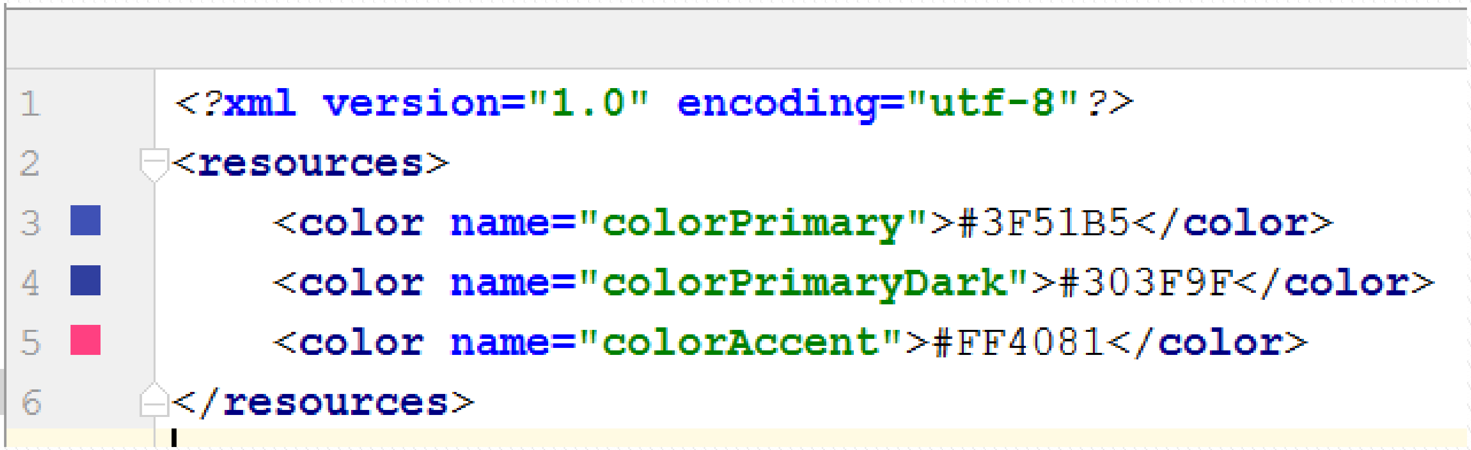
Programming mobile devices

Basic GUI elements – part 2

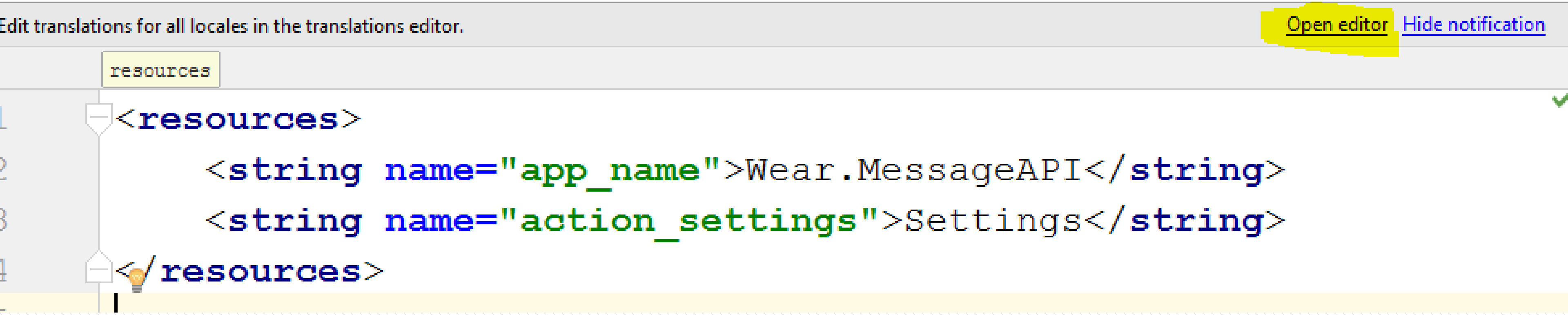
Resource files

- Items:
  - Strings
  - String arrays
  - Colors
  - Dimensions
  - Background colors
  - etc.



