

Aufgaben 6

1.

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical" >
        <TextView android:id="@+id/viewfrage"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Frage"/>
        <EditText android:id="@+id/editantwort"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Antwort"/>
        <Button android:id="@+id/buttontest"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:onClick="onClicktest"
            android:text="TEST" />
        <TextView android:id="@+id/viewergebnis"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="ergebnis"/>
        <Button android:id="@+id/buttonweiter"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:onClick="onClickweiter"
            android:text="WEITER" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>

<resources>
    <string name="app_name">Auf1_22_23</string>
    <string-array name="fragen">
        <item> frage1</item>
        <item> frage2</item>
    </string-array>
    <string-array name="antworten">
        <item> antwort1</item>
        <item> antwort2</item>
    </string-array>
</resources>
```

```

public class MainActivity extends AppCompatActivity {
    String[] fragenArray;
    String[] antwortenArray;
    TextView fragenview=null;
    EditText antwortenedit=null;
    TextView ergebnisview=null;
    int number=0;
    Random random=null;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        fragenview=(TextView)findViewById(R.id.viewfrage);
        antwortenedit=(EditText) findViewById(R.id.editantwort);
        ergebnisview=(TextView)findViewById(R.id.viewergebnis);
        fragenArray=getResources().getStringArray(R.array.fragen);
        antwortenArray=getResources().getStringArray(R.array.antworten);
        number=0;
        random=new Random();
        fragenview.setText(fragenArray[number]);
    }
    public void onClicktest(View view){
        if
(antwortenedit.getText().toString().equals(antwortenArray[number]))
            ergebnisview.setText("richtig");
        else
            ergebnisview.setText("falsch");
    }
    public void onClickweiter(View view){
        number=random.nextInt(fragenArray.length);
        fragenview.setText(fragenArray[number]);
        antwortenedit.setText(" ");
        ergebnisview.setText("Ergebnis");
    }
}

```

2.

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <EditText android:id="@+id/editdatum"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="yyyy-mm-dd"/>
    <Button android:id="@+id/buttonlos"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="onClick"
        android:text="LOS" />
</LinearLayout>

```

```

public class MainActivity extends AppCompatActivity {
    private final String KEY="datum";
    EditText editdatum=null;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        editdatum=(EditText)findViewById(R.id.editdatum);
    }

    public void onClick(View view) {
        String tosend=editdatum.getText().toString();
        Intent intent=new
Intent(getApplicationContext(),MainActivity2.class);
        intent.putExtra(KEY,tosend);
        startActivity(intent);
    }
}

```

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <TextView android:id="@+id/wochentagview"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="wochentag"/>
</LinearLayout>

```

```

public class MainActivity2 extends AppCompatActivity {
    private final String KEY="datum";
    TextView viewwochentag=null;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);
        viewwochentag=(TextView) findViewById(R.id.wochentagview);
        Intent intent=getIntent();
        String s=intent.getStringExtra(KEY);
        String wochentag="nicht zu ermitteln";
        try {
            LocalDate date = LocalDate.parse(s);
            wochentag = date.getDayOfWeek().toString();
        }
        catch(Exception e){
        }
        viewwochentag.setText(wochentag);
    }
}

```

3.

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <TextView android:id="@+id/beschriftung"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Ich achte auf den Akkustand, aber nur während der
Laufzeit" />
</LinearLayout>
```

```
public class MainActivity extends AppCompatActivity {
    BroadcastReceiver receiver = null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    @Override
    protected void onResume() {
        super.onResume();
        receiver = new MyReceiver();
        IntentFilter filter = new IntentFilter(Intent.ACTION_BATTERY_LOW);
        registerReceiver(receiver, filter);
    }

    @Override
    protected void onPause() {
        super.onPause();
        if (receiver != null) {
            unregisterReceiver(receiver);
            receiver = null;
        }
    }
}
```

```
public class MyReceiver extends BroadcastReceiver {

    @Override
    public void onReceive(Context context, Intent intent) {
        if (Intent.ACTION_BATTERY_LOW.equals(intent.getAction())) {
            Log.i("Receiver", "Der Akku ist bald alle!");
            Toast.makeText(context, "Achtung! Der Akku ist bald alle!",
Toast.LENGTH_LONG).show();
        }
    }
}
```

4.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-permission
        android:name="android.permission.INTERNET"/>

    <application
...
    </application>

</manifest>
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    tools:layout_editor_absoluteX="1dp">
    <EditText
        android:id="@+id/editArt"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Haussperling" />
    <EditText
        android:id="@+id/editOrt"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Freiberg" />
    <EditText
        android:id="@+id/editAnzahl"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="0" />
    <Button
        android:id="@+id/buttonStart"
        android:onClick="onStartClick"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="START" />
</LinearLayout>
```

