

Meet the Teacher

sqlite

OR-Mapper derzeit in Entwicklung

JB team TC build success Download 0.10.2 license Apache License 2.0

Exposed - Kotlin SQL Library

Exposed is a prototype for a lightweight SQL library written over JDBC driver for Kotlin language. It does have two layers of database access: typesafe SQL wrapping DSL and lightweight data access objects

Exposed is currently available for maven/gradle builds at <https://bintray.com/kotlin/exposed/exposed/view#>

Dialects

Currently supported database dialects:

- PostgreSQL
- MySQL
- Oracle
- SQLite
- H2
- SQL Server

<https://github.com/JetBrains/Exposed>

Wird aber nicht verwendet



AddTeachers in DB implementieren

- Geladene Lehrerdaten in Datenbank speichern
- Nur eine Tabelle
- Textspalten und eine BLOB-Spalte

```
ABERGER Christian  
ATAK Mehmet  
BODENSTORFER Bernhard  
BRANDSTETTER Stefanie  
HANNESSCHLÄGER Jürgen  
HAUNSCHMID Wilfried  
KLEWEIN David  
NIEDERMAIER Maren  
SOKOLI Anila  
STRECKER Alexander  
STRUBER Ingrid  
WINTER Lukas  
AISTLEITNER Gerald  
AITENBICHLER Herbert  
APOLLONIO Eva-Maria  
AUBERGER Elisabeth  
AUER Peter  
AUERNIG Franz
```



Create Android Project

Application name**Company domain****Project location** ...**Package name**

- Include C++ support
- Include Kotlin support



Target Android Devices

Select the form factors and minimum SDK

Some devices require additional SDKs. Low API levels target more devices, but offer fewer API features.

Phone and Tablet

API 19: Android 4.4 (KitKat)

By targeting **API 19 and later**, your app will run on approximately **90.1%** of devices. [Help me choose](#)

Include Android Instant App support

Wear OS

API 23: Android 6.0 (Marshmallow)

TV

API 21: Android 5.0 (Lollipop)

Android Auto

Android Things

API 24: Android 7.0 (Nougat)

