

# Image Request

Lecturer: Chuor Porchourng

Presented by: Han Leangsiv

Meng Taingon

Group: CS2A

# CONTENT

- What is Picasso?
- Concept of using image request
- How to implement image request?
- Result

# What is Picasso?

Picasso

A powerful image downloading and caching library for Android

- Picasso is mostly used in image **loading** library for Android.
- It simplifies the process of displaying images from external locations

# Concept of using image request

- In order to do an Image Request, we need a **library** such as **Picasso, Glide v4, etc.**
- We use **Picasso** in this case to do the image request.
- It downloads the image using url from internet to display it on ImageView

# How to implement image request?

**STEP1:** Implement the Picasso Library into to build.gradle(Module.app) and Sync it

```
dependencies {  
    implementation fileTree(dir: 'libs', include: ['*.jar'])  
    implementation 'androidx.appcompat:appcompat:1.0.2'  
    implementation 'androidx.constraintlayout:constraintlayout:1.1.3'  
    testImplementation 'junit:junit:4.12'  
    androidTestImplementation 'androidx.test:runner:1.2.0'  
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.2.0'  
    ADD //Add a Picasso Library for requesting image  
    implementation 'com.squareup.picasso:picasso:2.71828'  
}
```

## STEP2: Give our app the PERMISSION of INTERNET into AndroidManifest.xml

GIVE



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.imagerequest">
    <uses-permission android:name="android.permission.INTERNET"></uses-permission>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="Image Request"
        android:roundIcon="@mipmap/ic_launcher_round"
        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>
    </application>
</manifest>
```

## STEP3: Create the ImageView (Give ID) in Main\_activity.xml

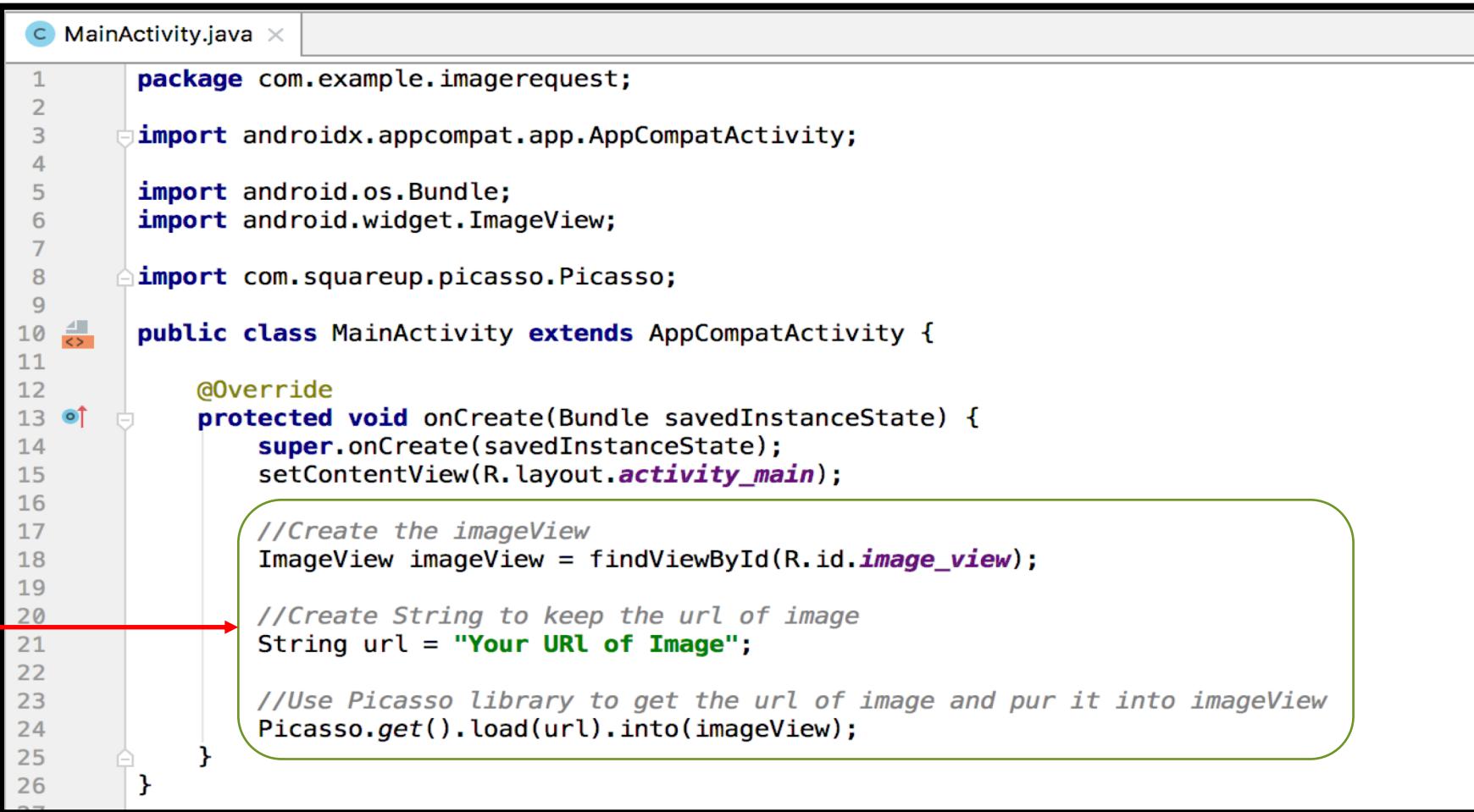


Create