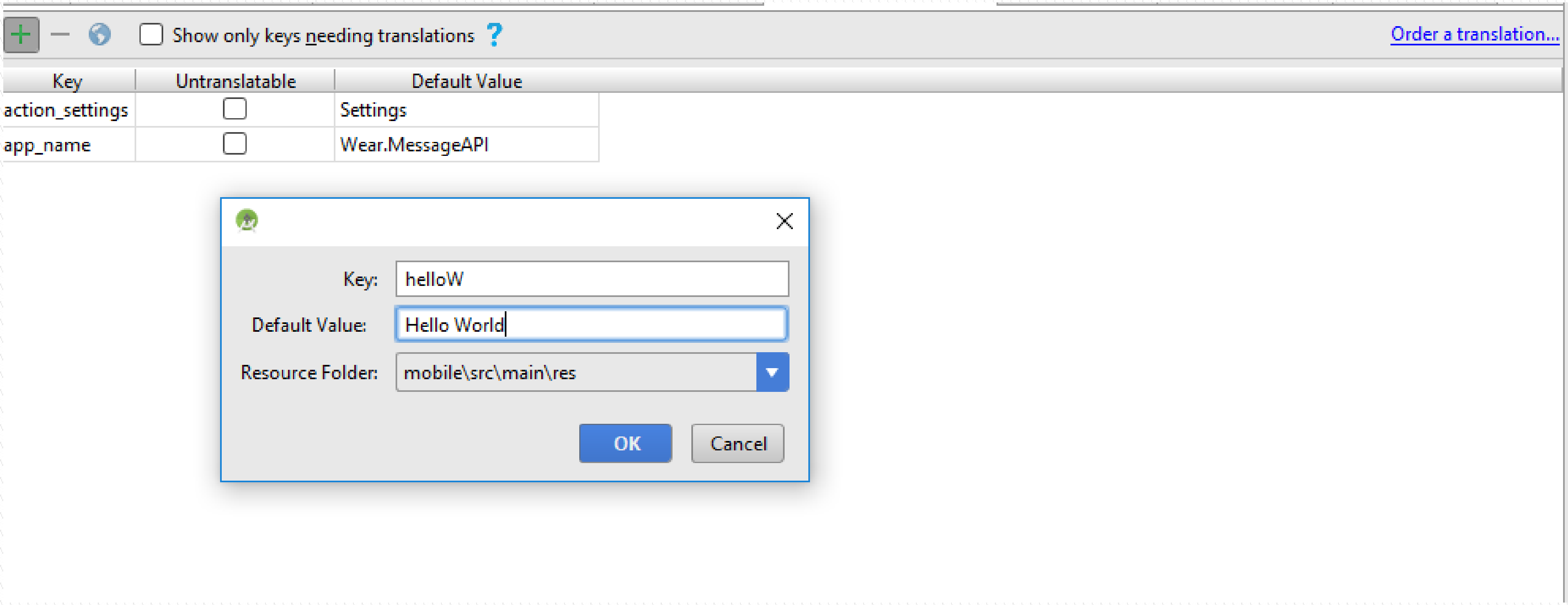
```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#3F51B5</color>
    <color name="colorPrimaryDark">#303F9F</color>
    <color name="colorAccent">#FF4081</color>
</resources>
```

String editor



Edit translations for all locales in the translations editor.

```
<resources>
    <string name="app_name">Wear.MessageAPI</string>
    <string name="action_settings">Settings</string>
</resources>
```



Key	Untranslatable	Default Value
action_settings	☐	Settings
app_name	☐	Wear.MessageAPI