Cancel

Previous

Next

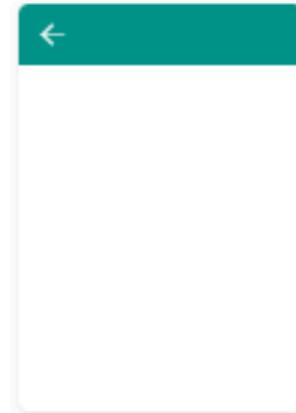
Finish



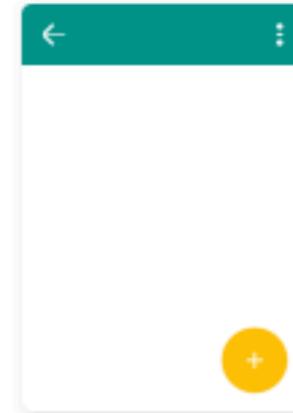
Add an Activity to Mobile



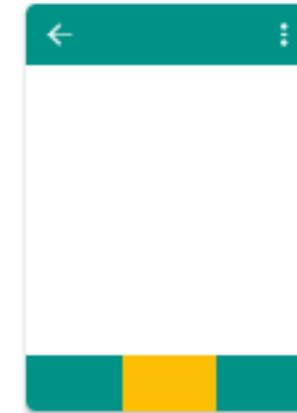
Add No Activity



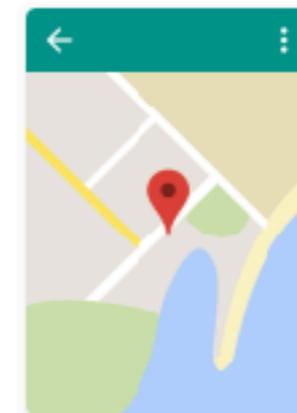
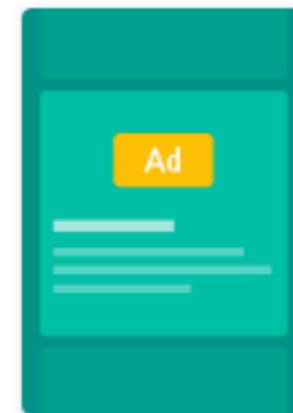
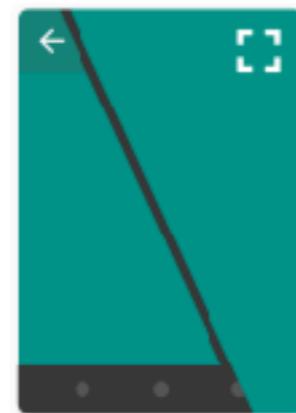
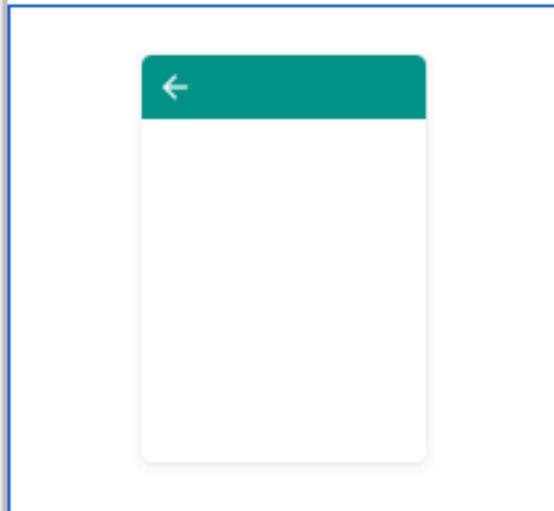
Activity & Fragment+ViewModel



Basic Activity



Bottom Navigation Activity



Cancel

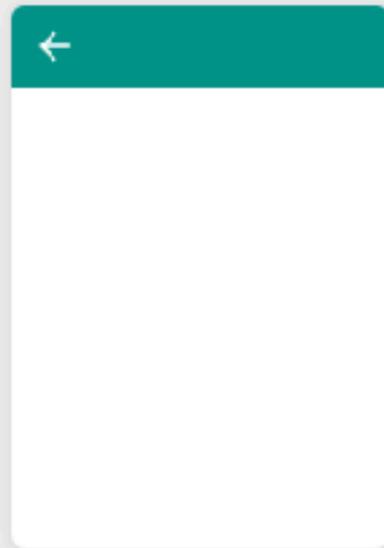
Previous

Next

Finish



Creates a new empty activity



Activity Name:

Generate Layout File

Layout Name:

Backwards Compatibility (AppCompat)

The name of the activity class to create

Cancel

Previous

Next

Finish

Dependencies

- Gradle Scripts
 - build.gradle (Project: MeetTheTeacher)
 - build.gradle (Module: app)**
 - gradle.properties
 - gradle-wrapper.properties
 - local.properties
 - proguard-rules.pro (ProGuard Rules for)
 - settings.gradle (Project Settings)

