Implizit Intents & Ressourcen

Prof. Dr.-Ing. V.Brovkov

Building an Intent

- There are two types of intents:
  - **Explicit intents** specify the component to start by name (the fully-qualified class name). You'll typically use an explicit intent to start a component in your own app, because you know the class name of the activity or service you want to start.
    - For example, start a new activity in response to a user action or start a service to download a file in the background.
  - **Implicit intents** do not name a specific component, but instead declare a general action to perform, which allows a component from another app to handle it.
    - For example, if you want to show the user a location on a map, you can use an implicit intent to request that another capable app show a specified location on a map.
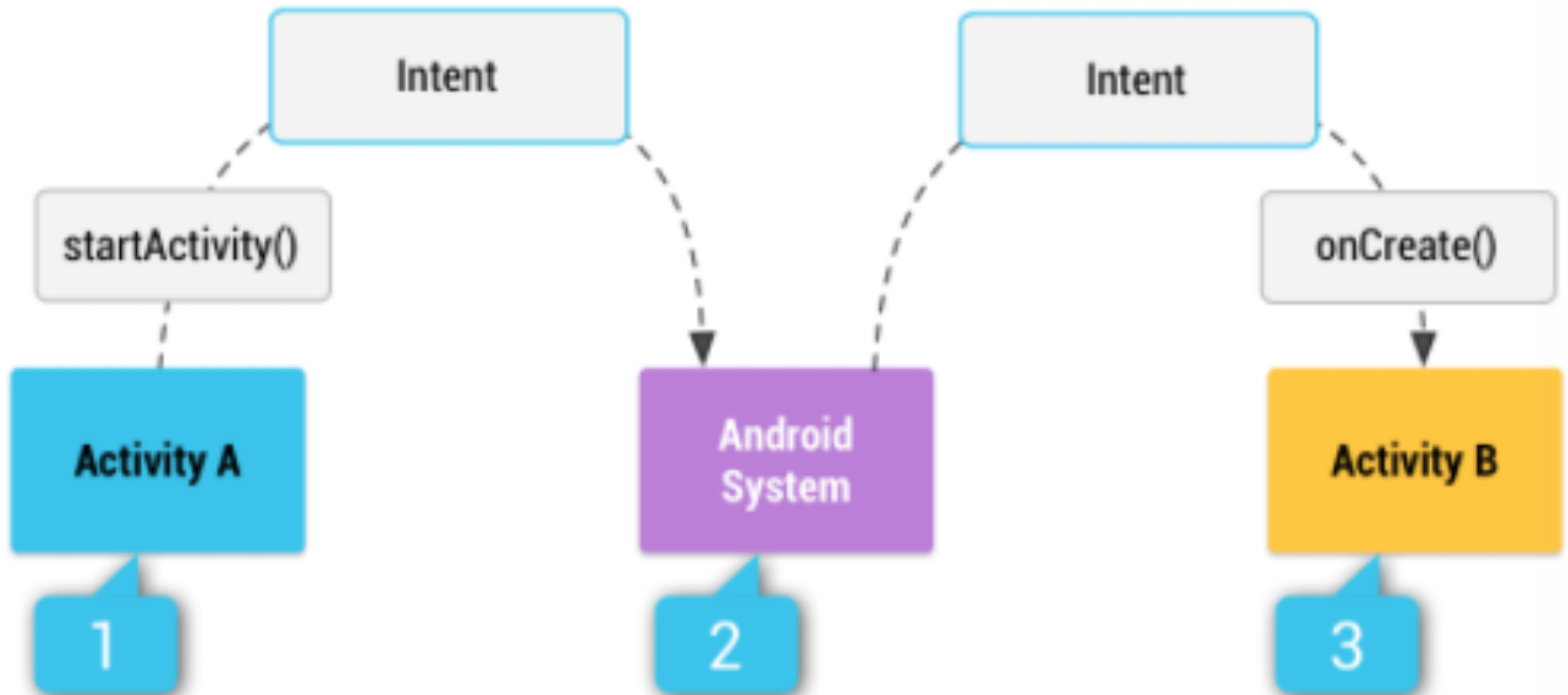
Building an Intent

- When you create an implicit intent, the Android system finds the appropriate component to start by comparing the contents of the intent to the intent filters declared in the manifest file of other apps on the device.

