



MM1 & MM3

Tahun Ajaran 2022/2023

PEMROGRAMAN

PERANGKAT BERGERAK

(KOTLIN)

Dosen : Muhammad Lulu Latif Usman,
S.Pd., M.Han

Asprak : Bunga Laelatul Muna

MODUL 7

LOGIN WITH SQLITE SHARED PREFERENCES

(Link Video: <https://youtu.be/2wIOfzd5NGQ>)

(Link Github: [bungagana/LoginWithSharedPreference \(github.com\)](https://github.com/bungagana/LoginWithSharedPreference))

SharedPreferences adalah mekanisme penyimpanan data kecil dalam bentuk key-value pairs pada aplikasi Android.

Data ini dapat disimpan pada perangkat dalam jangka waktu yang lama dan dapat diakses oleh berbagai komponen dalam aplikasi. SharedPreferences sangat berguna dalam menyimpan preferensi pengguna, seperti pengaturan aplikasi, riwayat login, bahasa pengguna, atau tema yang dipilih.

Code to Access Shared Preference

```
val savedLogin = getSharedPreferences( name: "Login", MODE_PRIVATE)
```

Code to Edit Shared Preference

```
val editSavedLogin = savedLogin.edit()
```

Code to Remove and Clear Shared Preference

```
editSavedLogin.remove( key: "username")  
editSavedLogin.apply()
```

```
editSavedLogin.clear()  
editSavedLogin.apply()
```

PRAKTIKUM

Style

```
<style name="EditTextLogin">
    <item name="android:layout_height">50dp</item>
    <item name="android:layout_width">250dp</item>
    <item name="android:drawablePadding">10dp</item>
    <item name="android:textColor">@color/black</item>
</style>
<style name="BtnLogin">
    <item name="android:layout_width">220dp</item>
    <item name="android:layout_height">wrap_content</item>
    <item name="android:textColor">@color/white</item>
    <item name="android:layout_margin">10dp</item>
    <item name="backgroundTint">@color/Navy</item>
</style>
```

Color

```
<color name="Navy">#263859</color>
<color name="btnpress">#004A85</color>
```

Buat 3 Empty Activity :

1. Splash Activity
2. Login Activity
3. Register Activity

Activity_splash.xml (TAMPILAN LAYOUT SPLASH SCREEN)

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    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=".SplashActivity"
    android:orientation="vertical">
```

```

        android:gravity="center">
        <ImageView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:src="@drawable/logo"/>
        <ProgressBar
            style="@style/Widget.AppCompat.ProgressBar.Horizontal"
            android:layout_width="200dp"
            android:layout_height="wrap_content"
            android:indeterminate="true"
            android:indeterminateTint="@color/btnpress"/>
    </LinearLayout>

```

Output



Login Activit.xml (TAMPILAN LAYOUT LOGIN)

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    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=".LoginActivity"
    android:orientation="vertical"
    android:gravity="center"
    android:layout_marginBottom="20dp">
    <ImageView
        android:layout_width="150dp"
        android:layout_height="150dp"

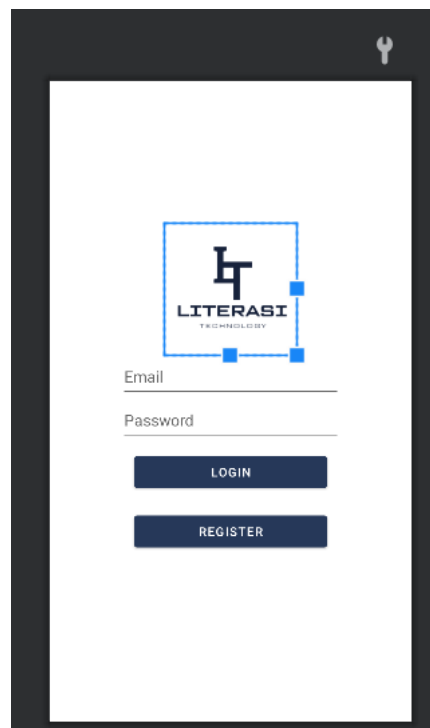
```

```

        android:src="@drawable/logo"/>
<EditText
    style="@style/EditTextLogin"
    android:id="@+id/EditUsernameLogin"
    android:hint="Email"
    android:inputType="textEmailAddress"/>
<EditText
    style="@style/EditTextLogin"
    android:id="@+id/EditPasswordLogin"
    android:hint="Password"
    android:inputType="textPassword"/>
<Button
    android:id="@+id/btnLogin"
    style="@style/BtnLogin"
    android:text="Login"
    android:backgroundTint="@color/Navy"/>
<Button
    android:id="@+id/btnregister"
    style="@style/BtnLogin"
    android:text="Register"
    android:backgroundTint="@color/Navy"/>
</LinearLayout>