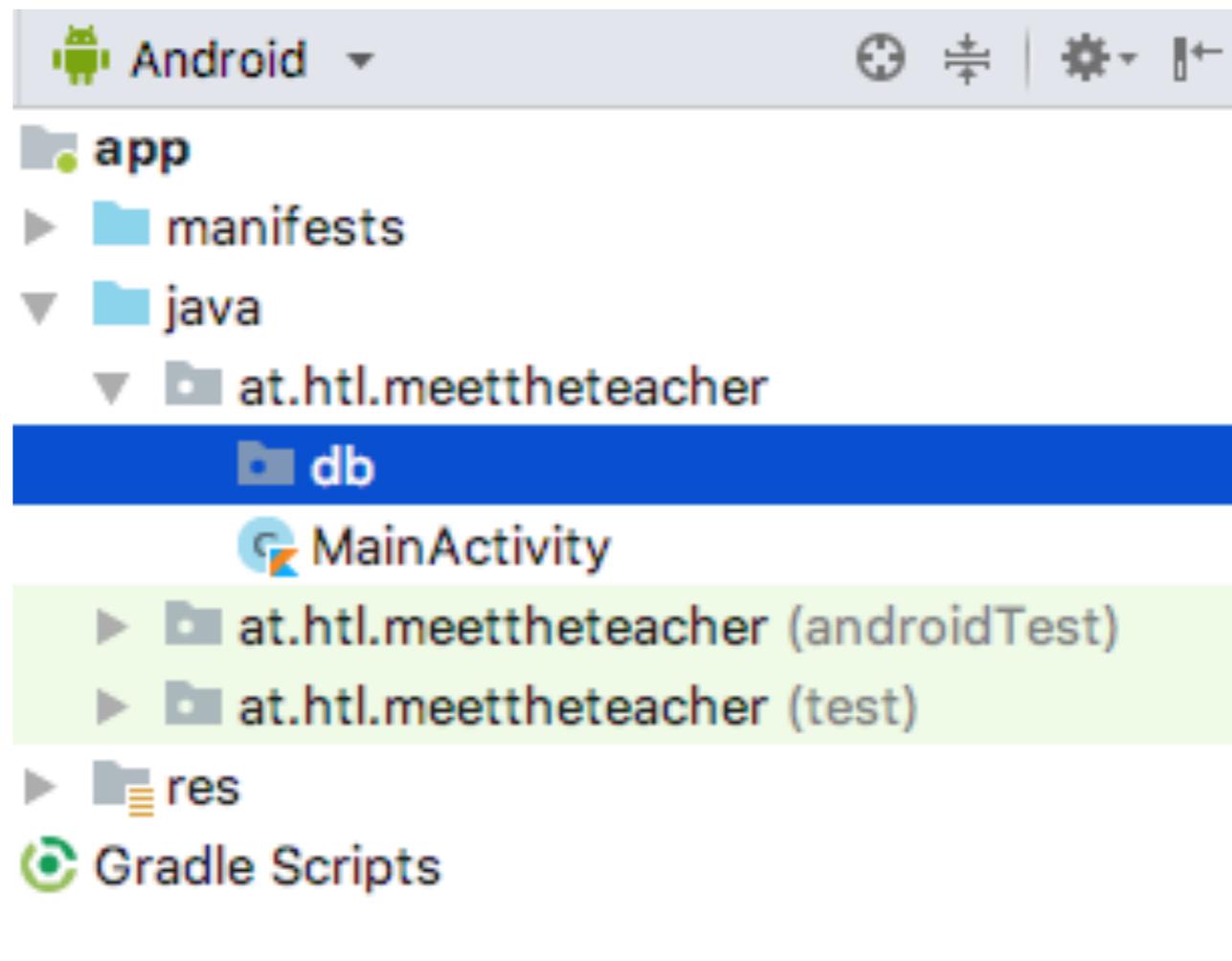
```
dependencies {  
    // implementation 'org.jetbrains.exposed:exposed:0.10.2'  
    implementation fileTree(include: ['*.jar'], dir: 'libs')  
    implementation "org.jetbrains.kotlin:kotlin-stdlib-jdk7:$kotlin_version"  
    implementation 'com.android.support:appcompat-v7:27.1.1'  
    implementation 'com.android.support.constraint:constraint-layout:1.1.0'  
    testImplementation 'junit:junit:4.12'  
    androidTestImplementation 'com.android.support.test:runner:1.0.2'  
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'  
    implementation 'com.android.support:cardview-v7:27.1.1'  
    implementation 'com.android.support:recyclerview-v7:27.1.1'  
    implementation 'org.jsoup:jsoup:1.11.3'  
}
```

manifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="at.htl.meettheteacher">
    <uses-permission android:name="android.permission.INTERNET" />
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

