Design Patterns

MSc in Computer Science

Produced by

Eamonn de Leastar (edeleastar@wit.ie)

Department of Computing, Maths & Physics
Waterford Institute of Technology

http://www.wit.ie

http://elearning.wit.ie

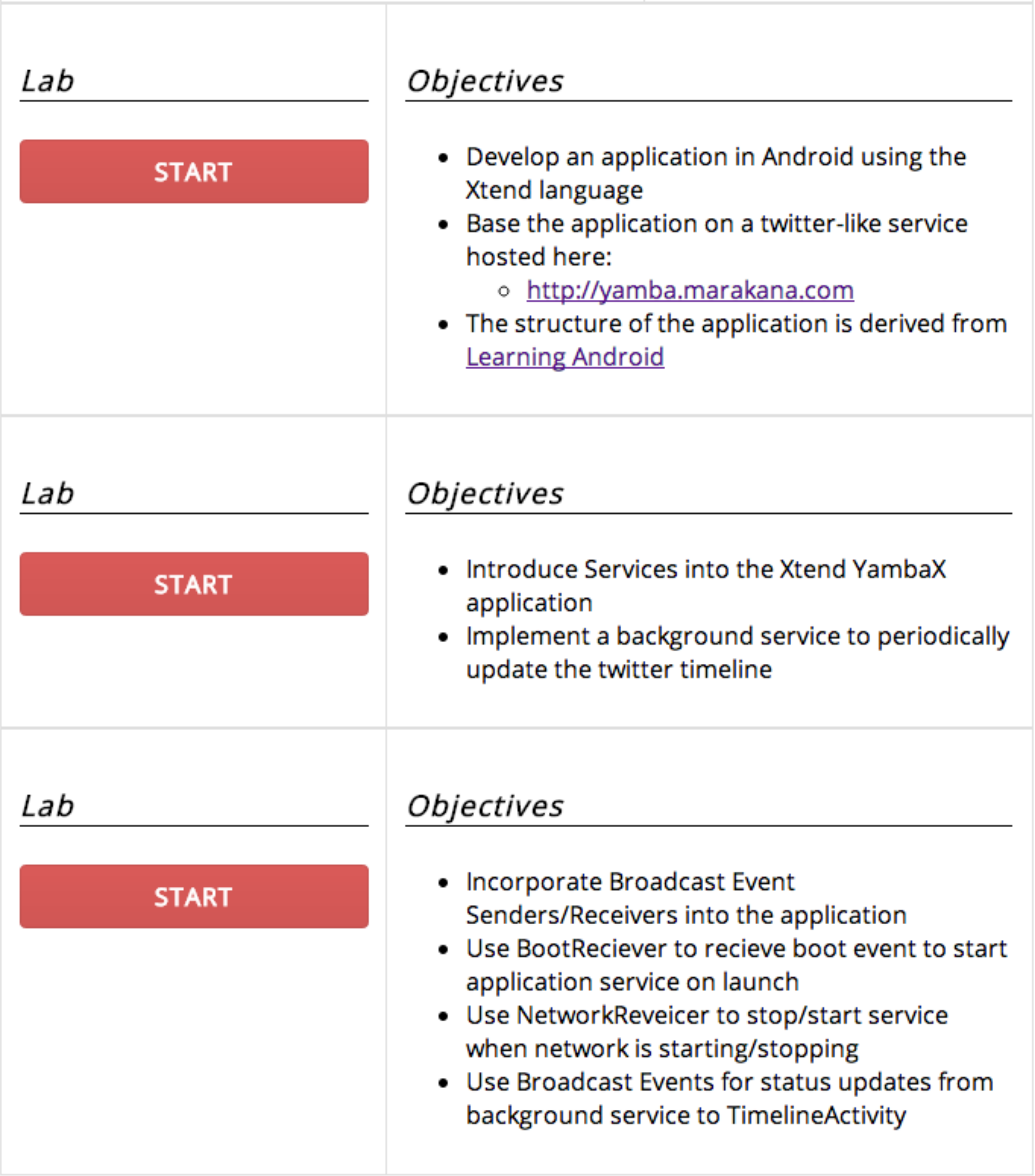
Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

eLearning
support unit

Yamba

Xtend version Encapsulated as 3 Labs

- A - Enable Simple Tweet + timeline update on background thread
- B - Move background thread to an Android Service + restructure application to use Lambdas + Command pattern
- C - replace custom event mechanism with generic Broadcast Receivers



Lab

Objectives

- Develop an application in Android using the Xtend language
- Base the application on a twitter-like service hosted here:
  - http://yamba.marakana.com
- The structure of the application is derived from Learning Android

Lab

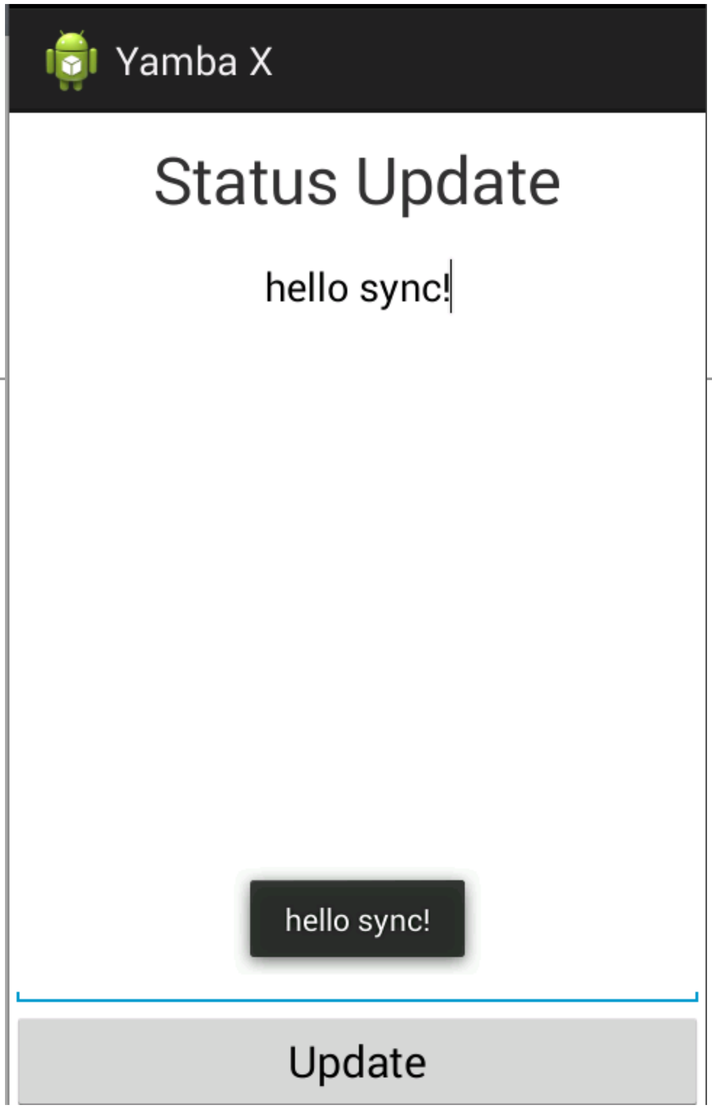
Objectives

- Introduce Services into the Xtend YambaX application
- Implement a background service to periodically update the twitter timeline

