

# Tutorial Penggunaan Shared Preferences pada Android



Fikry Farenza

Follow

Jun 29, 2018 · 4 min read

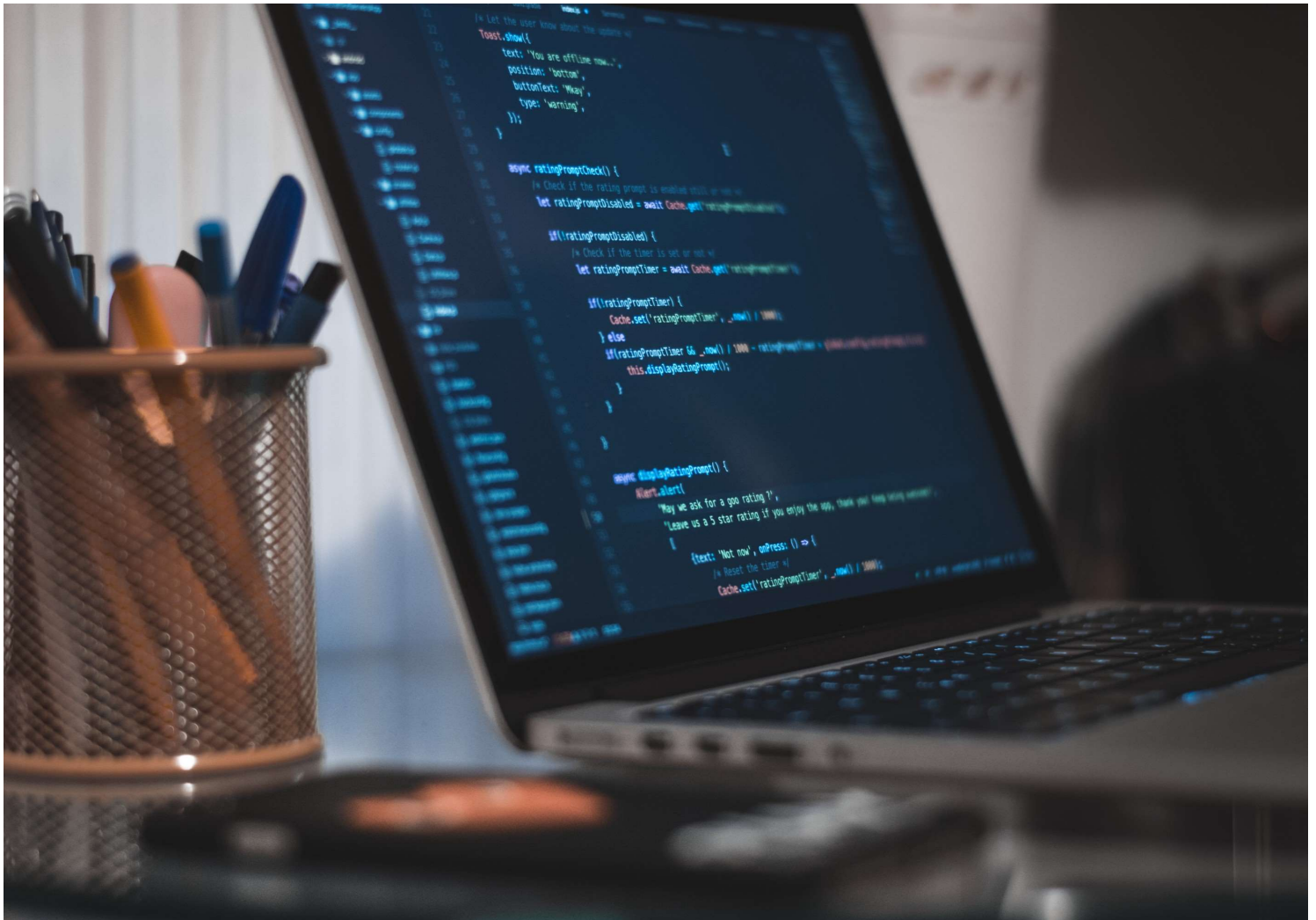


Photo by [Fabian Grohs](#) on [Unsplash](#)

**Di** Android, ada berbagai macam opsi storage. *Shared Preferences* adalah salah satunya. Karakter penyimpanan ini adalah key-value storage, Sehingga hanya

bisa menyimpan data bertipe primitif (*float, int, long, string, boolean*). Karenanya penyimpanan ini tergolong sangat minimalis dan sangat cocok untuk penyimpanan data yang sedikit.