Eigenes Package für DB-Funktionalität



Contracts definieren

```
package at.htl.meettheteacher.db

import android.provider.BaseColumns

val DATABASE_NAME = "meettheteacher.db"
val DATABASE_VERSION = 10 // Version 1.0
```

```
object TeacherEntry : BaseColumns {
    val TABLE_NAME = "teacher"
    val _ID = "id"
    val NAME_COL = "name"
    val DAY_COL = "day"
    val TIME_COL = "time"
    val ROOM_COL = "room"
    val BITMAP_COL = "bitmap"
}
```

 BaseColumns

```
package android.provider;
```

```
public interface BaseColumns {
    String _COUNT = "_count";
    String _ID = "_id";
}
```

Kotlin - object-Declaration

Object declarations

[Singleton](#) may be useful in several cases, and Kotlin (after Scala) makes it easy to declare singletons:

```
object DataManager {  
    fun registerDataProvider(provider: DataProvider) {  
        // ...  
    }  
  
    val allDataProviders: Collection<DataProvider>  
        get() = // ...  
}
```

<https://kotlinlang.org/docs/reference/object-declarations.html>

Teacher.kt

```
data class Teacher (val name: String,  
                    val day: String,  
                    val time: String,  
                    val room : String,  
                    val detailsId : Int?,  
                    var bitmap: Bitmap?)
```

Datenbank verwalten

TeacherDbHelper

```
class TeacherDbHelper(context: Context) : SQLiteOpenHelper(context, DATABASE_NAME, null, DATABASE_VERSION) {  
  
    private val SQL_CREATE_ENTRIES = "CREATE TABLE ${TeacherEntry.TABLE_NAME} (" +  
        "${TeacherEntry._ID} INTEGER PRIMARY KEY," +  
        "${TeacherEntry.NAME_COL} TEXT," +  
        "${TeacherEntry.DAY_COL} TEXT," +  
        "${TeacherEntry.TIME_COL} TEXT," +  
        "${TeacherEntry.ROOM_COL} TEXT," +  
        "${TeacherEntry.BITMAP_COL} BLOB" +  
        ")"  
  
    private val SQL_DELETE_ENTRIES = "DROP TABLE IF EXISTS ${TeacherEntry.TABLE_NAME}"  
  
    override fun onCreate(db: SQLiteDatabase) {  
        db.execSQL(SQL_CREATE_ENTRIES)  
    }  
  
    override fun onUpgrade(db: SQLiteDatabase, oldVersion: Int, newVersion: Int) {  
        db.execSQL(SQL_DELETE_ENTRIES)  
        onCreate(db)  
    }  
}
```

Add teacher - Java-Way

TeacherDbTable.kt

```
fun add(teacher: Teacher): Long {  
    val db = dbHelper.writableDatabase  
  
    val values = ContentValues()  
    values.put(TeacherEntry.NAME_COL, teacher.name)  
    values.put(TeacherEntry.DAY_COL, teacher.day)  
    values.put(TeacherEntry.TIME_COL, teacher.time)  
    values.put(TeacherEntry.ROOM_COL, teacher.room)  
    values.put(TeacherEntry.BITMAP_COL, toByteArray(teacher.bitmap))  
    db.beginTransaction()  
    val result = try {  
        val returnValue = db.insert(TeacherEntry.TABLE_NAME, null, values)  
        db.setTransactionSuccessful()  
        returnValue  
    } finally {  
        db.endTransaction()  
    }  
    db.close()  
    Log.d(LOG_TAG, "Stored new teacher to the DB $teacher")  
    return result  
}
```

Bitmap —> ByteArray —> BLOB —> DB

- Bitmap ist nullable
- Null-Safe-Operator
- <https://kotlinlang.org/docs/reference/null-safety.html>

```
private fun toByteArray(bitmap: Bitmap?): ByteArray {  
    val stream = ByteArrayOutputStream()  
    bitmap?.compress(Bitmap.CompressFormat.PNG, 0, stream)  
    return stream.toByteArray()  
}
```