```
public class MainActivity extends AppCompatActivity {
    public static final String KEY_DATA = "data";
    EditText type = null;
    EditText location = null;
    EditText count = null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

```

        type = (EditText) this.findViewById(R.id.editArt);
        location = (EditText) this.findViewById(R.id.editOrt);
        count = (EditText) this.findViewById(R.id.editAnzahl);
    }

    public void onStartClick(View view) {
        String data = type.getText().toString() + " " +
location.getText().toString() + " " + count.getText().toString();
        Data myData = new Data.Builder()
            .putString(KEY_DATA, data)
            .build();
        WorkRequest workRequest =
            new OneTimeWorkRequest.Builder(MyWorker.class)
                .setInputData(myData)
                .build();
        WorkManager
            .getInstance(getApplicationContext())
            .enqueue(workRequest);
    }
}

public class MyWorker extends Worker {
    public static final String KEY_DATA = "data";

    public MyWorker(@NonNull Context context, @NonNull WorkerParameters
workerParams) {
        super(context, workerParams);
    }
    @NonNull
    @Override
    public Result doWork() {
        String s = getInputData().getString(KEY_DATA);
        Log.i("Worker", s);
        try {
            DatagramSocket socket = new DatagramSocket();
            InetAddress adr = InetAddress.getByName("10.0.2.2");
            byte[] inhalt = s.getBytes();
            DatagramPacket packet = new DatagramPacket
                (inhalt, inhalt.length, adr, 5555);
            socket.send(packet);
            socket.close();
            return Result.success();
        } catch (UnknownHostException e) {
            return Result.failure();
        } catch (SocketException e) {
            return Result.failure();
        } catch (IOException e) {
            return Result.failure();
        }
    }
}
}

```

5.

```
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <Button android:id="@+id/buttonStart"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="onClickStart"
        android:text="START" />
    <Button android:id="@+id/buttonStop"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="onClickStop"
        android:text="STOP" />
</LinearLayout>
```

```
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    //PendingIntent.FLAG_IMMUTABLE or PendingIntent.FLAG_UPDATE_CURRENT
    // setting the mutability flag
    public void onClickStart(View view) {
        Intent intent=new Intent(getApplicationContext(), MyService.class);
        PendingIntent
        pendingIntent=PendingIntent.getService(getApplicationContext(),0,intent,
        PendingIntent.FLAG_IMMUTABLE);
        //SimpleDateFormat sdf = new SimpleDateFormat("dd-M-yyyy
        hh:mm:ss");
        //String dateString = "01-10-2021 10:05:00";
        //Date date = sdf.parse(dateString);
        //Calendar calendar=Calendar.getInstance();
        //calendar.setTime(date);
        //Long firstStart=calendar.getTimeInMillis();
        long firstStart=System.currentTimeMillis();
        long interval=60000;
        //Long interval=24*60*60*1000;
        AlarmManager alarmManager= (AlarmManager)
        getSystemService(Context.ALARM_SERVICE);
        alarmManager.setRepeating(AlarmManager.RTC_WAKEUP,firstStart,
        interval,pendingIntent);
    }

    public void onClickStop(View view) {
        AlarmManager alarmManager= (AlarmManager)
        getSystemService(Context.ALARM_SERVICE);
        Intent intent=new Intent(getApplicationContext(), MyService.class);

        alarmManager.cancel(PendingIntent.getService(getApplicationContext(),0,inte
```

```

nt, PendingIntent.FLAG_IMMUTABLE));
    }
}

```

```

public class MyService extends Service {
    public static final String CHANNEL_ID="123456789";
    public MyService() {
    }

    @Override
    public IBinder onBind(Intent intent) {
        // TODO: Return the communication channel to the service.
        throw new UnsupportedOperationException("Not yet implemented");
    }

    @Override
    public int onStartCommand(Intent intent, int flags, int startId) {
        //return super.onStartCommand(intent, flags, startId);
        Random random=new Random();
        String zahlen="";
        for (int i=0;i<6;i++){
            zahlen=zahlen+" "+String.valueOf(random.nextInt(49)+1);
        }
        zahlen=zahlen+" Superzahl: "+String.valueOf(random.nextInt(10));
        NotificationChannel channel=new NotificationChannel(CHANNEL_ID,
        "Lottoinformationen", NotificationManager.IMPORTANCE_DEFAULT);
        channel.setDescription("eigentlich nur Lottozahlen");
        NotificationManager notificationManager =
        getApplicationContext().getSystemService(NotificationManager.class);
        notificationManager.createNotificationChannel(channel);
        NotificationCompat.Builder builder = new
        NotificationCompat.Builder(this, CHANNEL_ID)
            .setSmallIcon(R.drawable.ic_stat_name)
            .setContentTitle("Deine Lottozahlen")
            .setContentText(zahlen)
            .setAutoCancel(true)
            .setPriority(NotificationCompat.PRIORITY_DEFAULT);

        //NotificationManagerCompat notificationManager =
        NotificationManagerCompat.from(this);
        int notificationId=9999;
        // notificationId is a unique int for each notification that you
        must define
        notificationManager.notify(notificationId, builder.build());
        this.stopSelf();
        return START_STICKY;
    }
}

```