Lab

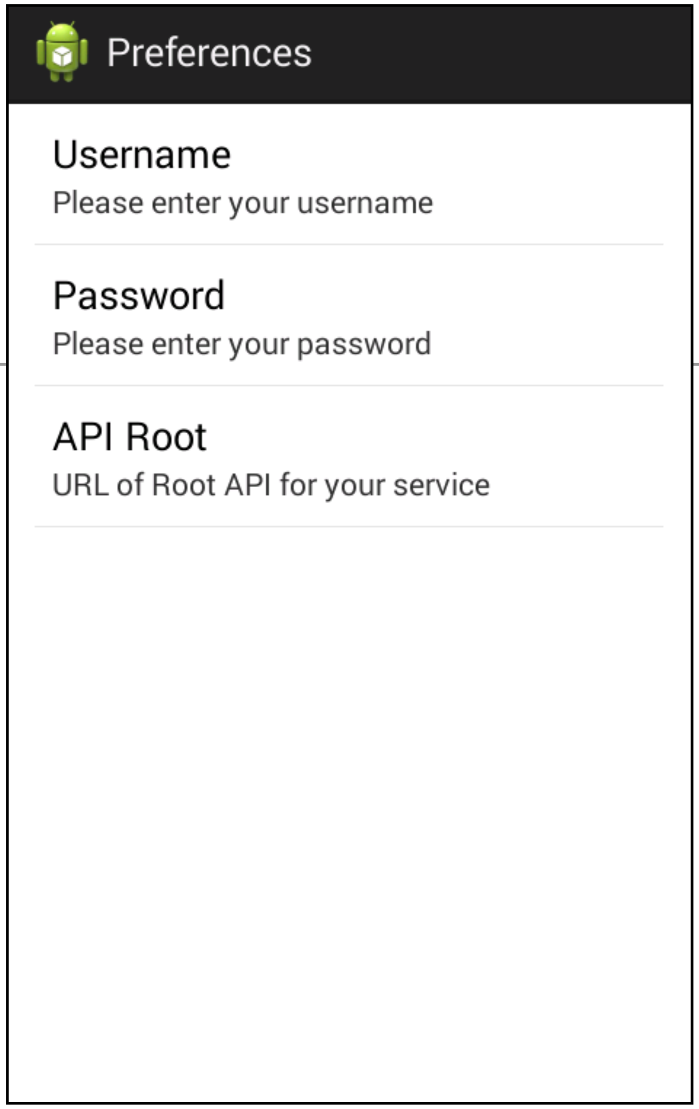
Objectives

- Incorporate Broadcast Event Senders/Receivers into the application
- Use BootReciever to recieve boot event to start application service on launch
- Use NetworkReveicer to stop/start service when network is starting/stopping
- Use Broadcast Events for status updates from background service to TimelineActivity



hello sync!

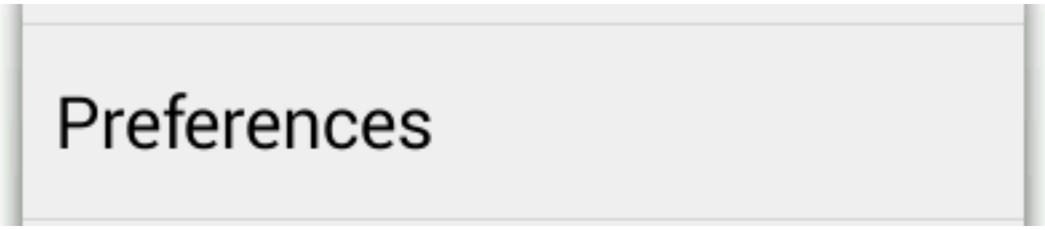
hello sync!



Username
Please enter your username

Password
Please enter your password

API Root
URL of Root API for your service



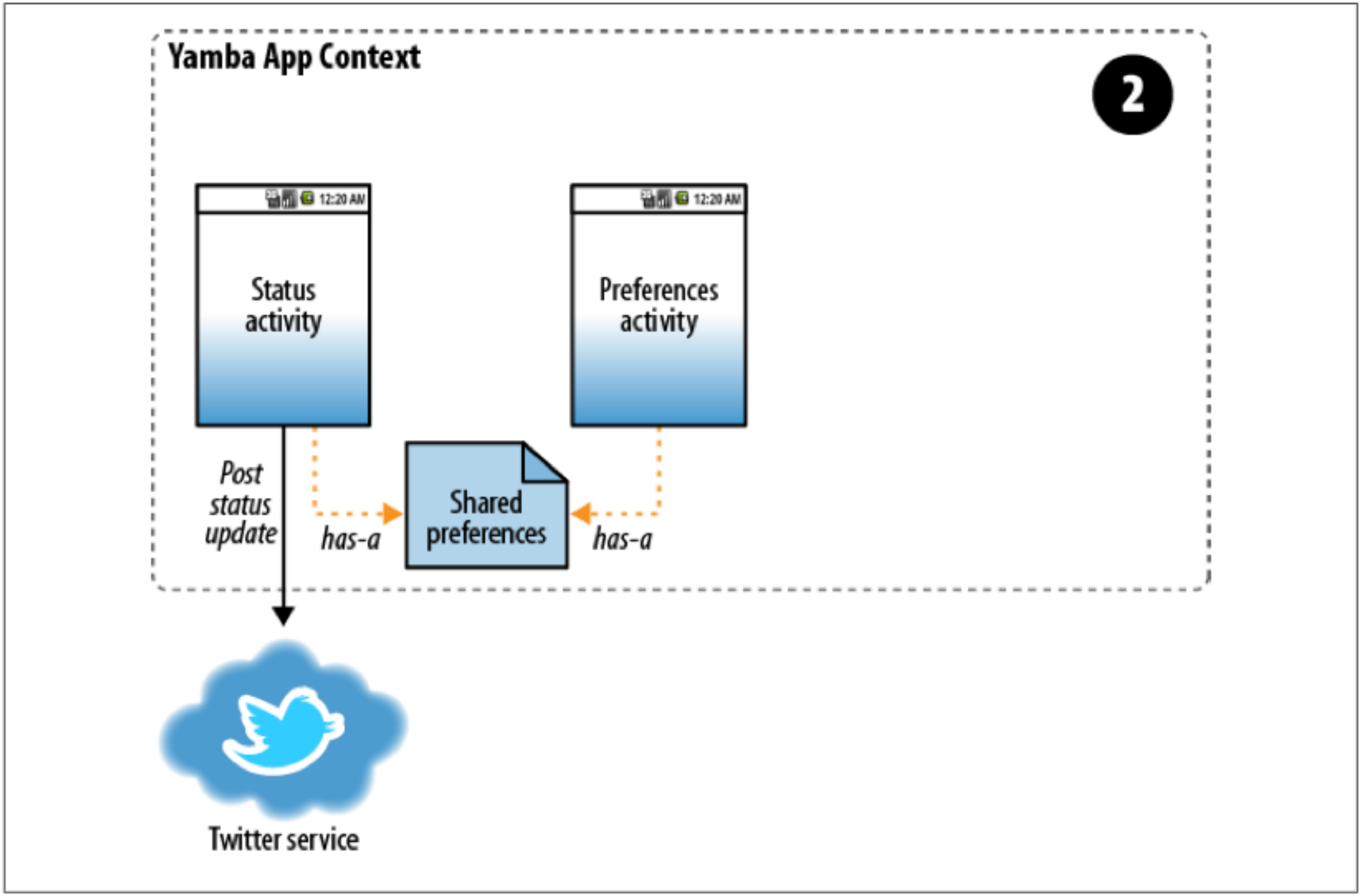
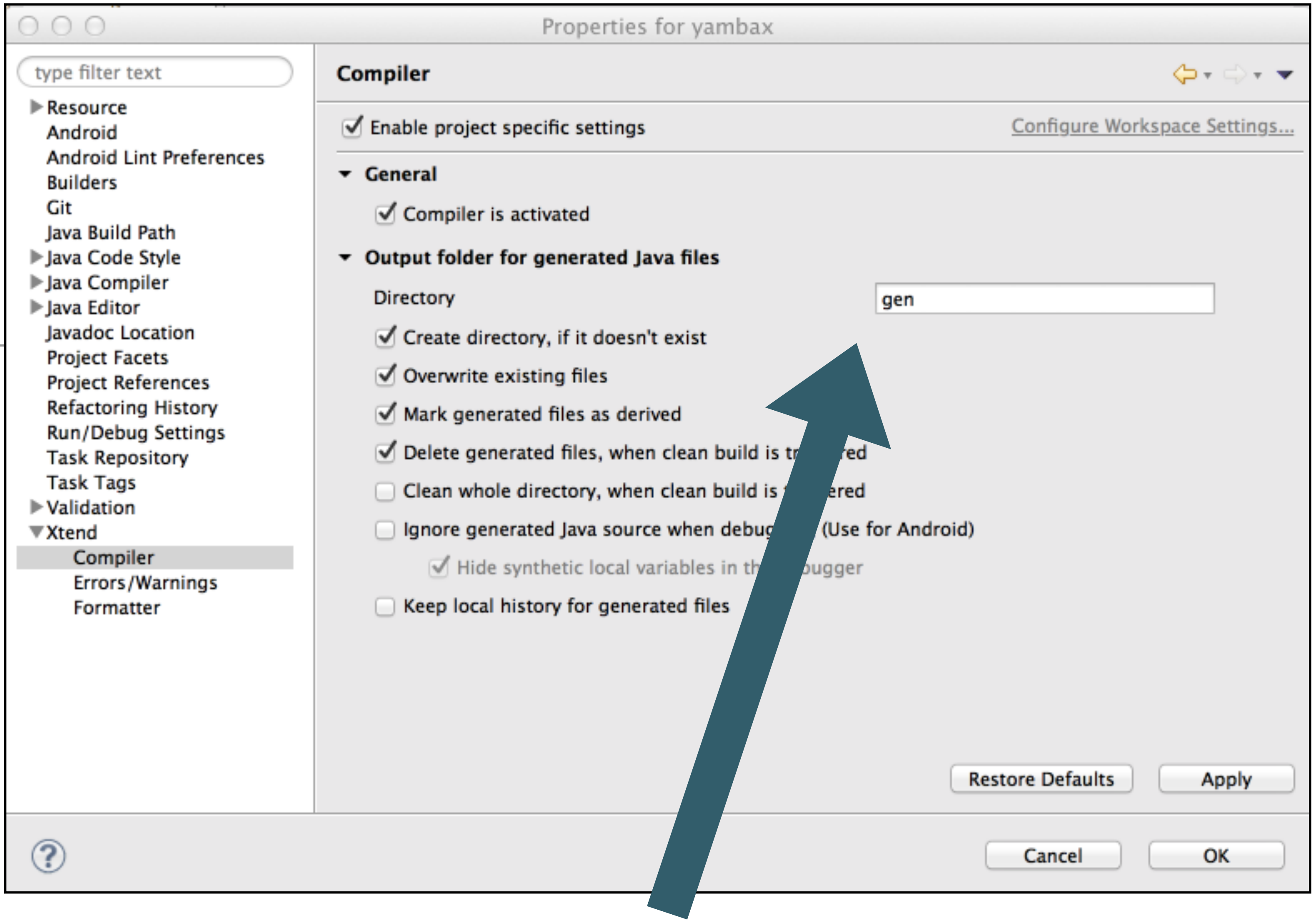