- If the intent matches an intent filter, the system starts that component and delivers it the Intent object.

- If multiple intent filters are compatible, the system displays a dialog so the user can pick which app to use.
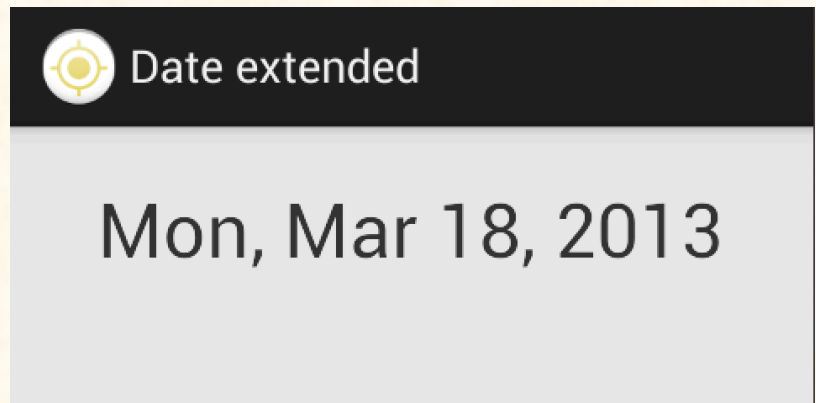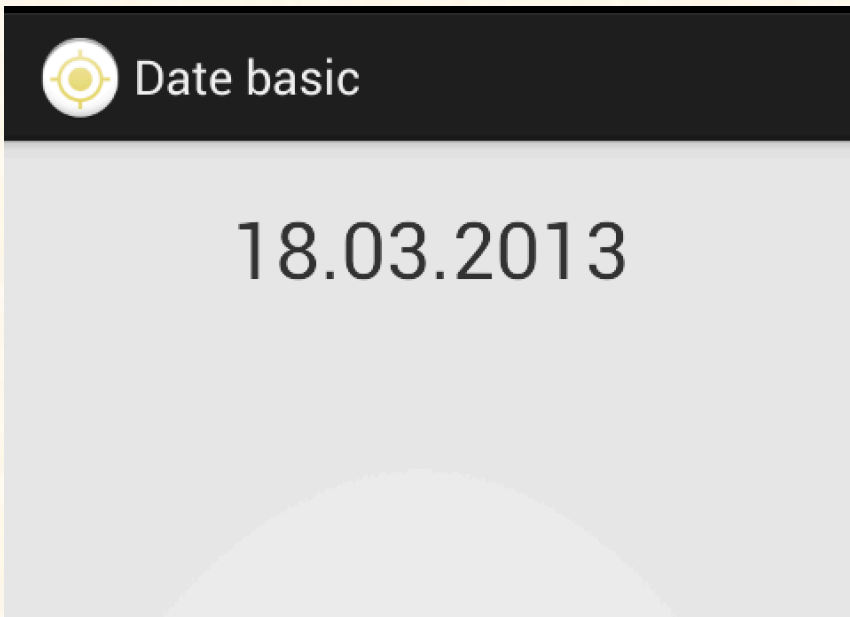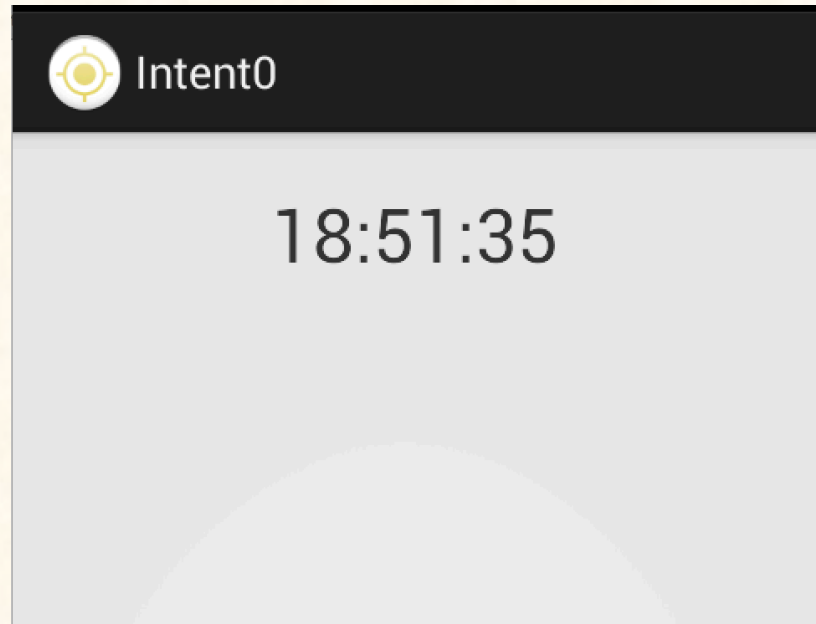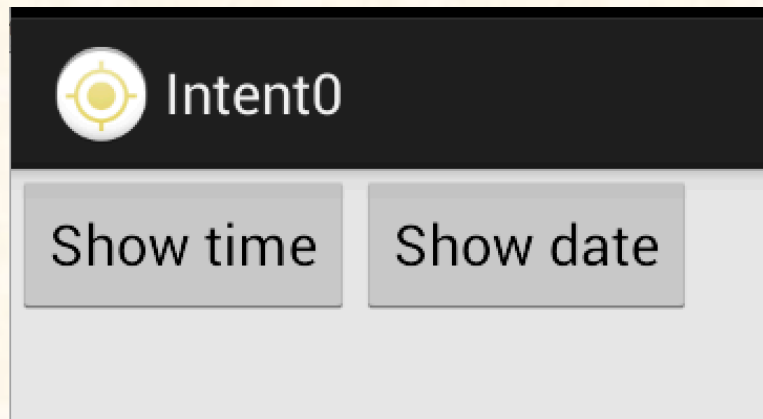
Building an Intent