Artikel ini akan menginformasikan Anda tentang dasar penggunaan dari *Shared Preference* dari pembuatan Aplikasi Login Sederhana.

Anda bisa mendownload Project ini [di sini](#)



## Lihat, apa yang bisa dia lakukan!

Tentu saja bisa *login*, *register*, dan *logout*. Tetapi dibalik layar, meskipun aplikasi telah ditutup, *Shared Preferences* mencegah aplikasi *logout*, kecuali Anda memutuskan untuk *logout*.

## Memangnya kenapa?

Anda tidak perlu melakukan *login* berulang kali setiap membuka aplikasinya. Sepele tapi penting bukan? Tertarik?

## Apa yang Anda Butuhkan?

1. Pengetahuan tentang Java setidaknya *basic* nya
2. Software Android Studio atau IntelliJ beserta alat untuk running app-nya
3. Koneksi internet setidaknya untuk mengambil referensi dari artikel ini ;P

## Let's Start

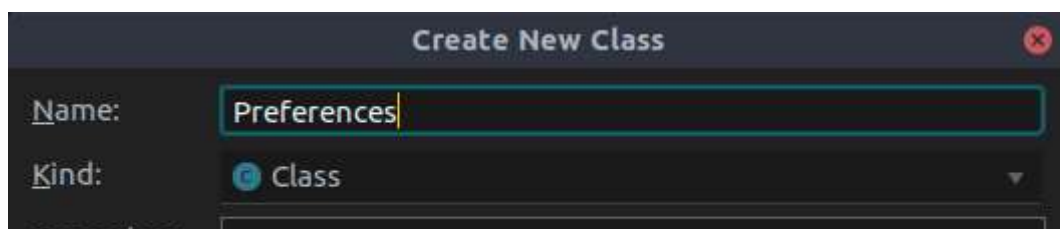
Memulai tutorial dengan memenuhi rekomendasi di atas akan sangat membantu dalam memahami tutorial, jadi?..  
Just Scroll down!

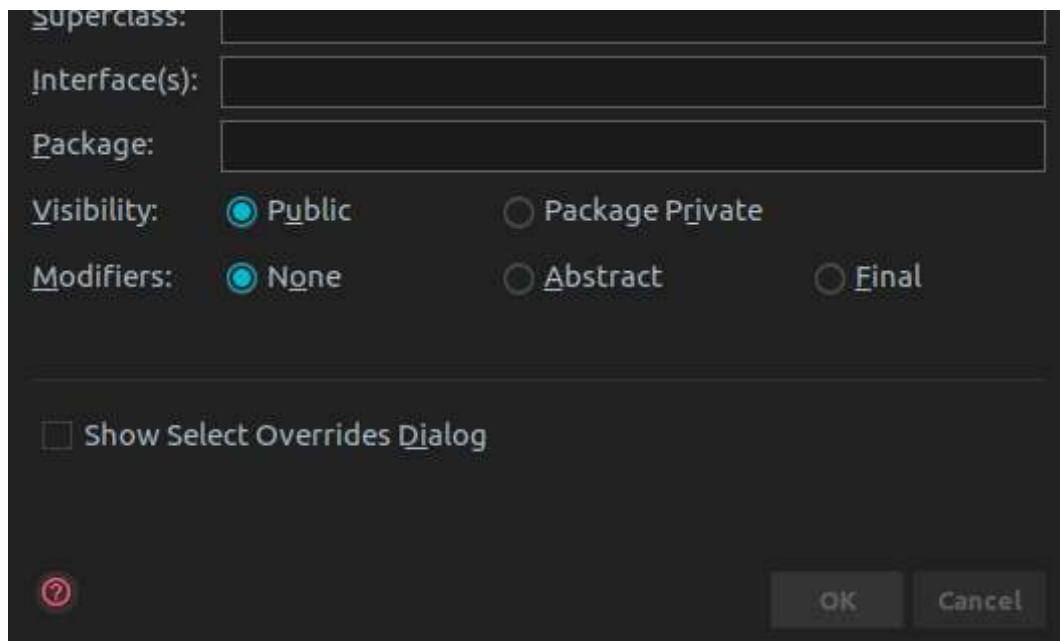
## Project Baru!