Figure 7-6. Yamba completion

Initial Xtend App



Properties for yambax

Compiler

☑ Enable project specific settings

General

☑ Compiler is activated

Output folder for generated Java files

Directory: gen

☑ Create directory, if it doesn't exist

☑ Overwrite existing files

☑ Mark generated files as derived

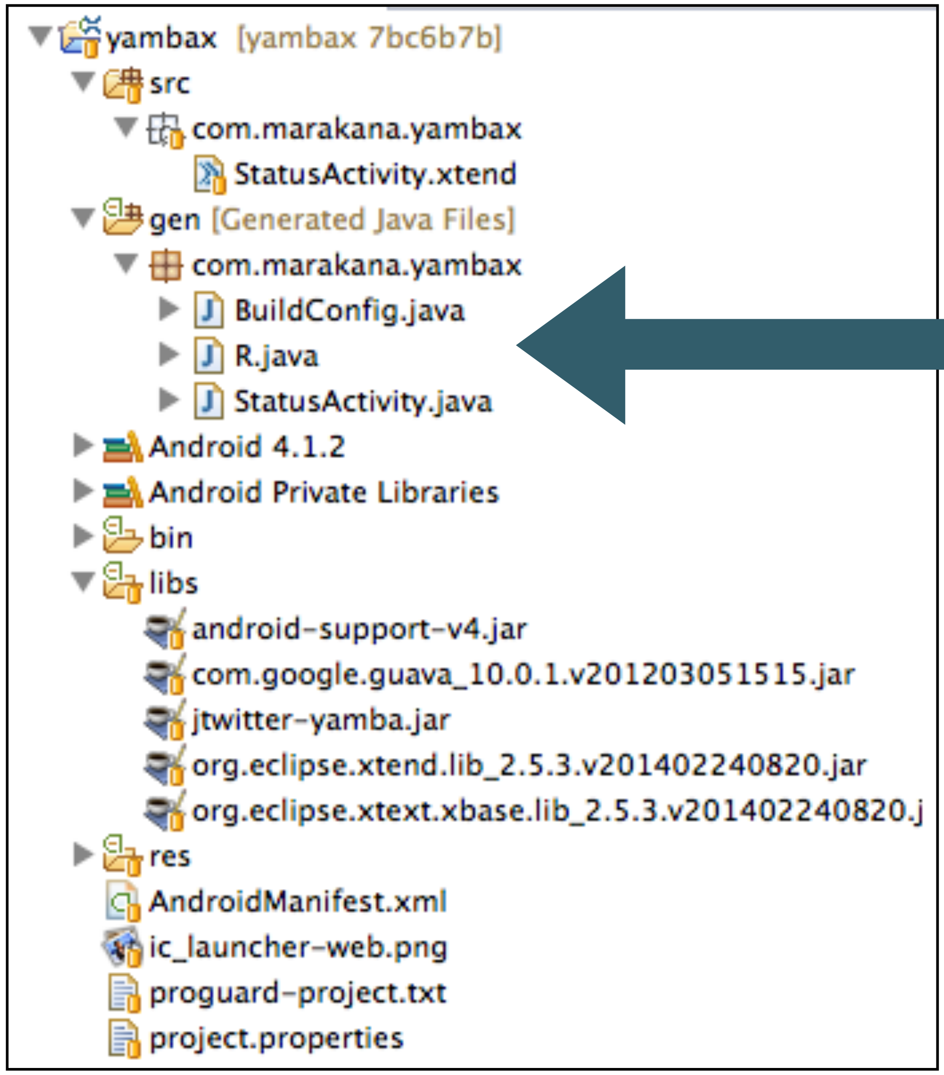
☑ Delete generated files, when clean build is tr red

☐ Clean whole directory, when clean build is ered

☐ Ignore generated Java source when debug (Use for Android)

☑ Hide synthetic local variables in th bugger

☐ Keep local history for generated files

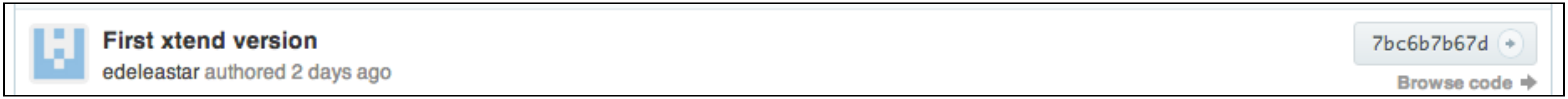


```
yambax [yambax 7bc6b7b]
  src
    com.marakana.yambax
      StatusActivity.xtend
  gen [Generated Java Files]
    com.marakana.yambax
      BuildConfig.java
      R.java
      StatusActivity.java
  Android 4.1.2
  Android Private Libraries
  bin
  libs
    android-support-v4.jar
    com.google.guava_10.0.1.v201203051515.jar
    jtwitter-yamba.jar
    org.eclipse.xtend.lib_2.5.3.v201402240820.jar
    org.eclipse.xtext.xbase.lib_2.5.3.v201402240820.j
  res
  AndroidManifest.xml
  ic_launcher-web.png
  proguard-project.txt
  project.properties
```

- Instruct Eclipse to generate java version of xtend classes into ‘gen’