```
activity_main.xml x
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      tools:context=".MainActivity"
8      android:background="@color/lightGray">
9
10 <ImageView
11     android:id="@+id/image_view"
12     android:layout_width="wrap_content"
13     android:layout_height="wrap_content"
14     app:layout_constraintBottom_toBottomOf="parent"
15     app:layout_constraintLeft_toLeftOf="parent"
16     app:layout_constraintRight_toRightOf="parent"
17     app:layout_constraintTop_toTopOf="parent" />
18
19 </androidx.constraintlayout.widget.ConstraintLayout>
```

## STEP4: Implement code in MainActivity.java

Coding

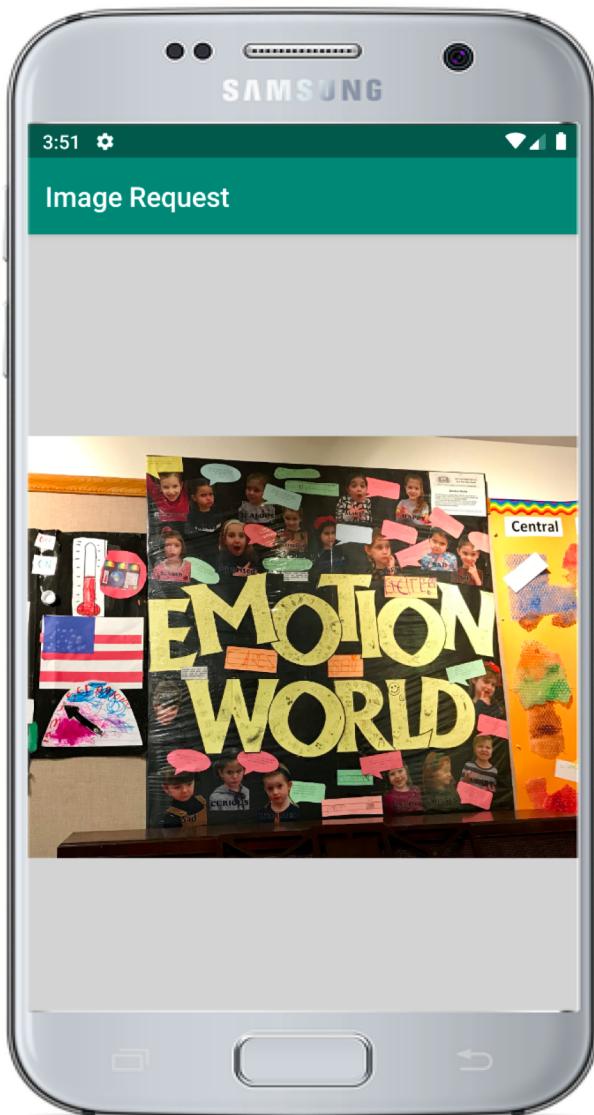


```
>MainActivity.java
```

```
1 package com.example.imagerequest;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.os.Bundle;
6 import android.widget.ImageView;
7
8 import com.squareup.picasso.Picasso;
9
10 public class MainActivity extends AppCompatActivity {
11
12     @Override
13     protected void onCreate(Bundle savedInstanceState) {
14         super.onCreate(savedInstanceState);
15         setContentView(R.layout.activity_main);
16
17         //Create the imageView
18         ImageView imageView = findViewById(R.id.image_view);
19
20         //Create String to keep the url of image
21         String url = "Your URL of Image";
22
23         //Use Picasso library to get the url of image and put it into imageView
24         Picasso.get().load(url).into(imageView);
25
26     }
27 }
```

The code implements the `onCreate()` method of the `MainActivity`. It first calls `super.onCreate(savedInstanceState)` and then sets the content view to `R.layout.activity_main`. Inside the `onCreate()` method, it creates an `ImageView` and finds its view by ID. It then creates a string variable `url` containing the URL of the image. Finally, it uses the `Picasso` library to load the image from the URL and set it into the `imageView`.

# Result







**QUESTION???**