```

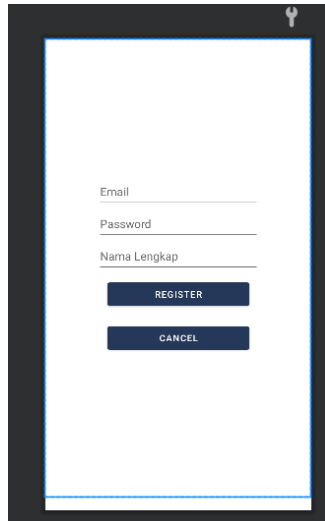
Output



Activity_register.xml
(TAMPILAN LAYOUT REGISTER)

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
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=".RegisterActivity"
android:orientation="vertical"
android:gravity="center"
android:layout_marginBottom="20dp">
    <EditText
        style="@style/EditTextLogin"
        android:id="@+id/EditEmailRegister"
        android:hint="Email"
        android:inputType="textEmailAddress"/>
    <EditText
        style="@style/EditTextLogin"
        android:id="@+id/EditPasswordRegister"
        android:hint="Password"
        android:inputType="textPassword"/>
    <EditText
        style="@style/EditTextLogin"
        android:id="@+id/EditFullnameRegister"
        android:hint="Nama Lengkap"
        android:inputType="text"/>
    <Button
        android:id="@+id/btnsubmitregister"
        style="@style/BtnLogin"
        android:text="Register"
        android:onClick="registerme"/>
    <Button
        android:id="@+id/btncancelregister"
        style="@style/BtnLogin"
        android:text="Cancel"
        android:onClick="cancelme"/>
</LinearLayout>
```

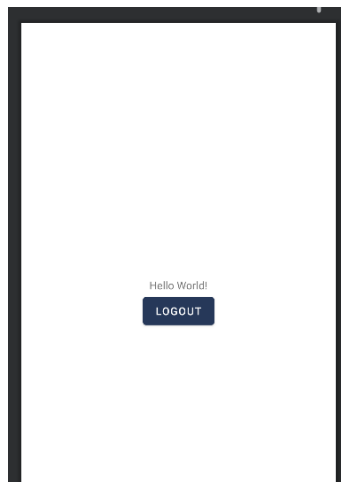
Output



Activity_main.xml (TAMPILAN LAYOUT KETIKA BERHASIL LOGIN)

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    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"
    android:orientation="vertical"
    android:gravity="center">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!" />
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/btnLogout"
        android:backgroundTint="@color/Navy"
        android:textColor="@color/white"
        android:text="Logout"/>
</LinearLayout>
```

Output



Buat Class Untuk Menampung variable yang akan digunakan untuk inputan pengguna

DataModelUser (Kotlin Class)

```
data class DataModelUser (  
    var email    : String = "",  
    var pass     : String = "",  
    var fullname: String = ""  
)
```

DBInfo (Object)