1. Pertama-tama, buatlah project android baru seperti biasa dengan nama yang ada di imajinasi kalian. Di tutorial ini, namanya 'Simple Login'.
2. Pilih Empty Activity sebagai main Activity saat muncul pemilihan inisialisasi Activity.
3. Shared Preferences tidak ada hubungannya dengan Gradle. Jadi tidak perlu sampai mengubah Gradle file untuk konfigurasi *Shared Preferences*.

## Shared Preferences

*Kita membutuhkan file berektensi .java baru sebagai tempat untuk Shared Preferences yang akan digunakan nanti.*





1. Anda bisa membuat Java baru dengan Klik File — New — Klik Java Class.
2. Tambahkan nama java di form 'Name:' dengan 'Preferences' seperti gambar di samping.
3. Setelah selesai memberikan nama, Anda bisa langsung klik tombol Ok di pojok kanan bawah *window*.
4. Berikut adalah class Preferences yang di gunakan dalam Tutorial ini.

```

1  import android.content.Context;
2  import android.content.SharedPreferences;
3  import android.preference.PreferenceManager;
4
5  public class Preferences {
6
7      /** Pendeklarasian key-data berupa String, untuk sebagai wadah penyimpanan data.
8       * Jadi setiap data mempunyai key yang berbeda satu sama lain */
9      static final String KEY_USER_TEREGISTER = "user", KEY_PASS_TEREGISTER = "pass";
10     static final String KEY_USERNAME_SEDANG_LOGIN = "Username_logged_in";
11     static final String KEY_STATUS_SEDANG_LOGIN = "Status_logged_in";
12
13     /** Pendeklarasian Shared Preferences yang berdasarkan paramater context */
14     private static SharedPreferences getSharedPreferences(Context context){
15         return PreferenceManager.getDefaultSharedPreferences(context);
16     }
17
18     /** Deklarasi Edit Preferences dan mengubah data