Implicit Intent & Lokale Activities.

- Projekt Intent0

Lokale Ressourcen Zugriff mit URI 1

Lokale Ressourcen Zugriff mit URI 2

```java
10  public class MainActivity extends Activity implements OnClickListener {
11
12      /** Called when the activity is first created. */
13      @Override
14      public void onCreate(Bundle savedInstanceState) {
15          super.onCreate(savedInstanceState);
16          setContentView(R.layout.activity_main);
17
18          Button btnTime = (Button) findViewById(R.id.btnTime);
19          Button btnDate = (Button) findViewById(R.id.btnDate);
20
21          btnTime.setOnClickListener(this);
22          btnDate.setOnClickListener(this);
23      }
24
```

Lokale Ressourcen Zugriff mit URI 3

```
25    @Override
26    public void onClick(View v) {
27      Intent intent;
28
29      switch(v.getId()) {
30      case R.id.btnTime:
31        intent = new Intent("ru.startandroid.intent.action.showtime");
32        startActivity(intent);
33        break;
34      case R.id.btnDate:
35        intent = new Intent("ru.startandroid.intent.action.showdate");
36        startActivity(intent);
37        break;
38      }
39    }
40 }
```

Lokale Ressourcen Zugriff mit URI 4

```
10  public class ActivityTime extends Activity {
11    @Override
12    protected void onCreate(Bundle savedInstanceState) {
13      super.onCreate(savedInstanceState);
14      setContentView(R.layout.time);
15
16      SimpleDateFormat sdf = new SimpleDateFormat("HH:mm:ss");
17      String time = sdf.format(new Date(System.currentTimeMillis()));
18
19      TextView tvTime = (TextView) findViewById(R.id.tvTime);
20      tvTime.setText(time);
21    }
22  }
```

Lokale Ressourcen Zugriff mit URI 5

```java
10  public class ActivityDate extends Activity {
11    @Override
12    protected void onCreate(Bundle savedInstanceState) {
13      super.onCreate(savedInstanceState);
14      setContentView(R.layout.date);
15
16      SimpleDateFormat sdf = new SimpleDateFormat("dd.MM.yyyy");
17      String date = sdf.format(new Date(System.currentTimeMillis()));
18
19      TextView tvDate = (TextView) findViewById(R.id.tvDate);
20      tvDate.setText(date);
21    }
22  }
```

Lokale Ressourcen Zugriff mit URI 6

```java
10  public class ActivityDateEx extends Activity {
11    @Override
12    protected void onCreate(Bundle savedInstanceState) {
13      super.onCreate(savedInstanceState);
14      setContentView(R.layout.date);
15
16      SimpleDateFormat sdf = new SimpleDateFormat("EEE, MMM d, yyyy");
17      String date = sdf.format(new Date(System.currentTimeMillis()));
18
19      TextView tvDate = (TextView) findViewById(R.id.tvDate);
20      tvDate.setText(date);
21    }
22  }
```