- 3 Xtend jar files

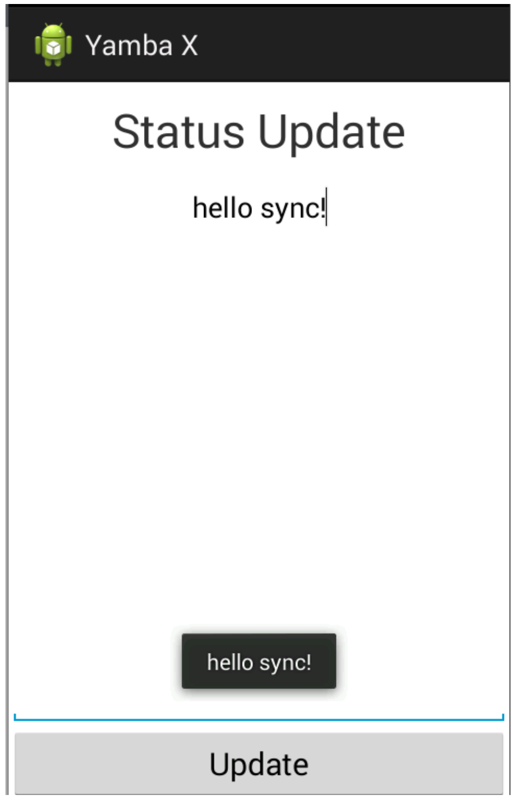
- 1 jtwitter api jar

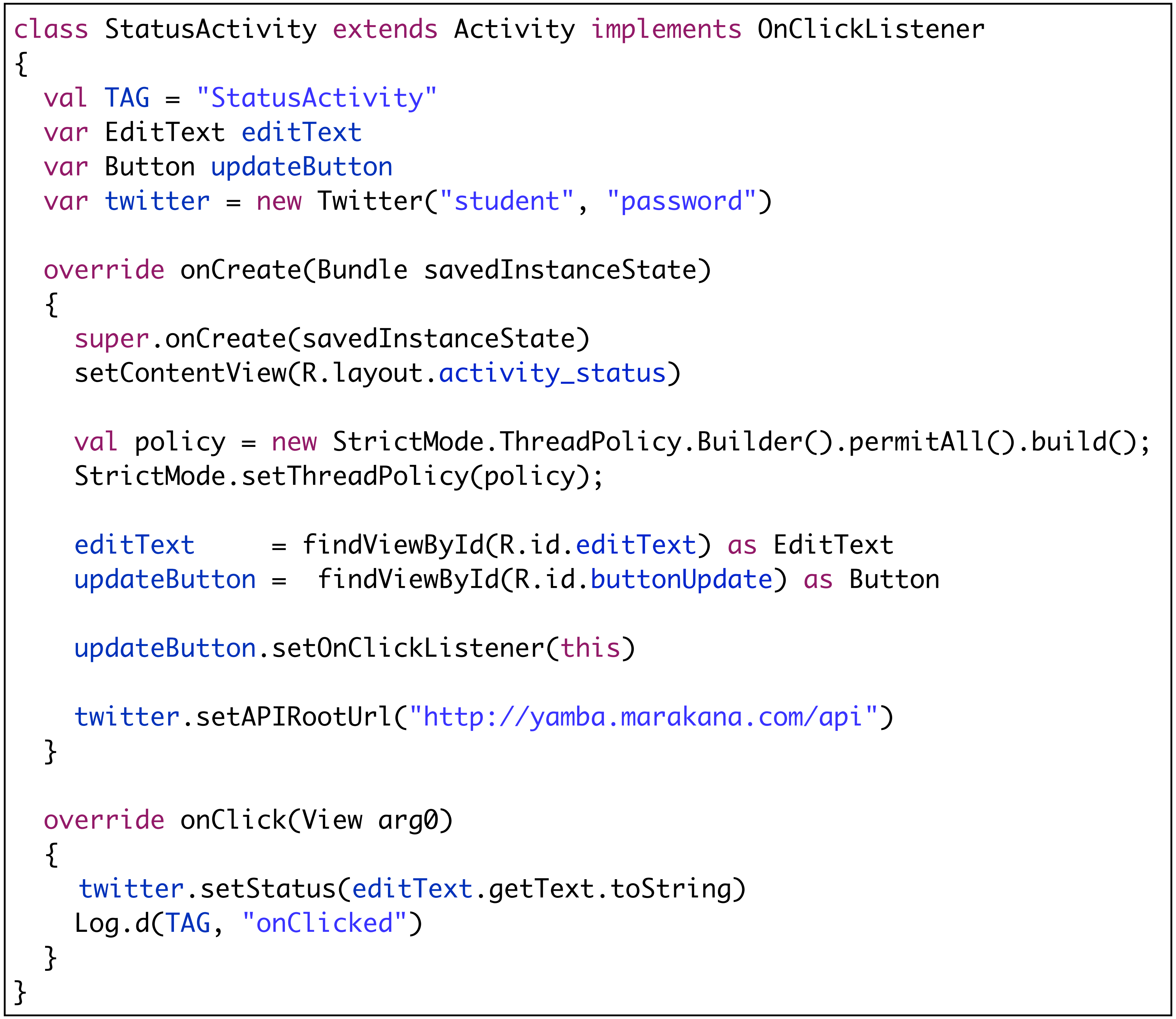


First xtend version

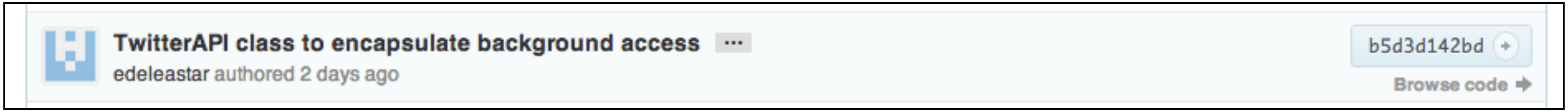
edeleastar authored 2 days ago

- Disable strictMode ensures app can access network on main Thread
- Post tweets to default user (student, password)





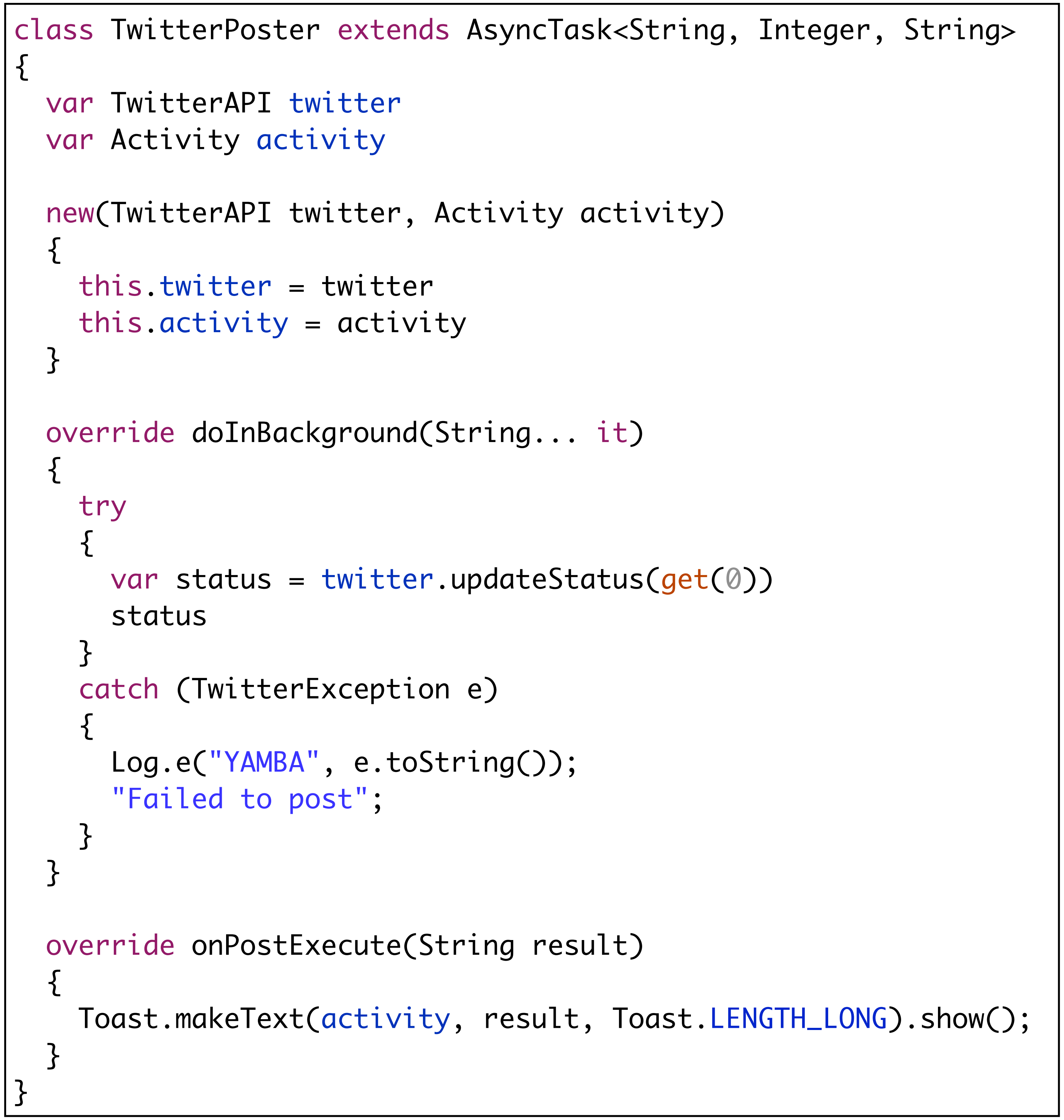
```
class StatusActivity extends Activity implements OnClickListener
{
  val TAG = "StatusActivity"
  var EditText editText
  var Button updateButton
  var twitter = new Twitter("student", "password")

  override onCreate(Bundle savedInstanceState)
  {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_status)

    val policy = new StrictMode.ThreadPolicy.Builder().permitAll().build();
    StrictMode.setThreadPolicy(policy);

    editText     = findViewById(R.id.editText) as EditText
    updateButton =  findViewById(R.id.buttonUpdate) as Button

    updateButton.setOnClickListener(this)

    twitter.setAPIRootUrl("http://yamba.marakana.com/api")
  }

  override onClick(View arg0)
  {
     twitter.setStatus(editText.getText.toString)
    Log.d(TAG, "onClicked")
  }
}
```