Ini menampung data table, disini membuat table dengan nama user dan memiliki 4 kolom.

```
import android.provider.BaseColumns  
object DBInfo {  
    class UserTable: BaseColumns {  
        companion object {  
            val TABLE_NAME  = "user"  
            val COL_EMAIL    = "email"  
            val COL_PASS     = "pass"  
            val COL_FULLNAME = "fullname"  
            val COL_JUMLAH   = "jumlah"  
        }  
    }  
}
```


DB Helper (Class)

```
class DBHelper(context: Context): SQLiteOpenHelper(context,
    DATABASE_NAME, null, DATABASE_VERSION) {
    companion object{
        val DATABASE_NAME = "myaps.db"
        val DATABASE_VERSION = 1
        private val SQL_CREATE_USER = "CREATE TABLE " +
DBInfo.UserTable.TABLE_NAME + "("+DBInfo.UserTable.COL_EMAIL+"
VARCHAR(200) PRIMARY KEY, " + DBInfo.UserTable.COL_PASS + " TEXT, "
+ DBInfo.UserTable.COL_FULLNAME + " TEXT);"
        private val SQL_DELETE_ENTRIES = "DROP TABLE IF EXISTS " +
DBInfo.UserTable.TABLE_NAME
    }

    override fun onCreate(db: SQLiteDatabase?) {
        db?.execSQL(SQL_CREATE_USER)
    }

    override fun onUpgrade(db: SQLiteDatabase?, oldVersion: Int,
newVersion: Int) {
        onUpgrade(db, oldVersion, newVersion)
    }
    @Throws(SQLiteConstraintException::class)
    fun RegisterUser(emailin: String, passin: String, fullnamein:
String) {
        val db = writableDatabase
        val namatable = DBInfo.UserTable.TABLE_NAME
        val emailt = DBInfo.UserTable.COL_EMAIL
        val passt = DBInfo.UserTable.COL_PASS
        val fullnamet = DBInfo.UserTable.COL_FULLNAME

        var sql = "INSERT INTO " + namatable + "(" + emailt + "," +
+ passt + "," + fullnamet + ") VALUES ('" + emailin + "','" +
passin + "','" + fullnamein + "');"
        db.execSQL(sql)
    }
    @SuppressWarnings("Range")
    fun cekUser(emailin: String): String {
        val db = writableDatabase
        var cursor: Cursor? = null
        var jumlah = ""
        try {
            cursor = db.rawQuery("SELECT COUNT(" +
DBInfo.UserTable.COL_EMAIL + ") as jumlah FROM " +
DBInfo.UserTable.TABLE_NAME + " WHERE " + DBInfo.UserTable.COL_EMAIL
+ "==" + emailin + " ", null)
        } catch (e: android.database.SQLException) {
            db.execSQL(SQL_CREATE_USER)
            return "Tabel Dibuat"
        }
        if (cursor!!.moveToFirst()) {
            jumlah =
cursor.getString(cursor.getColumnIndex(DBInfo.UserTable.COL_JUMLAH))
        }
    }
}
```

```

    }
    return jumlah
}
@SuppressLint("Range")
fun cekLogin(emailin: String, passin: String): String {
    val db = writableDatabase
    var cursor: Cursor? = null
    var jumlah = ""
    try {
        cursor = db.rawQuery("SELECT COUNT(" +
            DBInfo.UserTable.COL_EMAIL + ") as jumlah FROM " +
            DBInfo.UserTable.TABLE_NAME + " WHERE " + DBInfo.UserTable.COL_EMAIL
            + "==" + emailin + "' AND " + DBInfo.UserTable.COL_PASS + "==" +
            passin + "' , null)
    } catch (e: android.database.SQLException) {
        db.execSQL(SQL_CREATE_USER)
        return "Tabel Dibuat"
    }
    if (cursor!!.moveToFirst()) {
        jumlah =
        cursor.getString(cursor.getColumnIndex(DBInfo.UserTable.COL_JUMLAH))
    }
    return jumlah
}
}