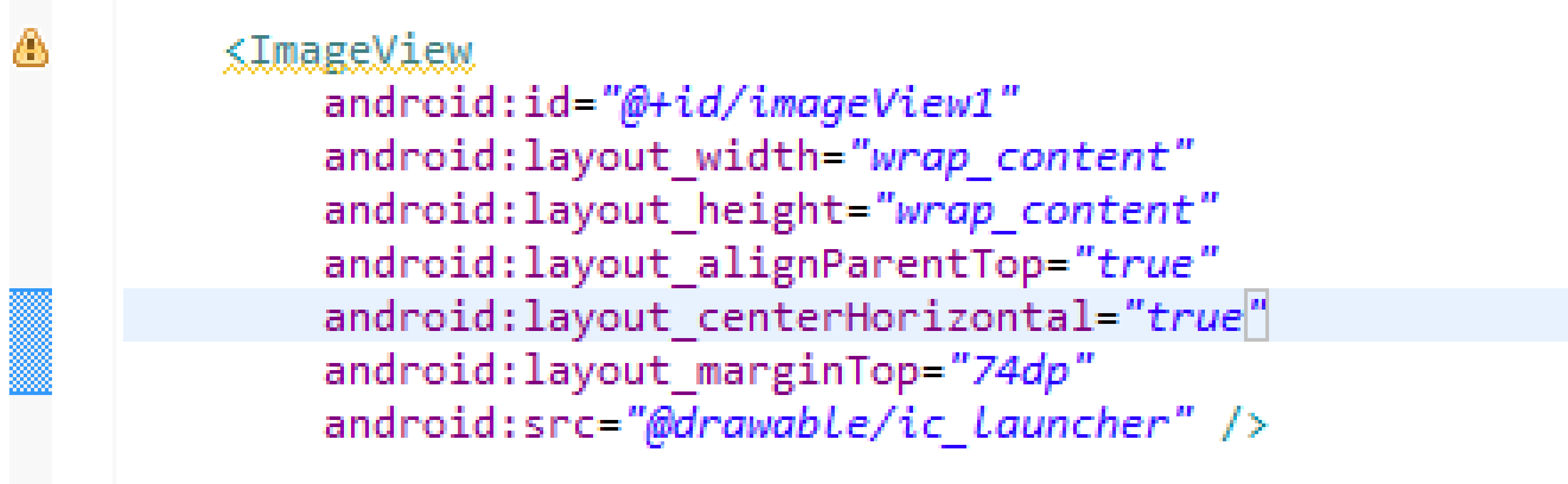
Key: helloW

Default Value: Hello World

Resource Folder: mobile\src\main\res

ImageView

- This is a component for displaying images on activity.
- Images can be stored as resource files (in the drawable directory)



```
<ImageView
    android:id="@+id/imageView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="74dp"
    android:src="@drawable/ic_launcher" />
```

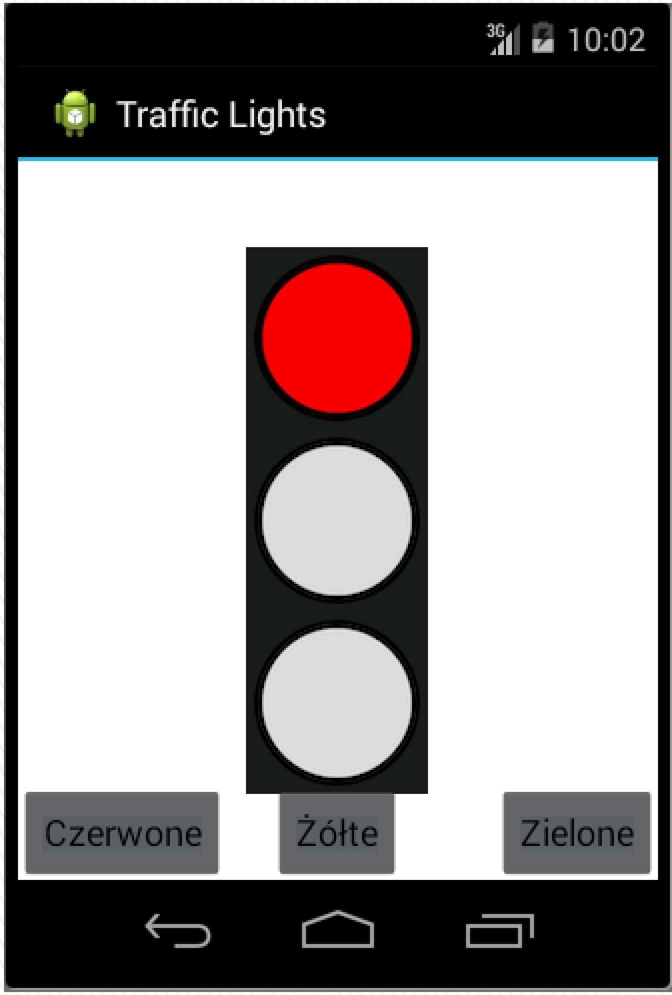
- To change the displayed image use:

```
ImageView im = (ImageView) findViewById(R.id.imageView1);
im.setImageResource(R.drawable.testimage);
```