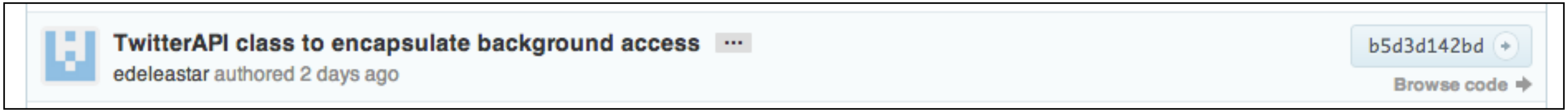
TwitterAPI class to encapsulate background access

edeleastar authored 2 days ago

- Use AsyncTask + jtwiter library



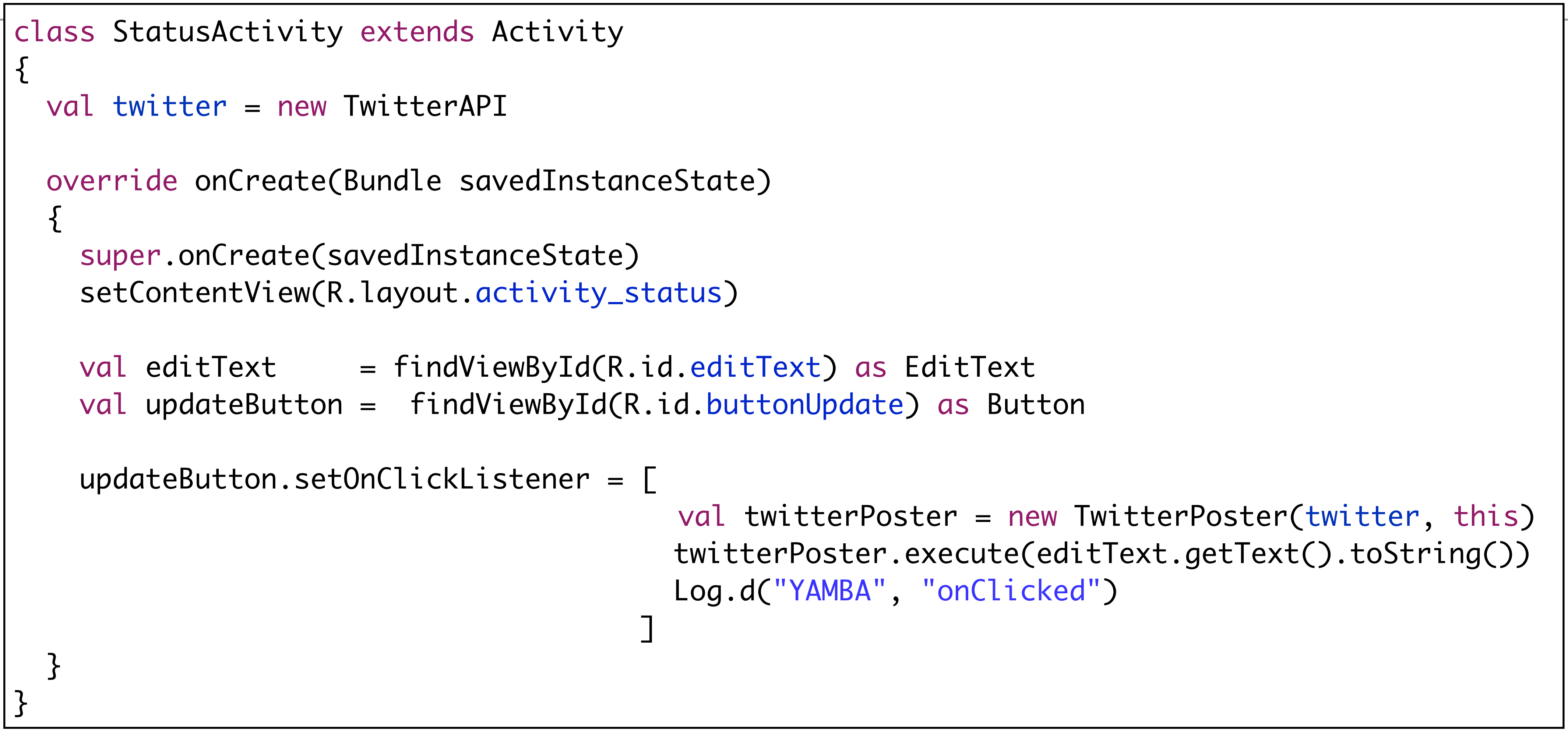
```
class TwitterPoster extends AsyncTask<String, Integer, String>
{
  var TwitterAPI twitter
  var Activity activity

  new(TwitterAPI twitter, Activity activity)
  {
    this.twitter = twitter
    this.activity = activity
  }

  override doInBackground(String... it)
  {
    try
    {
      var status = twitter.updateStatus(get(0))
      status
    }
    catch (TwitterException e)
    {
      Log.e("YAMBA", e.toString());
      "Failed to post";
    }
  }

  override onPostExecute(String result)
  {
    Toast.makeText(activity, result, Toast.LENGTH_LONG).show();
  }
}
```



TwitterAPI class to encapsulate background access

edeleastar authored 2 days ago

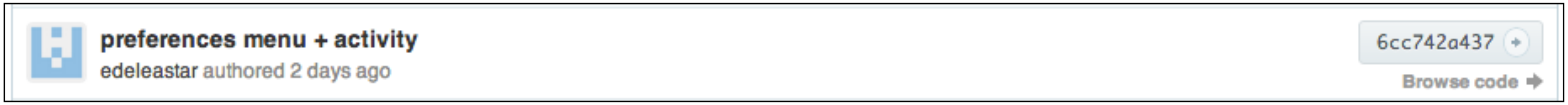
b5d3d142bd



```
class StatusActivity extends Activity
{
  val twitter = new TwitterAPI

  override onCreate(Bundle savedInstanceState)
  {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_status)

    val editText     = findViewById(R.id.editText) as EditText
    val updateButton =  findViewById(R.id.buttonUpdate) as Button

    updateButton.setOnClickListener = [
                                        val twitterPoster = new TwitterPoster(twitter, this)
                                        twitterPoster.execute(editText.getText().toString())
                                        Log.d("YAMBA", "onClicked")
                                      ]
  }
}
```

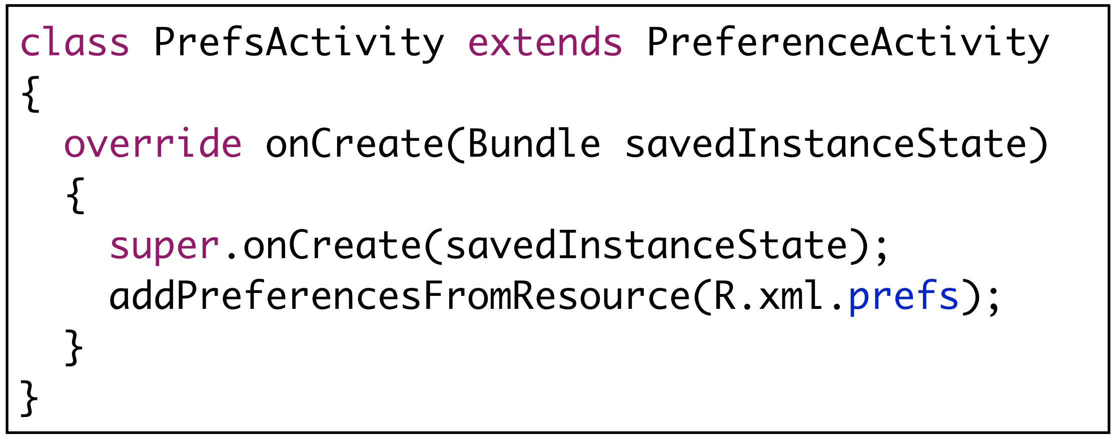
- Use lambda to initialise event handler