Lokale Ressourcen Zugriff mit URI 7

Lokale Ressourcen Zugriff mit URI 8

Lokale Ressourcen Zugriff mit URI 9

```xml
1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      package="ua.opu.brovkov.intent0"
4      android:versionCode="1"
5      android:versionName="1.0" >
6      <uses-sdk
7          android:minSdkVersion="7"
8          android:targetSdkVersion="17" />
9      <application
10         android:allowBackup="true"
11         android:icon="@drawable/ic_launcher"
12         android:label="@string/app_name"
13         android:theme="@style/AppTheme" >
14         <activity
15             android:name="ua.opu.brovkov.intent0.MainActivity"
16             android:label="@string/app_name" >
17             <intent-filter>
18                 <action android:name="android.intent.action.MAIN" />
19
20                 <category android:name="android.intent.category.LAUNCHER" />
21             </intent-filter>
22         </activity>
23         <activity android:name="ActivityTime">
24             <intent-filter>
25                 <action android:name="ru.startandroid.intent.action.showtime"></action>
26                 <category android:name="android.intent.category.DEFAULT"></category>
27             </intent-filter>
28         </activity>
29         <activity android:name="ActivityDate" android:label="Date basic">
30             <intent-filter>
31                 <action android:name="ru.startandroid.intent.action.showdate"></action>
32                 <category android:name="android.intent.category.DEFAULT"></category>
33             </intent-filter>
34         </activity>
35         <activity android:name="ActivityDateEx" android:label="Date extended">
36             <intent-filter>
37                 <action android:name="ru.startandroid.intent.action.showdate"></action>
38                 <category android:name="android.intent.category.DEFAULT"></category>
39             </intent-filter>
40         </activity>
41     </application>
42  </manifest>
```

