



Exercise session 1

11.09.2018

Sergey Ryabov

Are you ready?

Do you have Android Studio installed?

Have you created and ran your first project?

If not yet, please, do exercise 0 first to set up your environment

Install Android Studio and have “Hello world” running on your phone:

<http://bit.ly/2LZVRZ4>

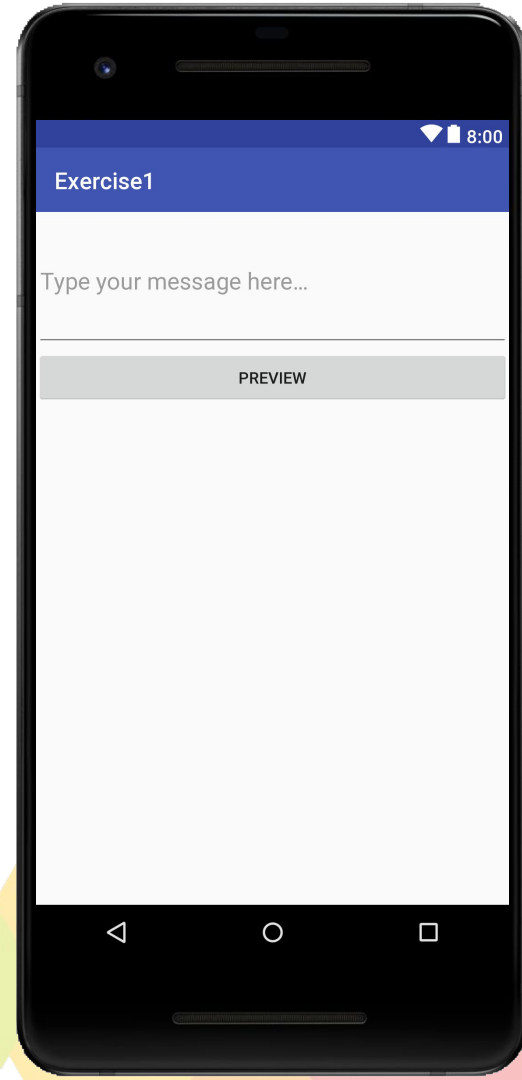


Overview

We'll create a simple app with two activities

First activity with an EditText view and a “Preview” button

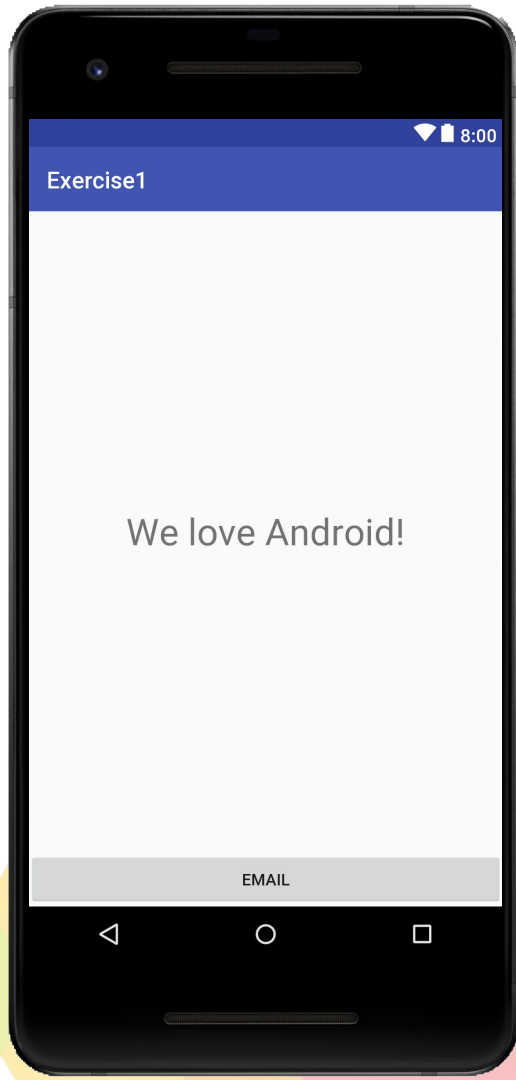
The user enters a message into the EditText and hits
“Preview”



Overview

This opens a second activity with “Email” button and a TextView that displays the message.

