Pokračujeme tedy dál na \*\*Android aplikace\*\*. Tohle je velmi důležitá část, pokud máš Android jako možnost pro maturitní úkol.

### \*\*Android Aplikace – Základy a Struktura\*\*

1. \*\*Co je Android?\*\*

Android je operační systém pro mobilní zařízení, který běží na jazyce \*\*Java (případně Kotlin)\*\*. Kód aplikace běží na \*\*JVM\*\* (Java Virtual Machine). Tvorba Android aplikací zahrnuje několik hlavních komponent, mezi které patří:

- \*\*Activities\*\* – představují jednotlivé obrazovky v aplikaci.

- \*\*Fragments\*\* – poskytují flexibilitu a umožňují rozdělit obsah na menší části.

- \*\*Services\*\* – běží na pozadí a umožňují dlouhotrvající úkoly.

- \*\*Broadcast Receivers\*\* – reagují na systémové zprávy.

- \*\*Content Providers\*\* – umožňují sdílení dat mezi aplikacemi.

---

### \*\*2. Struktura projektu v Android Studiu\*\*

Každý Android projekt má specifickou strukturu, která zahrnuje:

- \*\*Manifests\*\* – soubor `AndroidManifest.xml` definuje informace o aplikaci, např. hlavní aktivitu.

- \*\*res\*\* – složka pro všechny zdroje, jako jsou obrázky, texty, a layouty.

- \*\*java\*\* – zde je umístěn kód aplikace.

```plaintext

app/

└── src/

└── main/

├── java/

│ └── com.example.mujaplikace/

│ └── MainActivity.java

├── res/

│ ├── layout/

│ │ └── activity\_main.xml

│ ├── values/

│ │ └── strings.xml

├── AndroidManifest.xml

```

- \*\*activity\_main.xml\*\* – obsahuje design pro hlavní obrazovku.

- \*\*strings.xml\*\* – ukládá textové řetězce pro více jazyků.

---

### \*\*3. Vytvoření jednoduché aplikace\*\*

1. \*\*Hlavní aktivita (MainActivity)\*\*

Aplikace má obvykle hlavní aktivitu, která je spuštěna při startu aplikace. Příklad hlavní aktivity:

```java

package com.example.mujaplikace;

import android.os.Bundle;

import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main); // Nastavení layoutu

TextView textView = findViewById(R.id.textView); // Získání textového pole

textView.setText("Ahoj světe!"); // Nastavení textu

}

}

```

2. \*\*Layout pro hlavní aktivitu\*\*

V souboru `activity\_main.xml` definujeme vzhled obrazovky:

```xml

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<TextView

android:id="@+id/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Toto je text!"

android:textSize="20sp"

android:layout\_centerInParent="true"/>

</RelativeLayout>

```

Tento layout obsahuje jednoduchý textový prvek, který je vycentrován na obrazovce.

---

### \*\*4. Přechod mezi obrazovkami (Intent)\*\*

Pokud máš víc aktivit, můžeš mezi nimi přecházet pomocí \*\*Intent\*\*. Příklad přechodu mezi dvěma aktivitami:

1. \*\*První aktivita\*\* (MainActivity):

```java

Intent intent = new Intent(MainActivity.this, DruhaAktivita.class);

startActivity(intent);

```

2. \*\*Druhá aktivita\*\* (DruhaAktivita.java):

```java

package com.example.mujaplikace;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

public class DruhaAktivita extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_druha);

}

}

```

3. \*\*Layout pro druhou aktivitu\*\* (`activity\_druha.xml`):

```xml

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Druhá aktivita"

android:textSize="20sp"

android:layout\_centerInParent="true"/>

</RelativeLayout>

```

Tímto způsobem uživatel přejde z jedné obrazovky na druhou.

---

### \*\*5. Ukládání dat pomocí SharedPreferences\*\*

Pro jednoduché ukládání dat, jako jsou uživatelské preference, můžeme využít `SharedPreferences`.

- \*\*Uložení hodnoty\*\*:

```java

SharedPreferences sharedPreferences = getSharedPreferences("mojePrefs", MODE\_PRIVATE);

SharedPreferences.Editor editor = sharedPreferences.edit();

editor.putString("username", "Petr");

editor.apply();

```

- \*\*Načtení hodnoty\*\*:

```java

SharedPreferences sharedPreferences = getSharedPreferences("mojePrefs", MODE\_PRIVATE);

String username = sharedPreferences.getString("username", "Neznámý");

```

---

### \*\*6. Práce s databází SQLite\*\*

SQLite je lehká databáze, kterou můžeš využít k ukládání dat do souboru v zařízení.

- \*\*Vytvoření databáze a tabulky\*\*:

```java

public class DBHelper extends SQLiteOpenHelper {

public DBHelper(Context context) {

super(context, "MojeDatabaze", null, 1);

}

@Override

public void onCreate(SQLiteDatabase db) {

db.execSQL("CREATE TABLE uzivatele (id INTEGER PRIMARY KEY, jmeno TEXT, vek INTEGER)");

}

@Override

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

db.execSQL("DROP TABLE IF EXISTS uzivatele");

onCreate(db);

}

}

```

- \*\*Uložení dat\*\*:

```java

DBHelper dbHelper = new DBHelper(this);

SQLiteDatabase db = dbHelper.getWritableDatabase();

ContentValues values = new ContentValues();

values.put("jmeno", "Petr");

values.put("vek", 25);

db.insert("uzivatele", null, values);

```

- \*\*Načtení dat\*\*:

```java

SQLiteDatabase db = dbHelper.getReadableDatabase();

Cursor cursor = db.query("uzivatele", null, null, null, null, null, null);

if (cursor.moveToFirst()) {

String jmeno = cursor.getString(cursor.getColumnIndex("jmeno"));

int vek = cursor.getInt(cursor.getColumnIndex("vek"));

cursor.close();

}

```

---

### \*\*7. Správa životního cyklu aktivit\*\*

V Androidu je velmi důležité porozumět životnímu cyklu aktivit. Každá aktivita prochází několika stavy:

1. \*\*onCreate()\*\* – aktivita je vytvořena.

2. \*\*onStart()\*\* – aktivita je viditelná.

3. \*\*onResume()\*\* – aktivita je na předním plánu.

4. \*\*onPause()\*\* – aktivita je na pozadí.

5. \*\*onStop()\*\* – aktivita není viditelná.

6. \*\*onDestroy()\*\* – aktivita je zničena.

Pokud potřebuješ uchovávat data mezi těmito stavy, můžeš využít \*\*onSaveInstanceState()\*\*.

---

### \*\*8. Zpracování vstupu (Button, EditText, atd.)\*\*

Android umožňuje interaktivní vstupy pomocí prvků, jako jsou tlačítka (`Button`), textová pole (`EditText`), atd.

```java

Button button = findViewById(R.id.button);

button.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

EditText editText = findViewById(R.id.editText);

String text = editText.getText().toString();

Toast.makeText(MainActivity.this, "Zadaný text: " + text, Toast.LENGTH\_SHORT).show();

}

});

```

---

Toto je základní přehled pro Android aplikace. Pokud potřebuješ ještě něco konkrétnějšího nebo vysvětlení nějaké specifické části, dej mi vědět!