Exercise 1

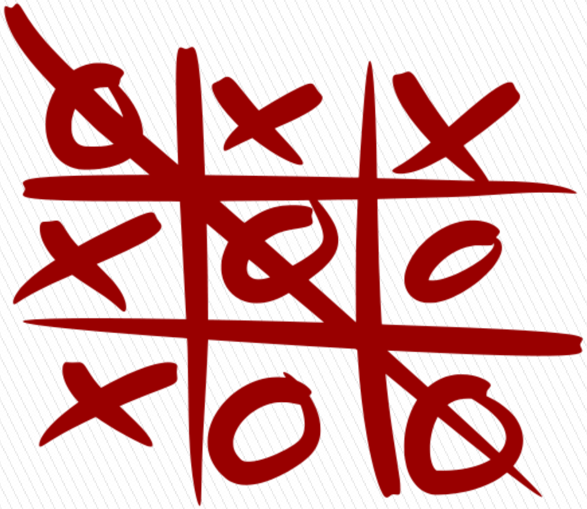
- Write an application simulating the behavior of the typical traffic lights.
- The application should use the ImageView and Button components.
- When you click on the button, the corresponding light will light up.
- Take care of error control – e.g. you can not light three lights at the same time.
- To change the presented picture use the **setImageResource** method
- Pictures:

www.tomaszx.pl/materialy/swiatla_drogowe_obrazki.zip



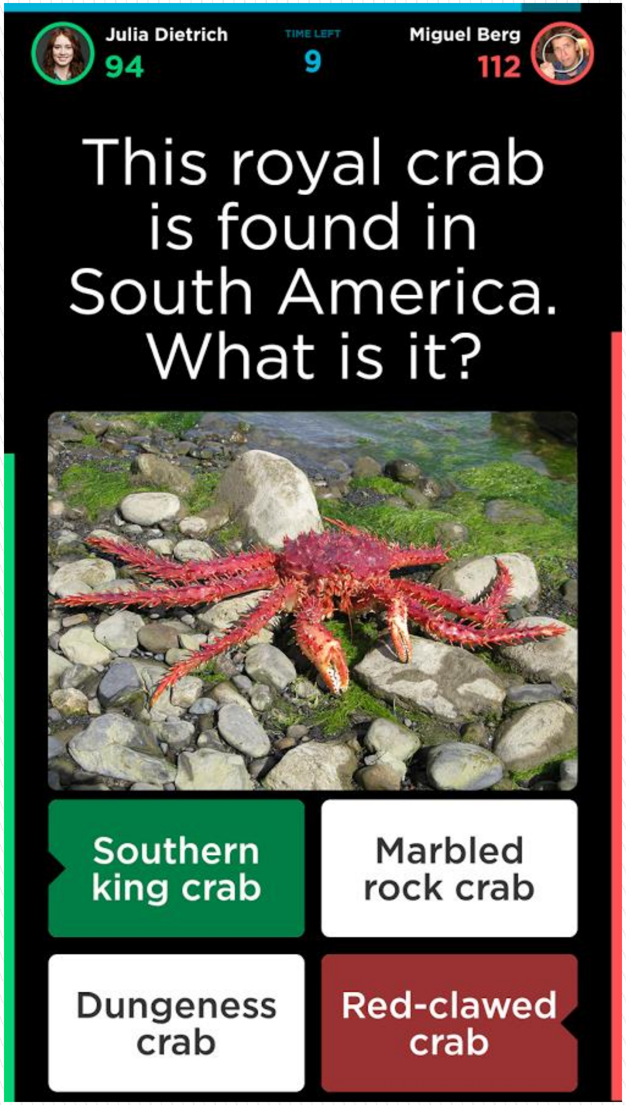
Exercise 2

- Implement a game of tic-tac-toe.
- Display a 3 x 3 grid (for two players).
- When you click on a field – it should change to the **nought** (O) or **cross** (X) of the corresponding player.
- The game should count the number of wins per player.



