup,popupMenu.getMenu());  
  
    popupMenu.show();  
}
```



# Sự kiện của popup menu

---

```
popupMenu.setOnMenuItemClickListener(new  
PopupMenu.OnMenuItemClickListener() {  
    @Override  
    public boolean onMenuItemClick(MenuItem item) {  
        switch (item.getItemId()) {  
            case R.id.menuThem:  
                btnPopup.setText("menuThem");  
                break;  
            ...  
        }  
        return false;  
    }  
});
```





# Menu popup recoure file

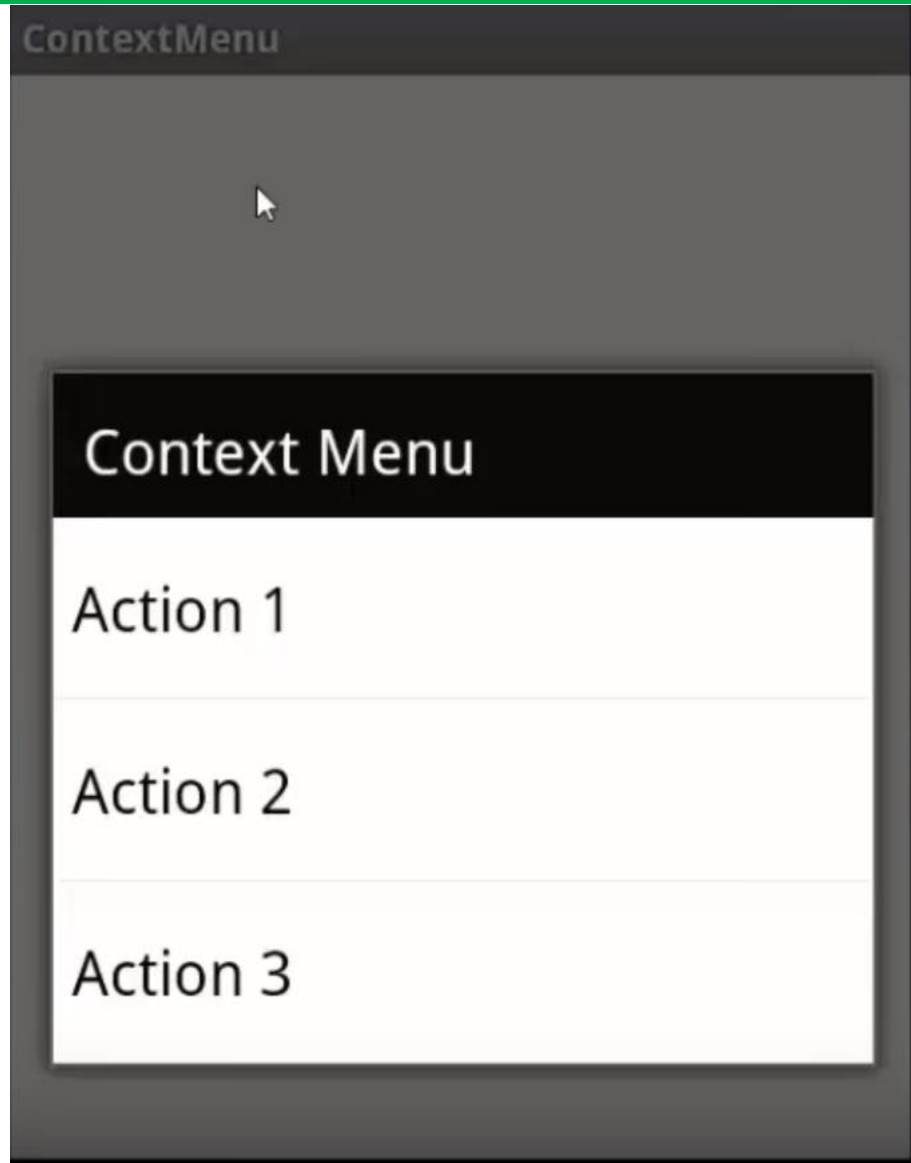
---

```
private void showMenu() {  
    PopupMenu popupMenu=new PopupMenu(this, btnPopup);  
    popupMenu.getMenuInflater().inflate(R.menu.menu_popup,popupMenu.getMenu());  
    popupMenu.setOnMenuItemClickListener(new PopupMenu.OnMenuItemClickListener() {  
        @Override  
        public boolean onMenuItemClick(MenuItem item) {  
            switch (item.getItemId()){  
                case R.id.menuThem:  
                    btnPopup.setText("menuThem");  
                    break;  
                case R.id.menuXoa:  
                    btnPopup.setText("menuXoa");  
                    break;  
                case R.id.menuSua:  
                    btnPopup.setText("menuSua");  
                    break;  
            }  
            return false;  
        }  
    });  
    popupMenu.show();  
}
```



# Context Menu

---





# Context Menu

---

- Tạo một layout để chứa context menu
  - Tạo thư mục (new directory) menu
  - Right click menu tạo Menu resource file
  - Trong menu resource file chứa tầng cặp thẻ <item>
- Các thuộc tính cơ bản của context menu
  - Title: hiện thị nhãn menu(bắt buộc)
  - Icon="**@android:drawable/ic\_menu\_add**":đưa menu ra ngoài làm biểu tượng
  - showAsAction="**always**": luôn hiện thị biểu tượng



# Menu context recourse file

---

```
<menu xmlns:android="">
    <item android:title="Màu xanh"
        android:id="@+id/menuXanh"></item>
    <item android:title="Màu Vàng"
        android:id="@+id/menuVang"></item>
    <item android:title="Màu Đ?"
        android:id="@+id/menuDo"></item>
</menu>
```



# Hiện thị context menu

---