```

LoginActivity.kt

```

class LoginActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_login)

        val savedLogin = getSharedPreferences("Login", MODE_PRIVATE)
        val editSavedLogin = savedLogin.edit()

        if (savedLogin.getString("Status", "Off")=="On"){
            startActivity(Intent(this, MainActivity::class.java))
        }

        val editUsername: EditText =
        findViewById(R.id.EditUsernameLogin)
        val editPassword: EditText =
        findViewById(R.id.EditPasswordLogin)
        val btnLogin: Button = findViewById(R.id.btnLogin)
        val btnRegister: Button = findViewById(R.id.btnregister)
        val userDBHelper = DBHelper(this)
        btnLogin.setOnClickListener {
            var emailku = editUsername.text.toString()
            var passku = editPassword.text.toString()
            var cekLogin = userDBHelper.cekLogin(emailku, passku)
            if(cekLogin=="1"){
                editSavedLogin.putString("Email", emailku)
                editSavedLogin.putString("Password", passku)
                editSavedLogin.putString("Status", "On")
            }
        }
    }
}

```

```

        editSavedLogin.commit()

        val intent = Intent(this, MainActivity::class.java)
        startActivity(intent)
    } else {
        val toast: Toast =
Toast.makeText(applicationContext,
        "Gagal Login", Toast.LENGTH_SHORT)
        toast.show()
    }
}
btnRegister.setOnClickListener {
    val intent = Intent(this, RegisterActivity::class.java)
    startActivity(intent)
}

}
}

```

RegisterActivity.kt

```

class RegisterActivity : AppCompatActivity() {
    lateinit var email: EditText
    lateinit var password: EditText
    lateinit var fullname: EditText
    lateinit var btnregister: Button
    lateinit var btncancel: Button
    lateinit var userDBHelper: DBHelper
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_register)

        email = findViewById(R.id.EditEmailRegister)
        password = findViewById(R.id.EditPasswordRegister)
        fullname = findViewById(R.id.EditFullnameRegister)
        btnregister = findViewById(R.id.btnsubmitregister)
        btncancel = findViewById(R.id.btncancelregister)
        userDBHelper = DBHelper(this)
    }
    fun registerme(view: View){
        var iemail = email.text.toString()
        var ipassword = password.text.toString()
        var ifullname = fullname.text.toString()

        var cekuser = userDBHelper.cekUser(iemail)
        var status = "Gagal"
        if (cekuser == "0") {
            userDBHelper.RegisterUser(iemail, ipassword, ifullname)
            status = "Sukses"
            val intent = Intent(this, LoginActivity::class.java)
            startActivity(intent)
        }
        val toast: Toast = Toast.makeText(applicationContext,
            status, Toast.LENGTH_SHORT)
        toast.show()
    }
}

```

```

    fun cancelme(view: View){
        startActivity(Intent(this, LoginActivity::class.java))
    }
}

```

MainActivity.kt

```

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        val btnLogout: Button = findViewById(R.id.btnLogout)
        val savedLogin = getSharedPreferences("Login", MODE_PRIVATE)
        val editSavedLogin = savedLogin.edit()
        btnLogout.setOnClickListener {
            editSavedLogin.putString("Email", null)
            editSavedLogin.putString("Password", null)
            editSavedLogin.putString("Status", "Off")
            editSavedLogin.commit()
            startActivity(Intent(this, LoginActivity::class.java))
        }
    }
}

```

SplashActivity.kt

```

@Suppress("DEPRECATION")
class SplashActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_splash)
        Handler().postDelayed({
            val pindah = Intent(this, LoginActivity::class.java)
            startActivity(pindah)
            finish()
        }, 2000)// ketika aplikasi pertama kali dibuka dia menuju
        splashscreen dan akan delay
        // selama 2000s kmdn lgsg menuju halaman login

        //agar saat program dirun munculnya di splash page, kita hrs
        menyettingnya di manifest
    }
}

```

Cara Akses Database

1. Klik Tab View
2. Pilih ToolWindow dan pilih 'Device Explorer'
3. Pilih Data, cari berdasar package kalian masing-masing