Exercise 3

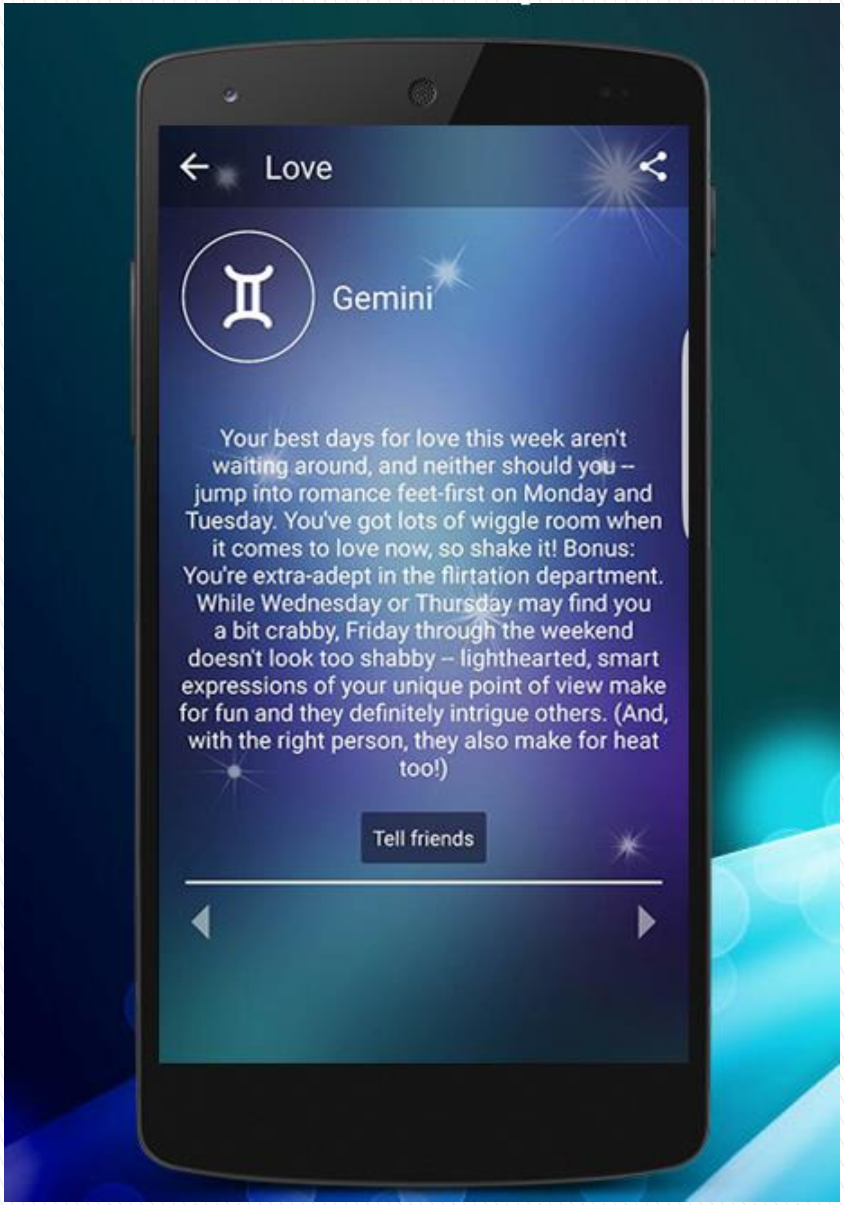
- Prepare a quiz app for your chosen topic.
- The application will display a picture of the selected object, and the user has to type what item is it (or select from several options suggested by the application).
- Choose any topic you like: Famous people, toy brands, business logos, etc.
- When writing text – make sure that the small / upper case does not cause the response to be ignored.