Verwendung von addTeacher() in Mainactivity

- Auslösen über Menü
 - Siehe PayYours
 - Brauchbares Icon suchen
- View zwischen RecyclerView und Import-View wechseln
 - Visibility
 - Fragments
- Aktueller Lehrer im Download
 - Text oder CardView wiederverwenden
- Progressbar dokumentiert Fortschritt
 - Nur für Bilder
 - Text ist innerhalb einer Sekunde geladen

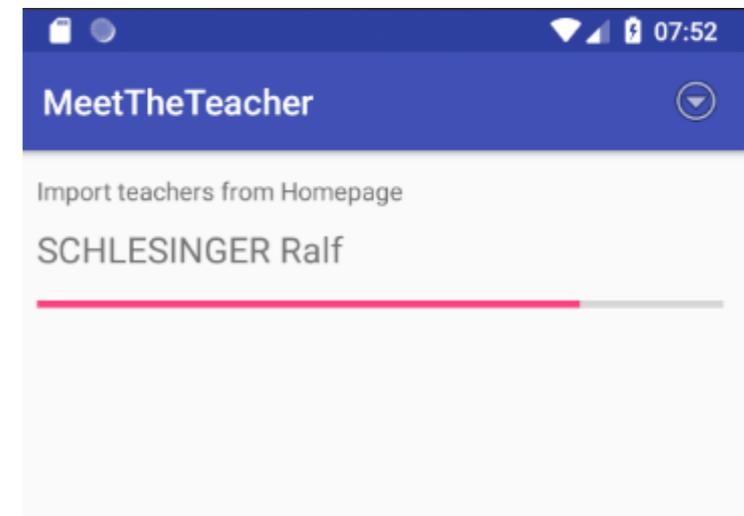


Bild trotz ID nicht verfügbar

—> Anonym

MeetTheTeacher		
	HOLZMANN Michael	Mittwoch 10:55 - 11:45 E08
	HUBER Gerhard	Dienstag 08:00 - 08:50 U9 1
	HUEMER Martin	Mittwoch 10:55 - 11:45 148
	HÖFER Gerhard	Dienstag 17:20 - 18:05 254

Menüpunkt ausgewählt

MainActivity.kt

```
private fun loadTeachersFromHomepage(){
    var imagesToLoad = 0
    ll_import.visibility=VISIBLE
    rv_teachers.visibility= INVISIBLE
    TeacherDbTable(this).deleteAllTeachers()
    val loadTeachersAsyncTask = LoadTeachersAsyncTask {
        //rv_teachers.adapter=TeachersAdapter(it)
        imagesToLoad = it.size
        pb_progress.max = imagesToLoad
        it.forEach {
            if (it.detailsId == 0) {
                it.bitmap = BitmapFactory.decodeResource(resources, R.drawable.anonym)
                imagesToLoad--
                pb_progress.progress++
                TeacherDbTable(this).add(it)
            } else {
                val loadTeachersImageAsyncTask = LoadTeachersImageAsyncTask {
                    tv_actual_teacher.text = it.name
                    TeacherDbTable(this).add(it)
                    imagesToLoad--
                    pb_progress.progress++
                    if (imagesToLoad == 0) { // alle Bilder geladen
                        ll_import.visibility = INVISIBLE
                        rv_teachers.visibility = VISIBLE
                        setTeachersAdapterToDb()
                    }
                }
                try {
                    loadTeachersImageAsyncTask.execute(it)
                } catch (e: Exception) {
                    e.printStackTrace()
                    Log.e(LOG_TAG, "getImage() load image, exception: " + e.message)
                }
            }
        }
    }
}
try {
    loadTeachersAsyncTask.execute(url)
    Log.e(LOG_TAG, "asynctask started")
} catch (e: Exception) {
    e.printStackTrace()
    Log.e(LOG_TAG, "Loading teachers failed: " + e.message)
}
```

RecyclerView-Adapter versorgen

- TeacherDbTable um readAllTeachers() erweitern

 MainActivity.kt

```
private fun setTeachersAdapterToDb(){  
    val teachers = TeacherDbTable(this).readAllTeachers()  
    rv_teachers.adapter = TeachersAdapter(teachers)  
}
```