You can check the text and if it’s good enough, you can send it via email app by clicking “Email” button



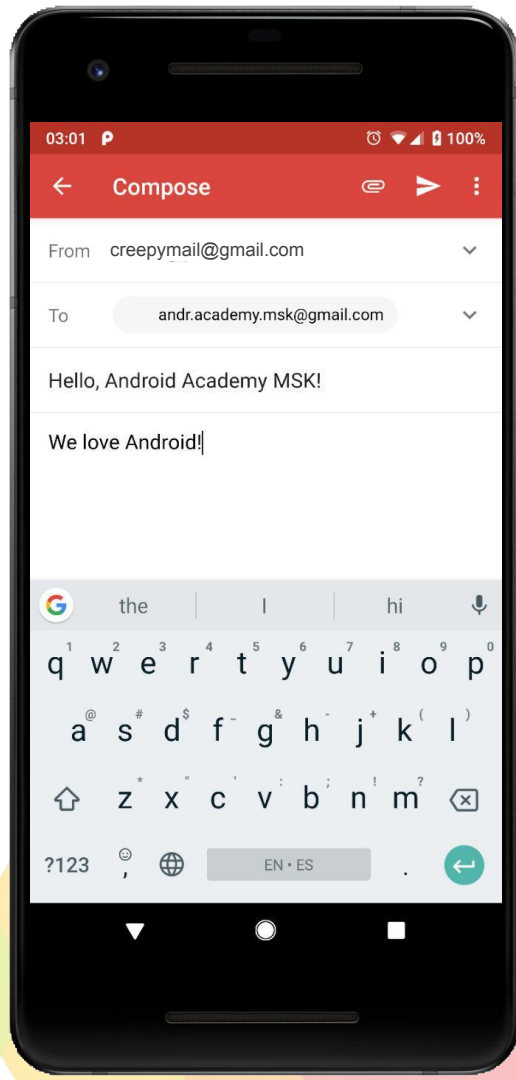
Overview

Clicking on the “Email” button

opens the email app which is installed on the device

(e.g. gmail)

with the message as a new email’s content.



Let's Begin

Solution will be available here: <https://bit.ly/2x1ZNno> in one week, before the next lecture.

Now it's time for you to make progress!!



Step 1 - Create a new project

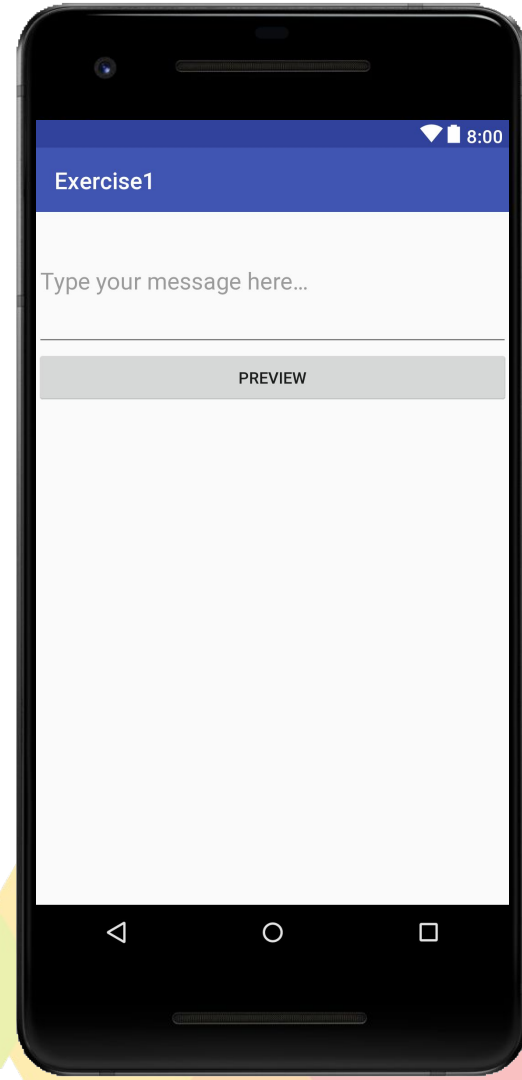
For help go to the lecture slides: <https://goo.gl/sQpvP4>