Lokale Ressourcen Zugriff mit URI 10

```xml
<activity
    android:name="ua.opu.brovkov.intent0.MainActivity"
    android:label="@string/app_name" >
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
<activity android:name="ActivityTime">
    <intent-filter>
        <action android:name="ru.startandroid.intent.action.showtime"></action>
        <category android:name="android.intent.category.DEFAULT"></category>
    </intent-filter>
</activity>
<activity android:name="ActivityDate" android:label="Date basic">
    <intent-filter>
        <action android:name="ru.startandroid.intent.action.showdate"></action>
        <category android:name="android.intent.category.DEFAULT"></category>
    </intent-filter>
</activity>
<activity android:name="ActivityDateEx" android:label="Date extended">
    <intent-filter>
        <action android:name="ru.startandroid.intent.action.showdate"></action>
        <category android:name="android.intent.category.DEFAULT"></category>
    </intent-filter>
</activity>
```
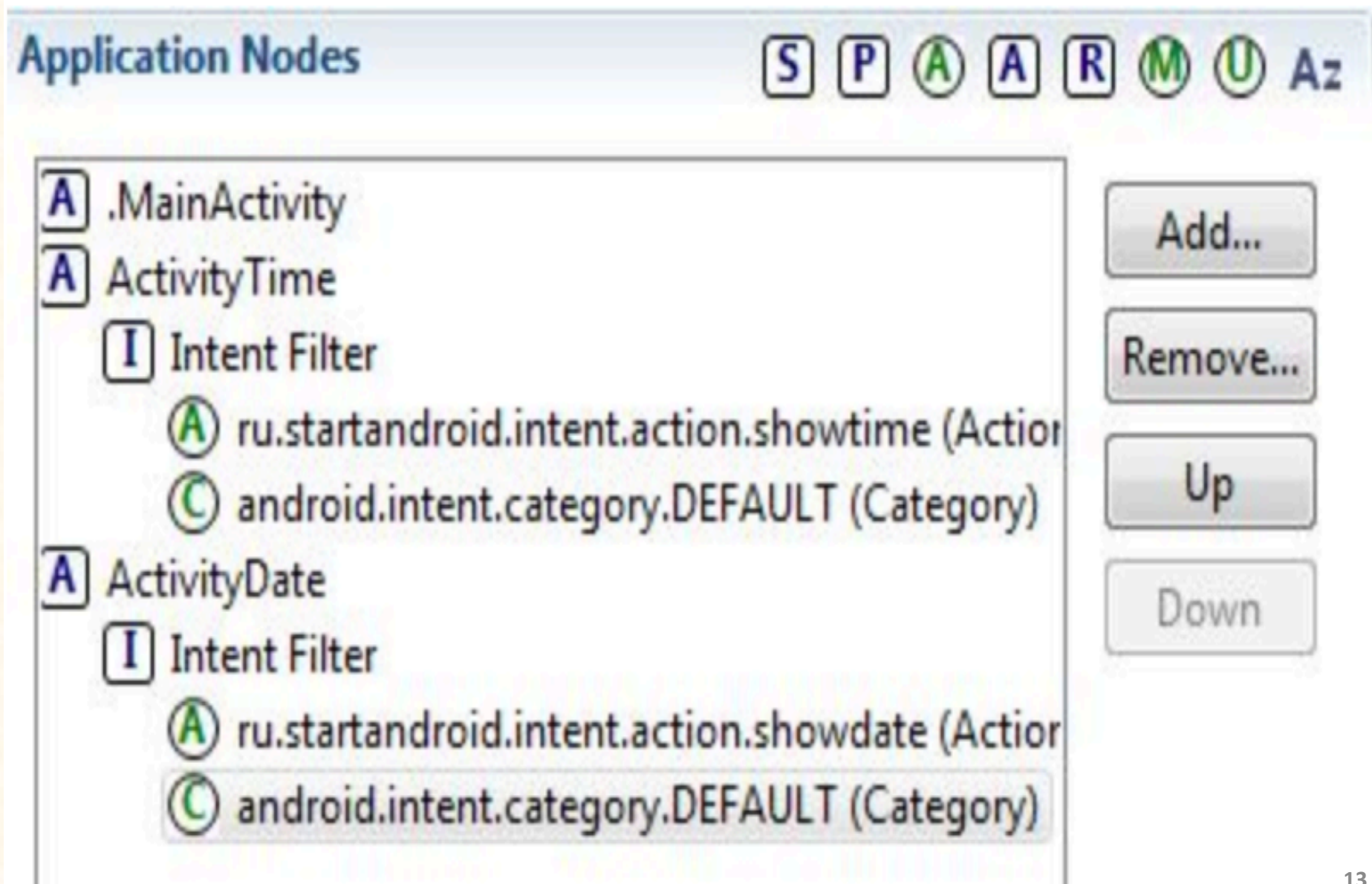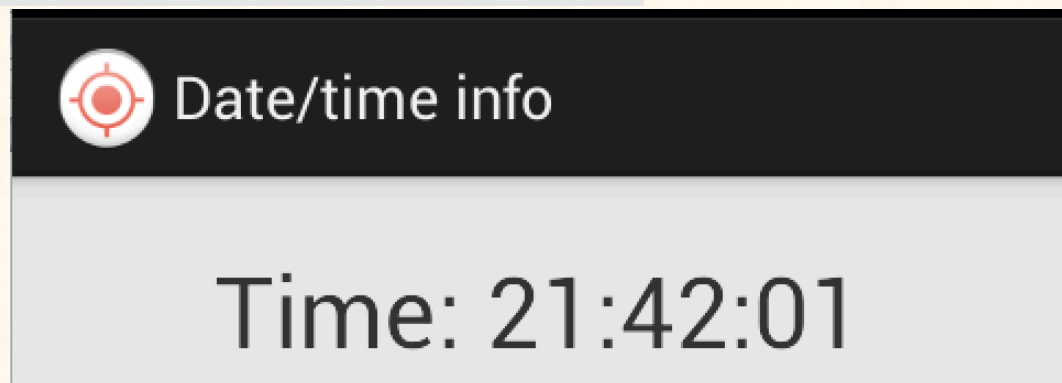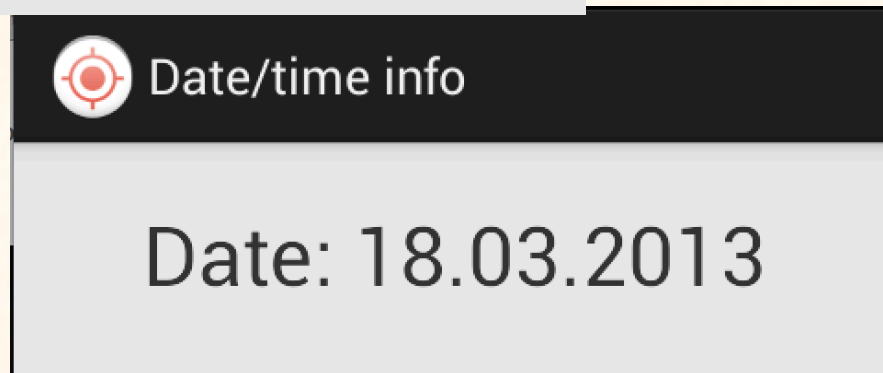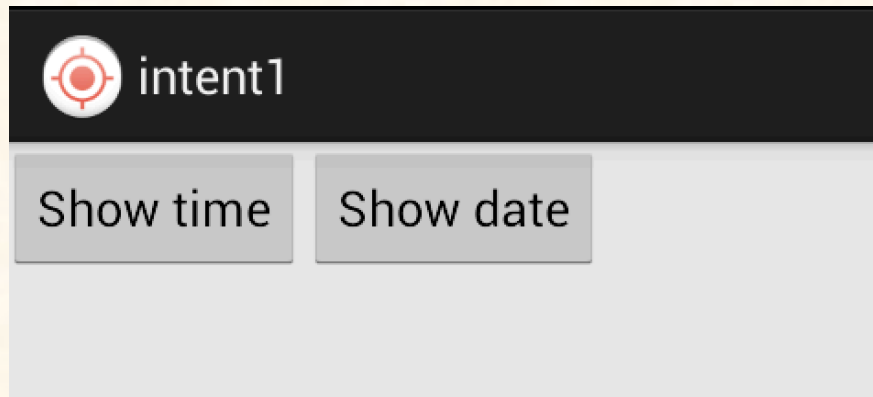
Lokale Ressourcen Zugriff mit URI 11