- Uses ‘Single Abstract Method’ convenience feature of tend

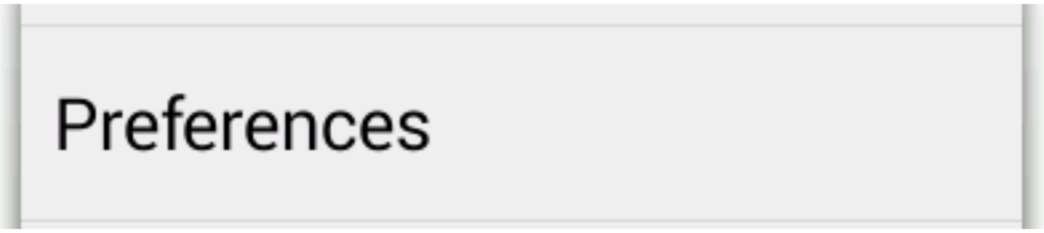


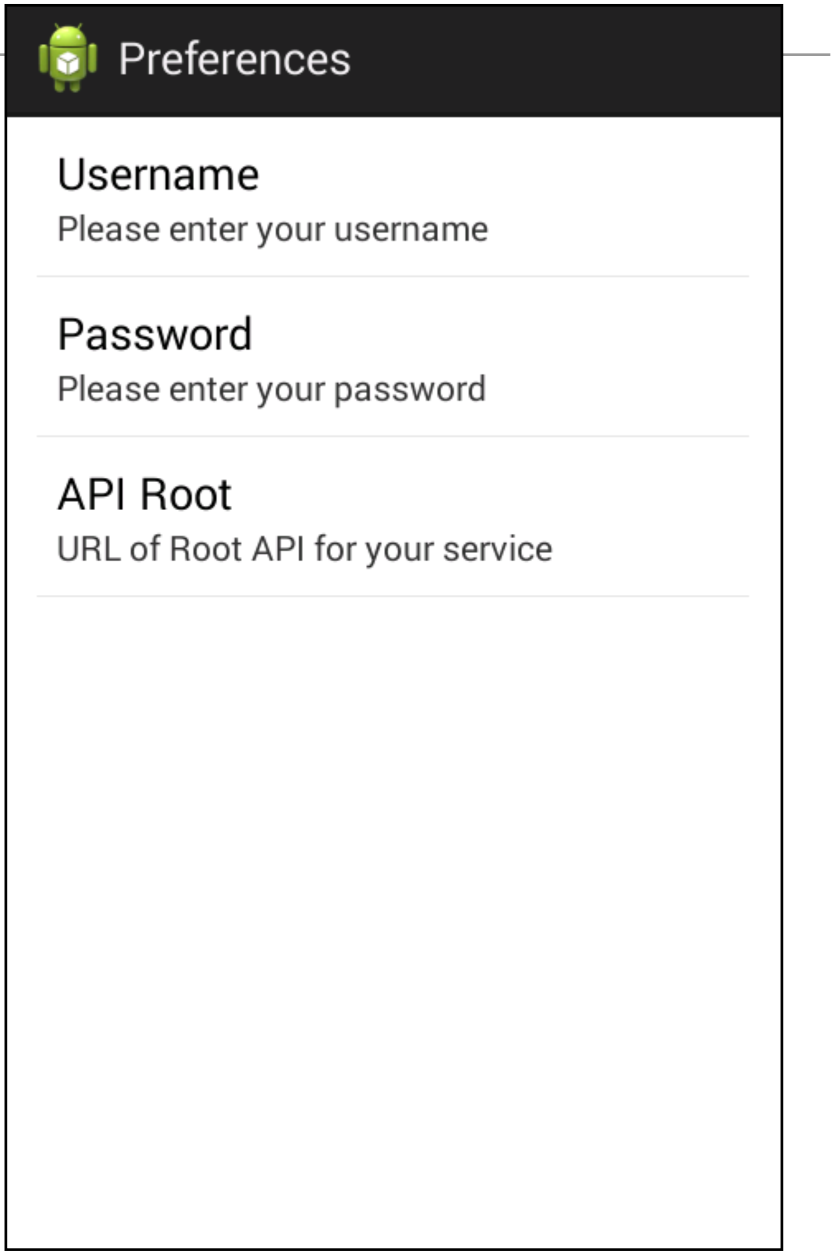
preferences menu + activity

edeleastar authored 2 days ago



```
class PrefsActivity extends PreferenceActivity
{
  override onCreate(Bundle savedInstanceState)
  {
    super.onCreate(savedInstanceState);
    addPreferencesFromResource(R.xml.prefs);
  }
}
```





Username
Please enter your username

Password
Please enter your password

API Root
URL of Root API for your service

- Preferences resources + activity

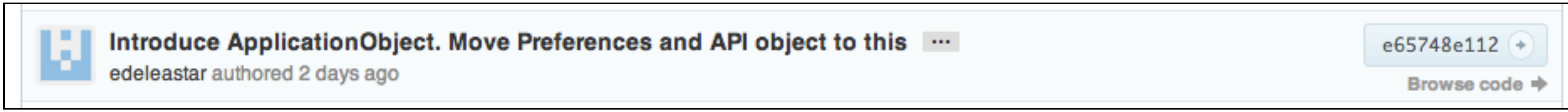
preferences menu + activity
edeleastar authored 2 days ago

6cc742a437

Browse code

```
class StatusActivity extends Activity
{
  var EditText       editText
  var Button         updateButton

  var update       = [ new TwitterPoster(this).execute(editText.getText.toString)  ] as OnClickListener

  override onCreate(Bundle savedInstanceState)
  {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_status)
    editText     = findViewById(R.id.editText) as EditText
    updateButton = findViewById(R.id.buttonUpdate) as Button
    updateButton.setOnClickListener = update
  }

  override  onCreateOptionsMenu(Menu menu)
  {
    getMenuInflater.inflate(R.menu.menu, menu)
    true
  }

   override onOptionsItemSelected(MenuItem item)
  {
    startActivity(new Intent(this, typeof(PrefsActivity)))
    true
  }
}
```