Step 2

Create the main activity layout, something like this

See next slide for more info



About EditText

EditText is a View that allows the user to enter text.

When the user clicks on it, the keyboard opens.

- Using view attribute “inputType”, you can indicate which keyboard should pop up (only numbers, all caps, email etc...)

Documentation:

<https://developer.android.com/reference/android/widget/EditText.html>

About EditText

“hint” attribute is the text that will be shown when no text was entered.

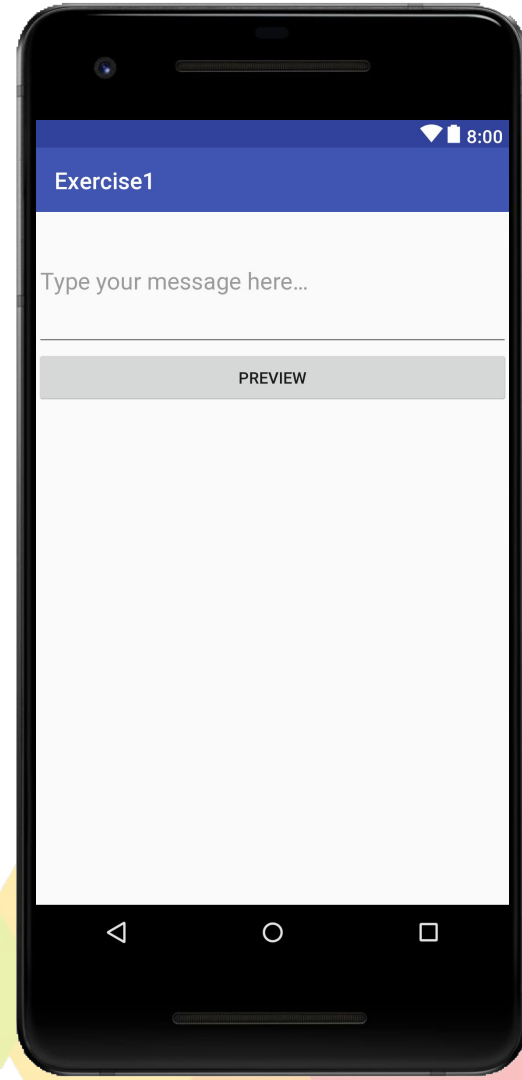
To get the entered text dynamically at runtime use
`editText.getText().toString()`.



Step 2 - Now you can

Create the main activity layout.

Feel free to edit the design
with what we learned in class
(don't waste too much time on it, we have more stuff to do :)