Kontrolle der Daten in SQLITE-DB

- Datenbankdatei aus Anwendungsverzeichnis auf PC kopieren
- Mit Commandline-Tools inspizieren
- Wechsel in Tools-Verzeichnis von Android

Administrator: Windows PowerShell

```
PS C:\Users\gera1\AppData\Local\Android\Sdk\platform-tools>
```

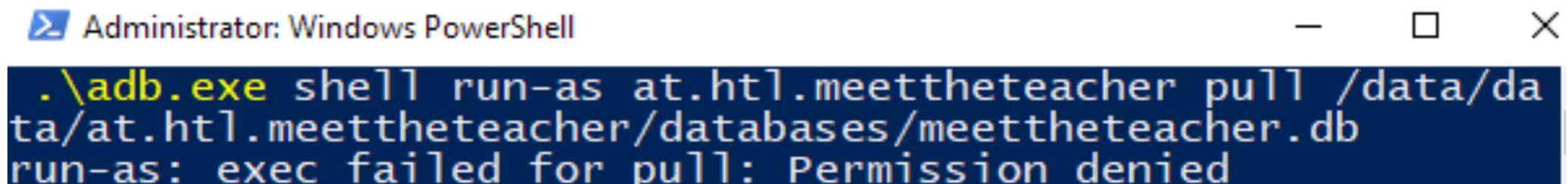
```
cd /opt/android-sdk-macosx/platform-tools/
```

Datenbankdatei herunterladen

- Zugriff mit Credentials der App
- Datenbank zuerst auf SD-Karte kopieren
 - `.\adb.exe shell run-as at.htl.meettheteacher cp /data/data/at.htl.meettheteacher/databases/meettheteacher.db /sdcard/`
- Daten von SD-Karte auf PC übertragen
 - `.\adb.exe pull /sdcard/meettheteacher.db`

```
.\adb.exe pull /sdcard/meettheteacher.db  
/sdcard/meettheteacher.db: 1.../s (3223552 bytes in 0.049s)
```

- Direktes pull aus Appverzeichnis ist nicht erlaubt



The screenshot shows a Windows PowerShell window titled "Administrator: Windows PowerShell". The command entered is `.\adb.exe shell run-as at.htl.meettheteacher pull /data/data/at.htl.meettheteacher/databases/meettheteacher.db`. The output is `run-as: exec failed for pull: Permission denied`.

```
Administrator: Windows PowerShell  
.\adb.exe shell run-as at.htl.meettheteacher pull /data/data/at.htl.meettheteacher/databases/meettheteacher.db  
run-as: exec failed for pull: Permission denied
```

Datenbankdatei analysieren

```
.\sqlite3.exe .\meettheteacher.db  
SQLite version 3.19.4 2017-08-18 19:28:12  
Enter ".help" for usage hints.  
sqlite> select name from teacher;
```

```
ABERGER Christian  
ATAK Mehmet  
BODENSTORFER Bernhard  
BRANDSTETTER Stefanie  
HANNESSCHLÄGER Jürgen  
HAUNSCHMID Wilfried  
KLEWEIN David  
NIEDERMAIER Maren  
SOKOLI Anila  
STRECKER Alexander  
STRUBER Ingrid  
WINTER Lukas  
AISTLEITNER Gerald  
AITENBICHLER Herbert  
APOLLONIO Eva-Maria  
AUBERGER Elisabeth  
AUER Peter  
AUERNIG Franz
```

Aufgaben

- TeacherDbTable erweitern
 - readAllTeachers() liest alle Lehrer sortiert nach Namen aus der Datenbank
- Hinweis: db.query() liefert Cursor zurück
 - deleteAllTeacher() löscht alle Lehrer aus der Datenbank
- Aufruf vor Start des Imports
- Soweit möglich, Kotlin-Features einsetzen
 - Extension-Functions, spezielle Functions (with, ...)