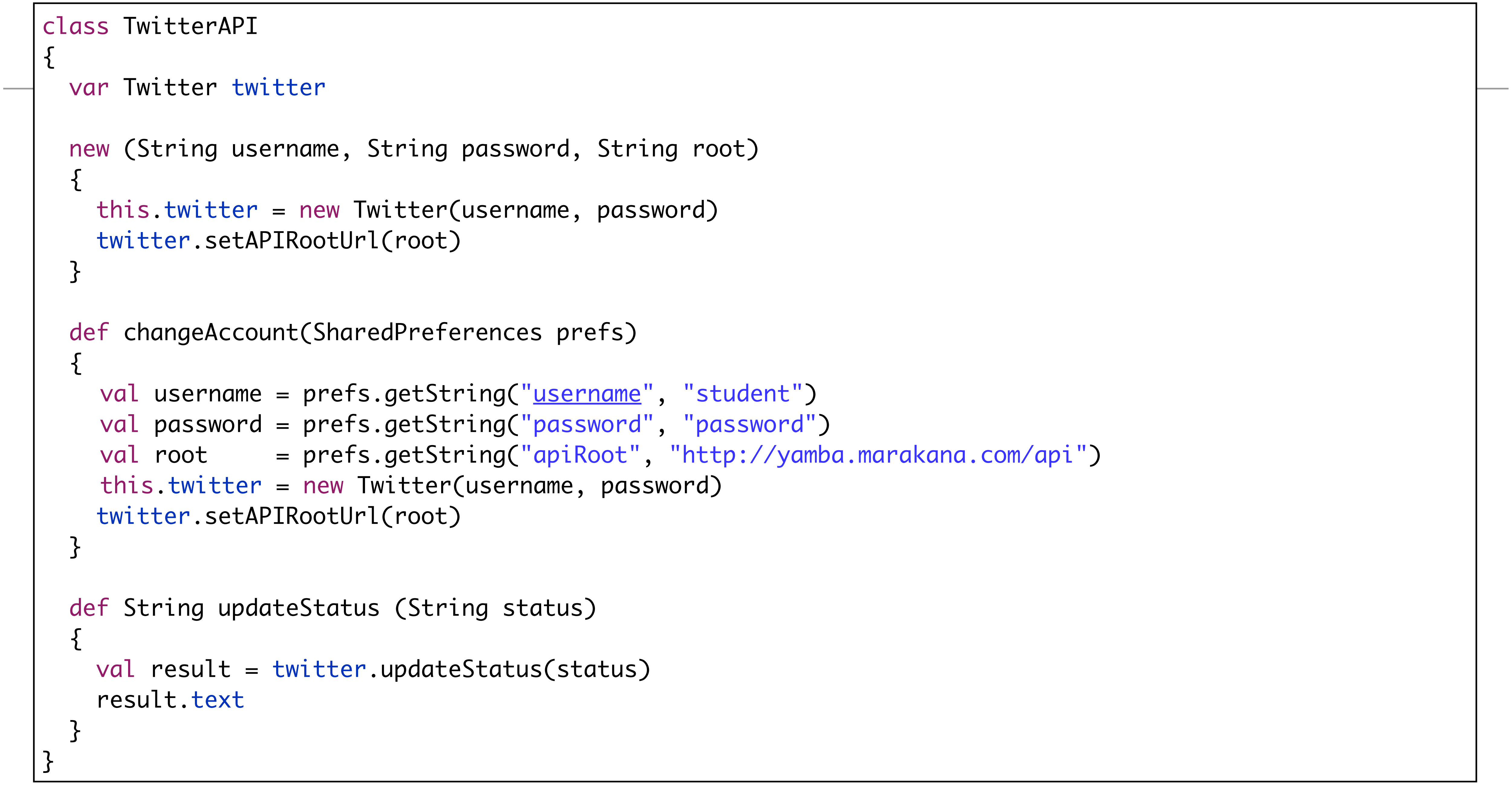
- Lambda now a class member, declared inline



Introduce ApplicationObject. Move Preferences and API object to this ···

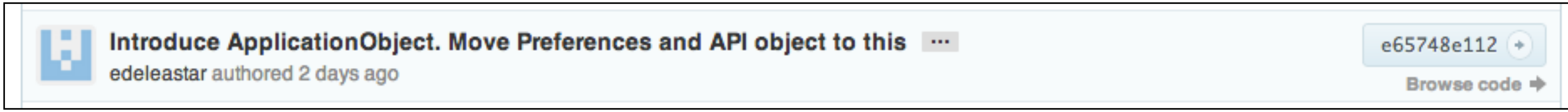
edeleastar authored 2 days ago

e65748e112



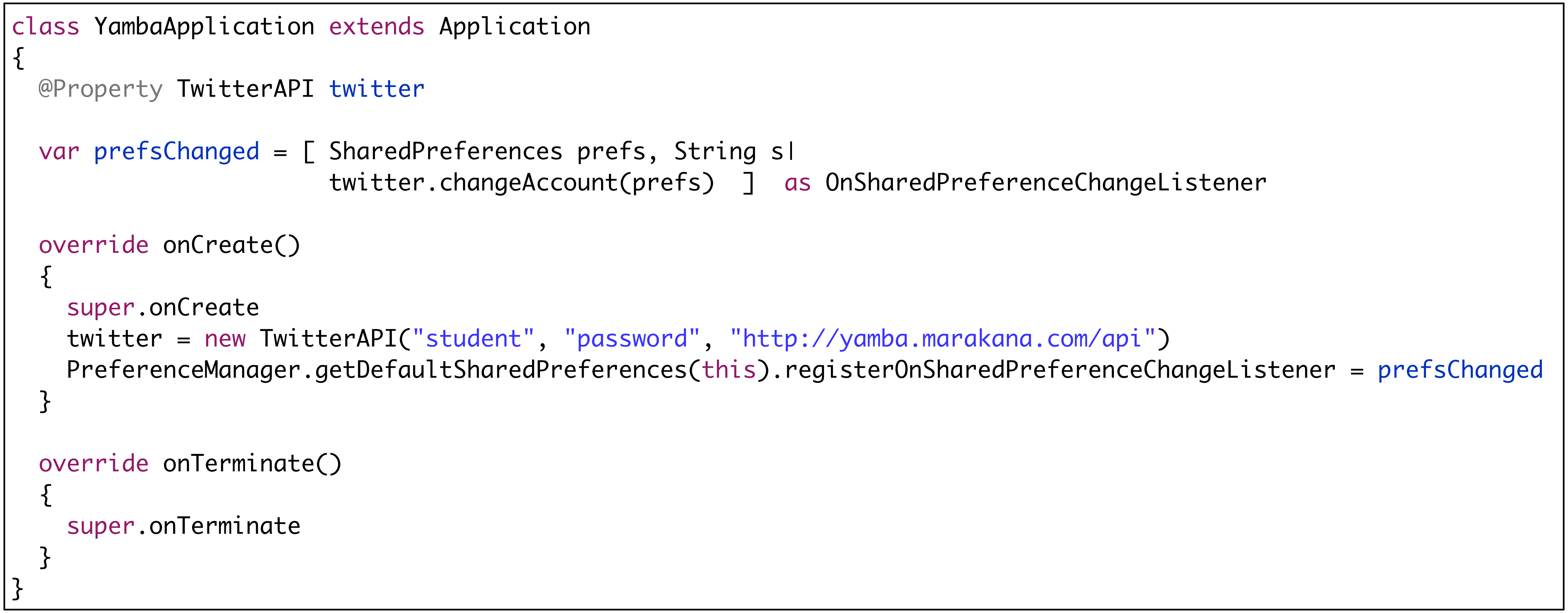
```
class TwitterAPI
{
  var Twitter twitter

  new (String username, String password, String root)
  {
    this.twitter = new Twitter(username, password)
    twitter.setAPIRootUrl(root)
  }

  def changeAccount(SharedPreferences prefs)
  {
    val username = prefs.getString("username", "student")
    val password = prefs.getString("password", "password")
    val root     = prefs.getString("apiRoot", "http://yamba.marakana.com/api")
    this.twitter = new Twitter(username, password)
    twitter.setAPIRootUrl(root)
  }

  def String updateStatus (String status)
  {
    val result = twitter.updateStatus(status)
    result.text
  }
}
```

- All twitter access through this ‘proxy’ object



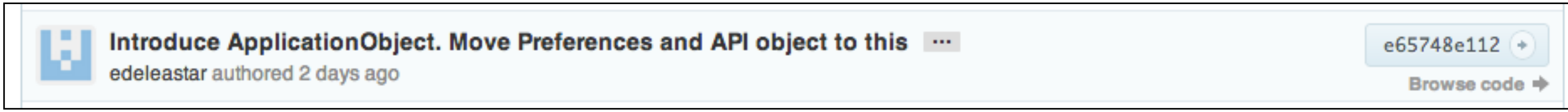
Introduce ApplicationObject. Move Preferences and API object to this

edeleastar authored 2 days ago



```
class YambaApplication extends Application
{
  @Property TwitterAPI twitter

  var prefsChanged = [ SharedPreferences prefs, String s|
                       twitter.changeAccount(prefs)  ]  as OnSharedPreferenceChangeListener

  override onCreate()
  {
    super.onCreate
    twitter = new TwitterAPI("student", "password", "http://yamba.marakana.com/api")
    PreferenceManager.getDefaultSharedPreferences(this).registerOnSharedPreferenceChangeListener = prefsChanged
  }

  override onTerminate()
  {
    super.onTerminate
  }
}
```