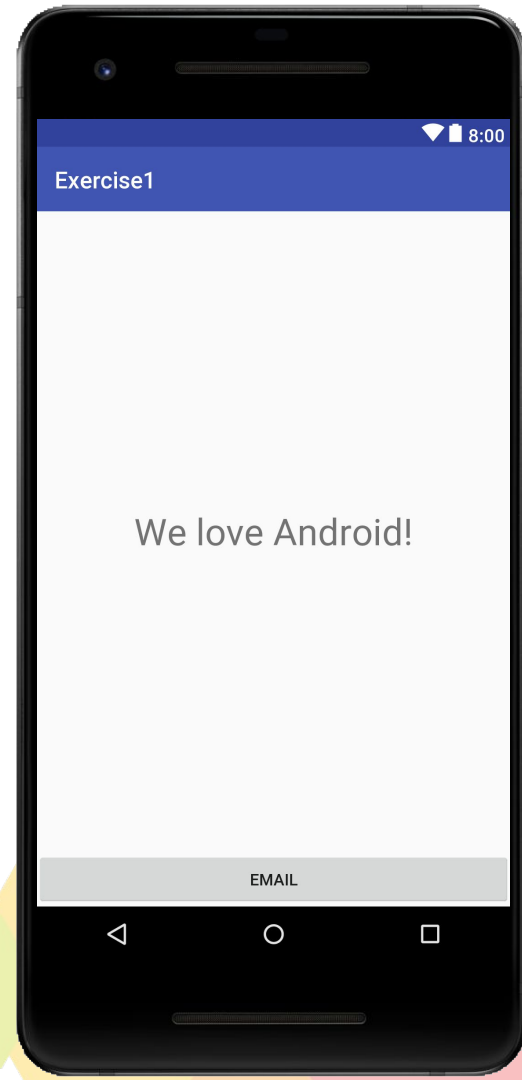


Step 3

On **Preview** button click:

Open a new activity with a TextView which shows the message you typed in on the previous activity.

[want more guidance? See steps 1-3 on next slides]



Step 3 - steps[1/3]

1. Create a new Activity class (Empty Activity)
2. Edit the activity's layout as needed



Step 3 - steps[2/3]

3. Set `OnClickListener` to your button and implement the code:

Create an intent that starts the new activity and passes the text from the `EditText` through the intent's extras, like in the lecture slides.



Step 3 - steps[3/3]

4. On the second activity: get the message from the intent's extras
5. On the second activity: show the message in the activity's TextView

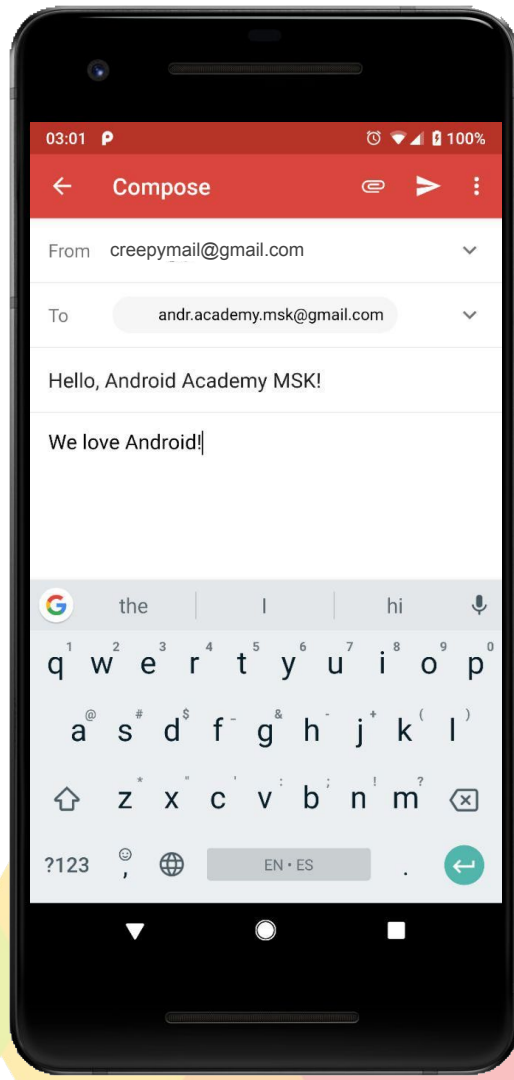


Step 4

On **Email** button click:

- Open a new email activity
- Pre-fill the email “To” with your email address
- Pre-fill the email “Subject” with some predefined text
- Pre-fill the email “Body” with your message

See next slide for more info



Open an Email App

In the lecture we used an **explicit intent** to open a specific activity which we created and knew about.

What happens when we want to start just any activity that can perform our task? I.e. any activity that can send an email or take a photo or dial a number or navigate.

We can use an **implicit intent**. We just tell what we want to do and Android will suggest the apps those can provide that functionality.