Zwei Intents in einer Activity
&
URI Datenzugriff 1

- Projekt Intent1

Zwei Intents in einer Activity & URI Datenzugriff 1

Zwei Intents in einer Activity & URI Datenzugriff 2

```java
@Override
public void onClick(View v) {
  Intent intent;

  switch(v.getId()) {
  case R.id.btnTime:
    intent = new Intent("ru.startandroid.intent.action.showtime");
    startActivity(intent);
    break;
  case R.id.btnDate:
    intent = new Intent("ru.startandroid.intent.action.showdate");
    startActivity(intent);
    break;
  }
```

Zwei Intents in einer Activity & URI Datenzugriff 3

```java
11  public class Info extends Activity {
12⊝   @Override
13    protected void onCreate(Bundle savedInstanceState) {
14      super.onCreate(savedInstanceState);
15      setContentView(R.layout.info);
16      // получаем Intent, который вызывал это Activity
17      Intent intent = getIntent();
18      // читаем из него action
19      String action = intent.getAction();
20      String format = "", textInfo = "";
21      // в зависимости от action заполняем переменные
22      if (action.equals("ru.startandroid.intent.action.showtime")) {
23        format = "HH:mm:ss";
24        textInfo = "Time: ";
25      }
26      else if (action.equals("ru.startandroid.intent.action.showdate")) {
27        format = "dd.MM.yyyy";
28        textInfo = "Date: ";
29      }
30      // в зависимости от содержимого переменной format
31      // получаем дату или время в переменную datetime
32      SimpleDateFormat sdf = new SimpleDateFormat(format);
33      String datetime = sdf.format(new Date(System.currentTimeMillis()));
34
35      TextView tvDate = (TextView) findViewById(R.id.tvInfo);
36      tvDate.setText(textInfo + datetime);
37    }
38  }
```

Zwei Intents in einer Activity & URI Datenzugriff 4

```
24          </activity>
25⊖         <activity android:label="Date/time info" android:name="Info">
26⊖             <intent-filter>
27                  <action android:name="ru.startandroid.intent.action.showtime"/>
28                  <action android:name="ru.startandroid.intent.action.showdate"/>
29                  <category android:name="android.intent.category.DEFAULT"/>
30              </intent-filter>
31          </activity>
```

QUESTIONS?