This royal crab is found in South America. What is it?

Exercise 4

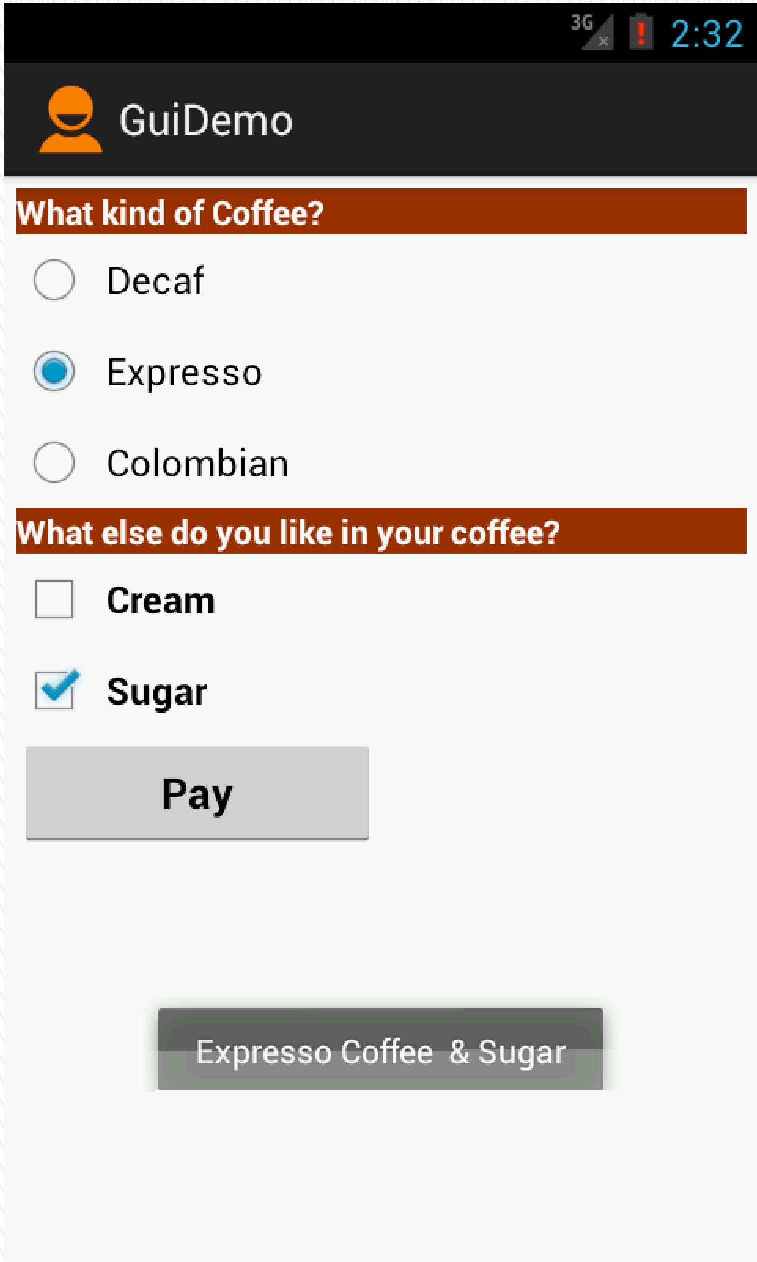
- Make a simple horoscope application.
- Add a zodiac sign to the strings.xml file.
- On the activity add an item asking the user for the date of birth. Use the application logic to check his zodiac sign.
- Display him a horoscope selected for his zodiac sign.



Your best days for love this week aren't waiting around, and neither should you -- jump into romance feet-first on Monday and Tuesday. You've got lots of wiggle room when it comes to love now, so shake it! Bonus: You're extra-adept in the flirtation department. While Wednesday or Thursday may find you a bit crabby, Friday through the weekend doesn't look too shabby -- lighthearted, smart expressions of your unique point of view make for fun and they definitely intrigue others. (And, with the right person, they also make for heat too!)