Implicit Intents

Use this:

<https://developer.android.com/guide/components/intents-common.html>

To find out how to create an email implicit intent and which keys to add to the intent bundle.

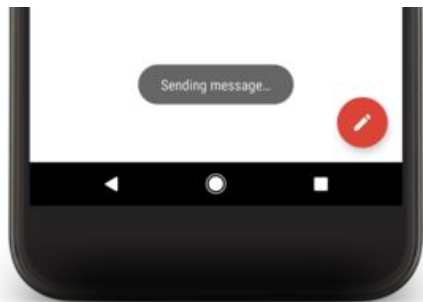
Hint: Look for SEND_TO action

WARNING: pay attention to this caution notice

! **Caution:** If there are no apps on the device that can receive the implicit intent, your app will crash when it calls `startActivity()`. To first verify that an app exists to receive the intent, call `resolveActivity()` on your `Intent` object. If the result is non-null, there is at least one app that can handle the intent and it's safe to call `startActivity()`. If the result is null, you should not use the intent and, if possible, you should disable the feature that invokes the intent.

Toasts

If there is no Email app on the user's device, show him some message about this. So that he won't be surprised if nothing happened after his click.



Toast is a simple feedback about an operation in a transient small popup.

The activity remains interactive - whereas the toast itself isn't.

```
Toast.makeText(this, "No Email app found", Toast.LENGTH_LONG).show();
```

BEWARE MOST COMMON BUG: don't forget to call `.show()` to actually display it, not only create ;)

<http://developer.android.com/guide/topics/ui/notifiers/toasts.html>

DONE?

That's amazing! Great job!!

But we have some bonus tasks if you wanna try.

If you have even a small question - don't forget to ask the mentors in class or on Slack.



Bonus

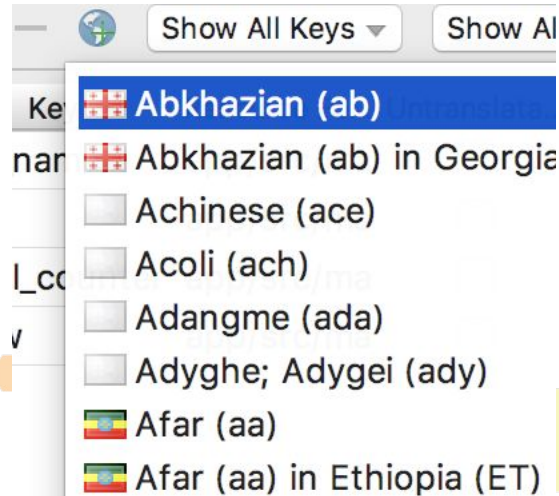
1. Make sure your all of your strings and dimensions are not hardcoded in XML files, but rather are extracted to the corresponding resource XML files (strings.xml, dims.xml)
2. Create a different layout for landscape as we saw in the lecture.
3. Make your app Russian (or other language) compatible. See next slide for details.



Localization [1/2]

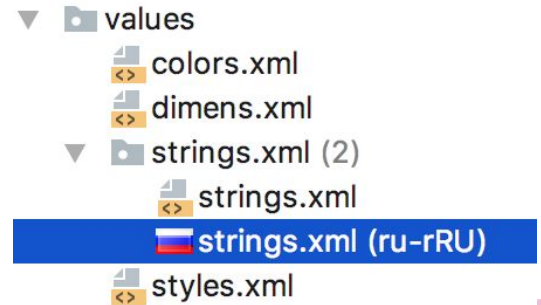
To create other language localization:

- Open strings.xml
- On the top right : “open editor”
- On the top left: click the globe icon and select the language you want.



Localization [2/2]

- Fill in the translations in the editor
- In the left Project structure panel find the new *strings.xml* that was created with the new locale. Click on it to explore the xml.
- Change your device's language settings (if not in Russian yet)
- Run and check your Russianized app!
- Check out real folder structure of your project to understand the concept



Awesome!

You're all done with
exercise 1!

See you next time!