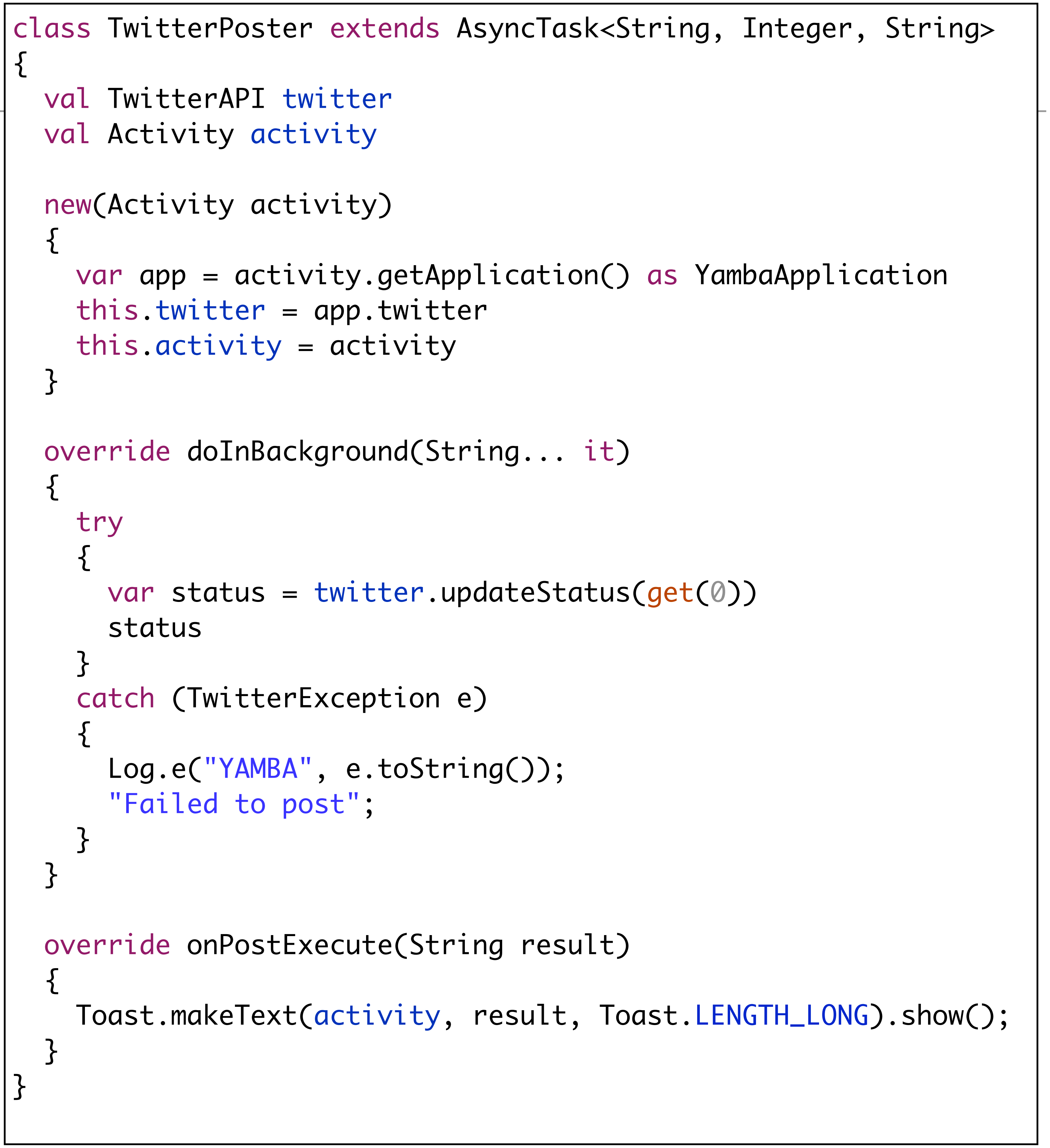
- Application ‘singleton’
- twitterapi object a ‘property’ that can be accessed and used elsewhere



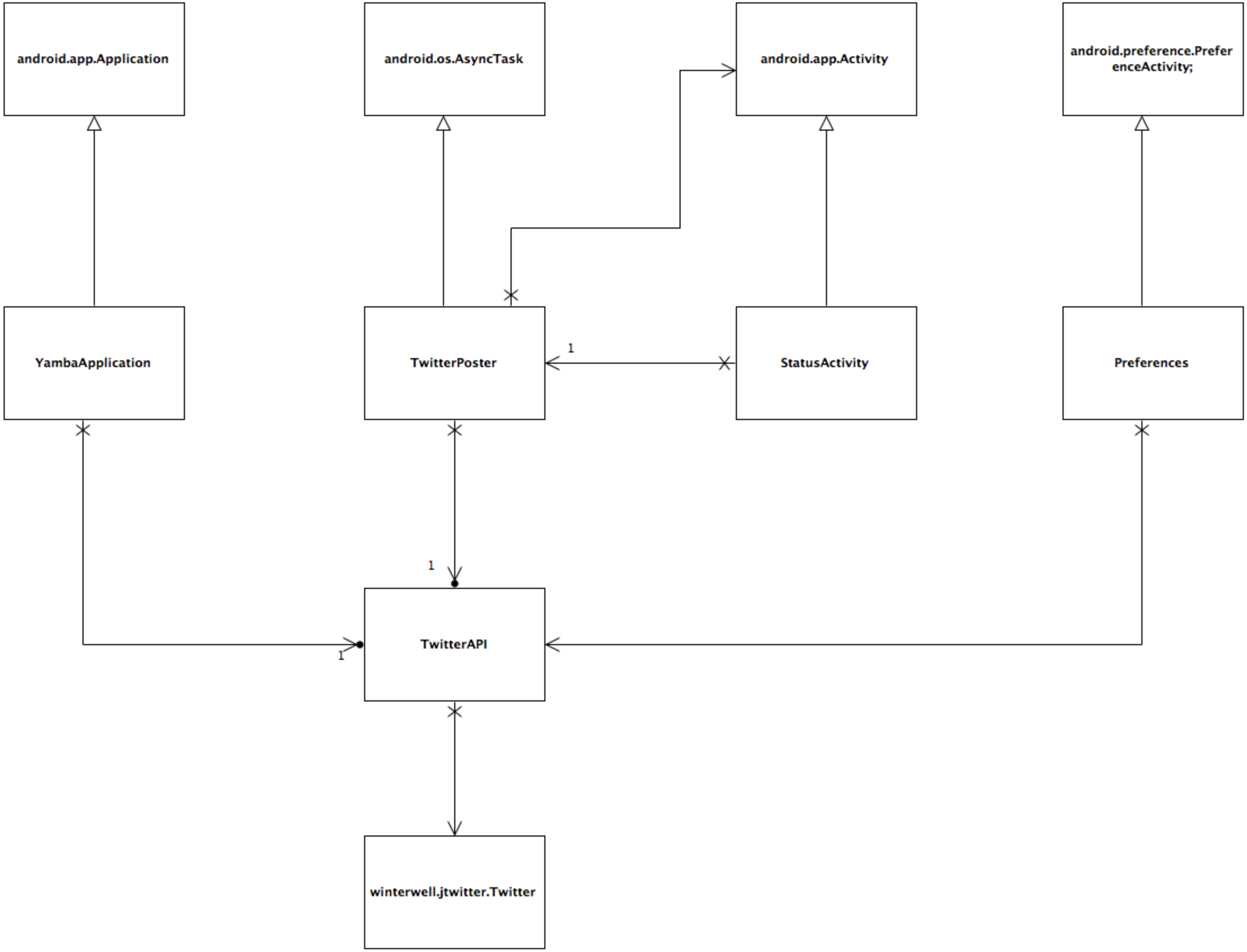
Introduce ApplicationObject. Move Preferences and API object to this …

edeleastar authored 2 days ago

- TwitterPoster AsyncTask will now use the API object, accessed as a property of the Application Object



```
class TwitterPoster extends AsyncTask<String, Integer, String>
{
  val TwitterAPI twitter
  val Activity activity

  new(Activity activity)
  {
    var app = activity.getApplication() as YambaApplication
    this.twitter = app.twitter
    this.activity = activity
  }

  override doInBackground(String... it)
  {
    try
    {
      var status = twitter.updateStatus(get(0))
      status
    }
    catch (TwitterException e)
    {
      Log.e("YAMBA", e.toString());
      "Failed to post";
    }
  }

  override onPostExecute(String result)
  {
    Toast.makeText(activity, result, Toast.LENGTH_LONG).show();
  }
}
```


android.app.Application
android.os.AsyncTask
android.app.Activity
android.preference.PreferenceActivity;
YambaApplication
TwitterPoster
1
StatusActivity
Preferences
1
TwitterAPI
1
winterwell.jtwitter.Twitter

Waterford Institute of Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

eLearning
support unit