Exercise 5

- Based on the coffee selection application, design a program to order a pizza.
- Required functionality:
  - Choice of dough (thin, thick)
  - Choosing the size of the pizza (small, medium, large)
  - Selection of minimum three basic ingredients from the list available (ham, cheese, mushrooms, olives, bacon, chicken, onion)
  - Optionally up to two additional ingredients (garlic, salami, shrimp, capers, tuna, tomato sauce, garlic sauce, oregano)
  - Each component has a price to be included in the cost of the order.
  - The application is supposed to display the message "**A small pizza on a thick dough, consisting of ham, cheese and mushrooms, was ordered. The price of this pizza is 40 PLN**"
  - It will be useful to use a ScrollView component.



What kind of Coffee?

○ Decaf

● Expresso

○ Colombian

What else do you like in your coffee?

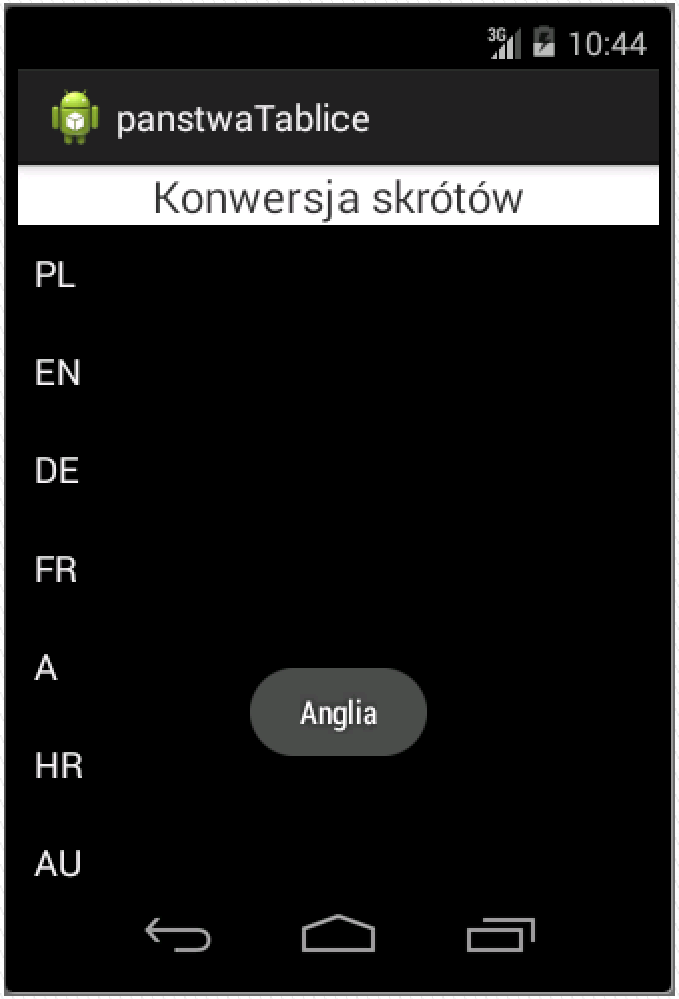
□ Cream

☑ Sugar

Expresso Coffee & Sugar

Exercise 6

- Write a program that transforms the state abbreviation (located on the car's license plate) into its name.
- The program must use the ListView component.
- Data on abbreviations and names of states are to be stored in two tables,
- `private String state[] = {„Poland”, „Germany”};`
- `private String abbr[] = {„PL”, „DE”};`



10:44

panstwaTablice

Konwersja skrótów

PL

EN

DE

FR

A

HR

AU

Anglia

Exercise 6a

- Modify the previous task to use an array of strings defined in the strings.xml resource file.
- To read data use:

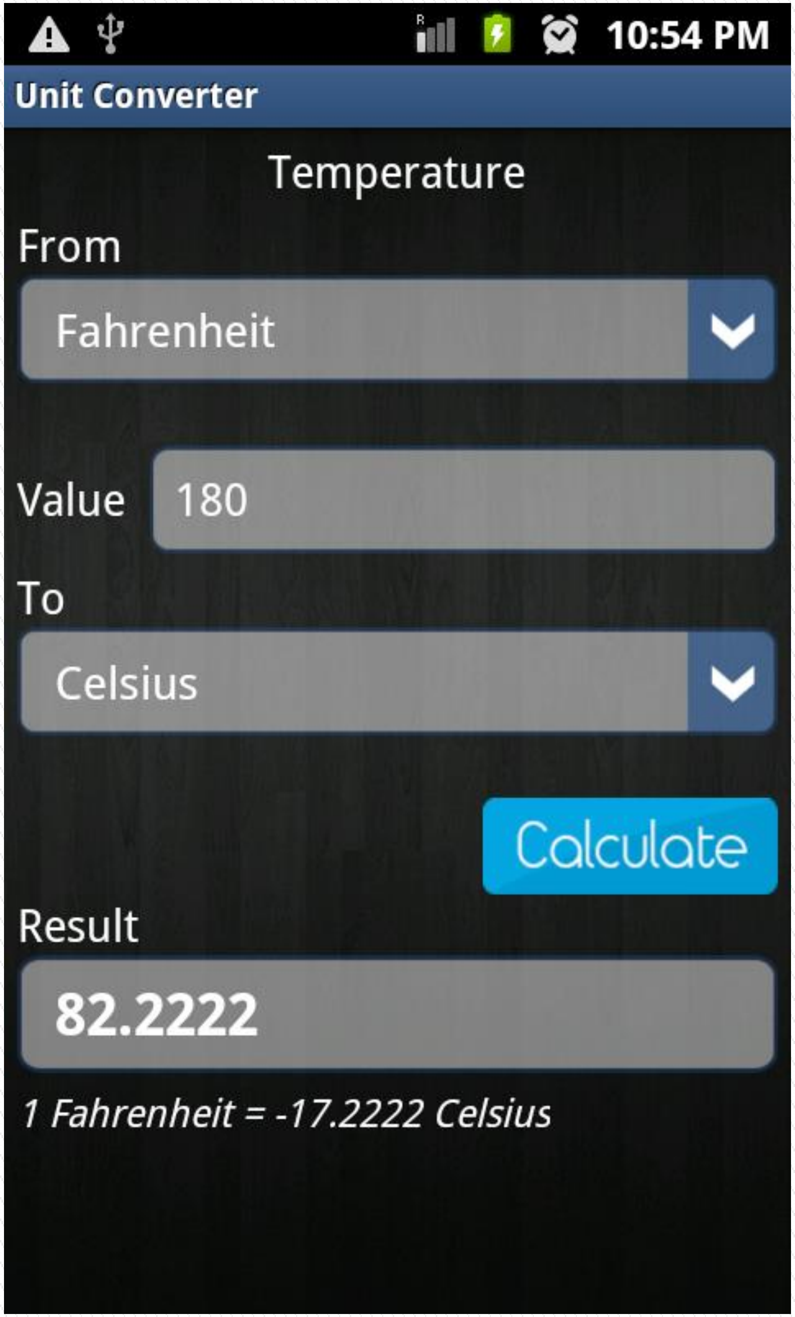
```
private String countries[];
countries =
getResources().getStringArray(
R.array.countries_array);
```

strings.xml:

```
<string-array
name="countries_array">
    <item>Polska</item>
    <item>Anglia</item>
    <item>Niemcy</item>
    <item>Francja</item>
    <item>Austria</item>
    <item>Chorwacja</item>
    <item>Ukraina</item>
    <item>Węgry</item>
</string-array>
<string-array
name="shortcuts_array">
    <item>PL</item>
    <item>EN</item>
    <item>DE</item>
    <item>FR</item>
    <item>A</item>
    <item>HR</item>
    <item>AU</item>
    <item>H</item>
</string-array>
```

Exercise 7

- Write a simple unit converter
- The user chooses the type of conversion, resulting in the creation of a new activity.
- Valid conversions include: temperature, distance, capacity



From: Fahrenheit

Value: 180

To: Celsius

Result: 82.2222

1 Fahrenheit = -17.2222 Celsius