```

```
19  * yang memiliki key isi KEY_USER_TEREGISTER dengan parameter username */
20  public static void setRegisteredUser(Context context, String username){
21      SharedPreferences.Editor editor = getSharedPreferences(context).edit();
22      editor.putString(KEY_USER_TEREGISTER, username);
23      editor.apply();
24  }
25  /** Mengembalikan nilai dari key KEY_USER_TEREGISTER berupa String */
26  public static String getRegisteredUser(Context context){
27      return getSharedPreferences(context).getString(KEY_USER_TEREGISTER, "");
28  }
29
30  /** Deklarasi Edit Preferences dan mengubah data
31  * yang memiliki key KEY_PASS_TEREGISTER dengan parameter password */
32  public static void setRegisteredPass(Context context, String password){
33      SharedPreferences.Editor editor = getSharedPreferences(context).edit();
34      editor.putString(KEY_PASS_TEREGISTER, password);
35      editor.apply();
36  }
37  /** Mengembalikan nilai dari key KEY_PASS_TEREGISTER berupa String */
38  public static String getRegisteredPass(Context context){
39      return getSharedPreferences(context).getString(KEY_PASS_TEREGISTER, "");
40  }
41
42  /** Deklarasi Edit Preferences dan mengubah data
43  * yang memiliki key KEY_USERNAME_SEDANG_LOGIN dengan parameter username */
44  public static void setLoggedInUser(Context context, String username){
45      SharedPreferences.Editor editor = getSharedPreferences(context).edit();
46      editor.putString(KEY_USERNAME_SEDANG_LOGIN, username);
47      editor.apply();
48  }
49  /** Mengembalikan nilai dari key KEY_USERNAME_SEDANG_LOGIN berupa String */
50  public static String getLoggedInUser(Context context){
51      return getSharedPreferences(context).getString(KEY_USERNAME_SEDANG_LOGIN, "");
52  }
53
54  /** Deklarasi Edit Preferences dan mengubah data
55  * yang memiliki key KEY_STATUS_SEDANG_LOGIN dengan parameter status */
56  public static void setLoggedInStatus(Context context, boolean status){
57      SharedPreferences.Editor editor = getSharedPreferences(context).edit();
58      editor.putBoolean(KEY_STATUS_SEDANG_LOGIN, status);
59      editor.apply();
60  }
61  /** Mengembalikan nilai dari key KEY_STATUS_SEDANG_LOGIN berupa boolean */
62  public static boolean getLoggedInStatus(Context context){
63      return getSharedPreferences(context).getBoolean(KEY_STATUS_SEDANG_LOGIN, false);
```

```

64     }
65
66     /** Deklarasi Edit Preferences dan menghapus data, sehingga menjadikannya bernilai default
67     * khusus data yang memiliki key KEY_USERNAME_SEDANG_LOGIN dan KEY_STATUS_SEDANG_LOGIN */
68     public static void clearLoggedInUser (Context context){
69         SharedPreferences.Editor editor = getSharedPreferences(context).edit();
70         editor.remove(KEY_USERNAME_SEDANG_LOGIN);
71         editor.remove(KEY_STATUS_SEDANG_LOGIN);
72         editor.apply();
73     }
74 }

```

## Apa kegunaan method atau function yang ada di class Preferences di atas?

4. Di dalam *class Preferences*, tambahkan *mutator SharedPreferences* yang berguna untuk pendeklarasian *SharedPreferences* di *activity* yang berbeda-beda dan setiap kali menggunakan *SharedPreferences*, harus melalui *mutator* ini

```

public static SharedPreferences getSharedPreferences (Context
    context) {
    return PreferenceManager.getDefaultSharedPreferences (context);
}