```
private void showMenu() {  
    PopupMenu popupMenu=new PopupMenu(this, btnPopup);  
  
    popupMenu.getMenuInflater().inflate(R.menu.menu_popup,popupMenu.getMenu());  
  
    popupMenu.show();  
}
```



# Hiện thị context menu

---

## ■ Cài đặt hàm onCreateContextMenu

```
@Override
public void onCreateContextMenu(ContextMenu menu, View
v, ContextMenu.ContextMenuInfo menuInfo) {
    super.onCreateContextMenu(menu, v, menuInfo);
    getMenuInflater().inflate(R.menu.menu_context, menu);
    menu.setHeaderTitle("Chọn màu");
    menu.setHeaderIcon(R.mipmap.ic_launcher);
}
```

## ■ Đăng ký view cho context menu

- `registerForContextMenu(btnChonMau);`



# Sự kiện của context menu

---

## ■ Cài đặt hàm onOptionsItemSelected

```
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case R.id.menuXanh:
            clManHinh.setBackgroundColor(Color.BLUE);
            break;

        ...
    }
    return super.onOptionsItemSelected(item);
}
```



Phần 2

# Làm việc với Alert Dialog





# Alert Dialog cơ bản

---

## ■ Khởi tạo và hiện thị

- `AlertDialog.Builder builder = new AlertDialog.Builder(MainActivity.this);`
- `builder.show();`

## ■ Các thuộc tính:

- Title: `builder.setTitle("Cảnh báo xóa...!");`
- Icon: `builder.setIcon(id ảnh);`
- Message: `builder.setMessage("Thông báo...");`

## ■ Nút bấm:

- Khẳng định: `builder.setPositiveButton`
- Phủ định: `builder.setNegativeButton`



# Alert Dialog nâng cao

CustomDialog

**Đăng nhập hệ thống**

Nhập tài khoản

Nhập mật khẩu

ĐĂNG NHẬP

HỦY



# Alert Dialog nâng cao

- Thiết kế dialog(create layout.xml)

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:background="#2de9d3"
    android:layout_height="match_parent">
    <TextView...>
    <EditText...>
    <EditText...>
    <LinearLayout
        android:weightSum="10"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <Button...>
        <Button...>
    </LinearLayout>
</LinearLayout>
```



# Alert Dialog nâng cao

---

## ■ Khởi tạo và hiện thị

- `Dialog dialog=new Dialog(this);`
- `dialog setContentView(R.layout.logindialog);`
- `dialog.show();`

## ■ Ảnh xạ:

- `btnLogin= dialog.findViewById(R.id.btnLogin);`
- ...

## ■ Các thuộc tính:

- `Tittle: builder.setTitle("Cảnh báo xóa...!");`
- `dialog.requestWindowFeature(Window.FEATURE_NO_TITLE);`
- `dialog.setCanceledOnTouchOutside(false);`



Phần 3

# Date Picker Dialog



# Calendar

## ■ Khởi tạo

- `Calendar calendar=Calendar.getInstance();`

## ■ Hàm cơ bản:

- `calendar.getTime():` trả về ngày giờ chi tiết;
- `calendar.get(Calendar.DATE):` return ngày;
- `calendar.get(Calendar.MONTH):` return tháng;
- `calendar.get(Calendar.YEAR):` return năm;
- `calendar.get(Calendar.HOUR):` return giờ;
- `calendar.get(Calendar.HOUR_OF_DAY):` return giờ trong ngày (24h);

## ■ Định dạng ngày và giờ:

- `new SimpleDateFormat("dd/MM/yyyy");`
- `simpleDateFormat.format(calendar.getTime());`
- `new SimpleDateFormat("hh:mm:ss a");`



# Date Picker Dialog

---

- `final Calendar calendar=Calendar.getInstance();`
- `int year, month, day;`
- `//Lấy ngày tháng năm hiện tại của hệ thống`
- `year=calendar.get(Calendar.YEAR);`
- `month=calendar.get(Calendar.MONTH);`
- `day=calendar.get(Calendar.DATE);`



# Date Picker Dialog

---

```
DatePickerDialog          dialog          =          new
DatePickerDialog(MainActivity.this,new
DatePickerDialog.OnDateSetListener, year, month, day);
dialog.show();
```

```
new DatePickerDialog.OnDateSetListener() {
    @Override
    public void onDateSet(DatePicker view, int year,
int month, int dayOfMonth) {
        SimpleDateFormat          simpleDateFormat=new
SimpleDateFormat("dd/MM/yyyy");
        calendar.set(year, month, dayOfMonth);
        edtNgay.setText(simpleDateFormat.format(calendar.g
etTime()));
    }
}
```





# Time Picker Dialog

---

- `final Calendar calendar=Calendar.getInstance();`
- `//Lấy giờ hiện tại của hệ thống`
- `int gio=calendar.get(Calendar.HOUR_OF_DAY); //24h`
- `int phut=calendar.get(Calendar.MINUTE);`



# Time Picker Dialog

---

```
TimePickerDialog                                dialog=new
TimePickerDialog(MainActivity.this,              new
TimePickerDialog, gio, phut, true);
dialog.show();

new TimePickerDialog.OnTimeSetListener() {
    public void onTimeSet(TimePicker view, int
hourOfDay, int minute) {
        calendar.set(0,0,0,hourOfDay, minute);
        SimpleDateFormat simpleDateFormat=new
SimpleDateFormat("HH:mm:ss");
        edtGio.setText(simpleDateFormat.format(calendar.ge
tTime()));
    }
}
```