```

5. Lalu deklarasi kan *key* yang akan kita gunakan nanti saat kita ingin menambah atau mengedit data

```

public class Preferences {
    static final String KEY_USER_TREGISTER =
        "user", KEY_PASS_TREGISTER = "pass";
    static final String KEY_USERNAME_SEDANG_LOGIN =
        "Username_logged_in";
    static final String KEY_STATUS_SEDANG_LOGIN =
        "Status_logged_in";
}

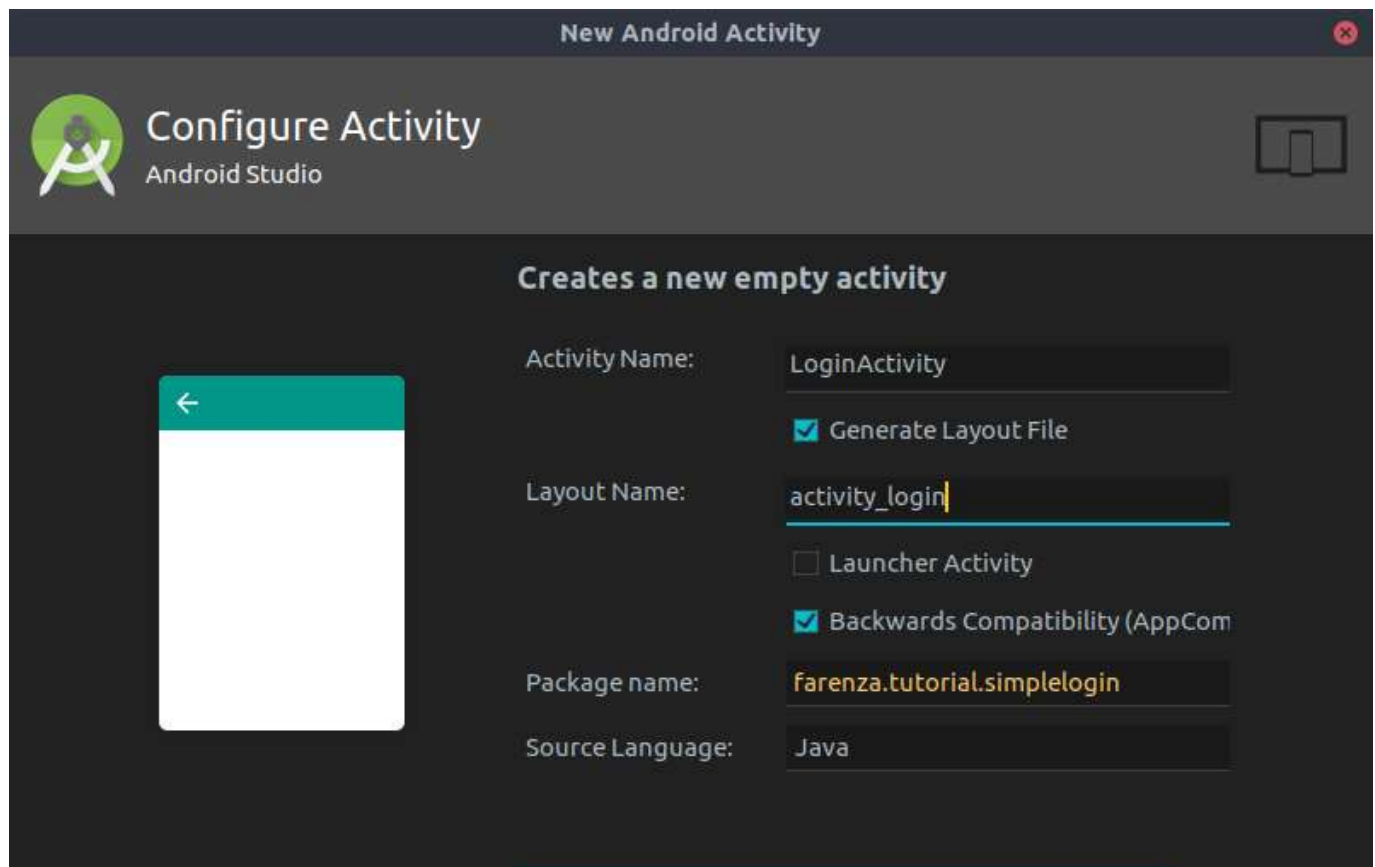
```

6. Tambahkan getter setter untuk semua key yang sudah di deklarasi atau Anda bisa menekan `ctrl + alt + insert`
7. Buat semua getter dan setter memiliki `public static void` serta berparameter **Context**. Khusus setter, tambahkan parameter **String**.
8. Tambahkan code ini untuk semua

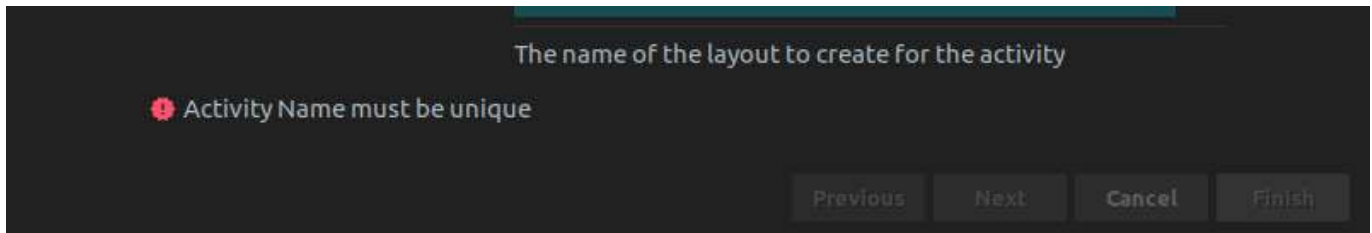
```
public static void setRegisteredUser(Context context, String
    username) {
    SharedPreferences.Editor editor =
    getSharedPreferences(context).edit();
    editor.putString(KEY_USER_TEREGISTER, username);
    editor.apply();
}
```

## Form untuk Login dan Register

*Untuk membuat Formulir Login dan Register, biasanya membutuhkan setidaknya 2 activity untuk membuatnya.*







Muncul Window setelah klik Empty Activity

1. Anda bisa mengklik langsung Klik File — New — Activity — Klik Empty Activity. Lalu akan muncul window baru seperti di samping.
2. Gantilah nama Activity, di form berlabel 'Activity Name:' dengan 'LoginActivity'[tanpa spasi] atau dengan nama lain yang Anda inginkan. Lalu bisa langsung klik Finish.
3. Anda hanya perlu menambahkan code berikut dalam tombol login

```
SharedPreferences.Editor editor =
getSharedPreferences(context).edit();
editor.putBoolean(KEY_STATUS_SEDANG_LOGIN,status); editor.apply();
```

Namun, jika sudah membuat class Preferences seperti langkah di atas, Anda hanya perlu menambahkan code berikut

```
Preferences.setLoggedInStatus(getBaseContext(), true);
```

[Klik disini untuk melihat contoh full codenya](#)

3. Lakukan langkah nomer 2 lagi untuk membuat Activity baru untuk Register dengan nama yang berbeda contohnya 'RegisterActivity'.

## RegisterActivity

*Sebuah Activity untuk mendaftarkan user ke SharedPreferences. [Silahkan lihat full code nya disini](#)*

1. Di layout ini, cukup berikan 3 *EditText* sebagai tempat mengisi form pendaftaran, seperti berikut



```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <EditText
        android:id="@+id/et_emailSignup"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Email"
        android:inputType="textEmailAddress"
        android:maxLines="1"
        android:singleLine="true"/>
    <EditText
        android:id="@+id/et_passwordSignup"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Password"
        android:inputType="textPassword"
        android:maxLines="1"
        android:singleLine="true" />
    <EditText
        android:id="@+id/et_passwordSignup2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/label_confirm_password"
        android:inputType="textPassword"
        android:maxLines="1"
        android:singleLine="true" android:imeActionId="6"
        android:imeActionLabel="@string/label_signup"
        android:imeOptions="actionUnspecified" />

```

2. Dan ini adalah contoh dari Login Activity layoutnya beserta javanya.

## The Main Activity

*Karena MainActivity sudah terbuat saat awal pertama kali membuat project, jadi Anda tidak perlu melakukan langkah untuk membuat Activity baru lagi.*

1. Berikan 2 TextView untuk menampilkan label dan user yang sedang login di *layout* `activity_main.xml`

```

<LinearLayout
    ....
    ....
    android:orientation="vertical">

```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="20sp"
    android:text="Masuk sebagai"/>
<TextView
    android:id="@+id/tv_namaMain"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="30sp"
    tools:text="Namaku adalah"/>
</LinearLayout>

```

## 2. Berikan *Button* untuk *logout* user seperti di bawah contohnya

```

<Button
    android:id="@+id/button_logoutMain"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Log Out" />

```

## 3. Sekarang di javanya, kita set *TextView* dengan nama user yang sedang login seperti code di bawah

```

TextView nama = findViewById(R.id.tv_namaMain);
nama.setText(Preferences.getLoggedInUser(getBaseContext()));

```

## 4. Berikan aksi untuk *logout* di *Button* nya

```

findViewById(R.id.button_logoutMain).setOnClickListener(new
    View.OnClickListener() {
        @Override public void onClick(View v) {
            Preferences.clearLoggedInUser(getBaseContext());
            startActivity(new
                Intent(getBaseContext(), LoginActivity.class));
            finish();
        }
    });

```

## Horee selesai

Begitulah tutorial tentang Shared Preferences. Jika ada salah kata mohon maaf lahir batin. Silahkan tinggalkan jejak untuk referensi artikel saya agar bisa lebih baik.

Selamat belajar dan berkarya

[Android Development](#)[Code Newbie](#)[Java](#)[Tutorial](#)[Mobile App Development](#)[About](#) [Help](#) [Legal](#)

Get the Medium app

