

Tutorial 1

I. Git

Selain belajar membuat aplikasi android, di kelas ini kalian akan diberikan sebuah pengalaman baru dengan menggunakan Git.

Version Control System (VCS)

VCS merupakan sebuah software yang membantu para developer untuk saling bekerja sama dan tetap mempertahankan riwayat kerja mereka secara utuh.

VCS memiliki beberapa fungsi, di antaranya mengizinkan para developer untuk bekerja secara bersamaan, tidak mengizinkan saling tumpang-tindih perubahan yang ada, dan mempertahankan riwayat dari setiap versi.

VCS sendiri terbagi menjadi dua tipe, yaitu Centralized VCS (CVCS) dan Distributed/Decentralized VCS (DVCS).

Git merupakan VCS yang sering digunakan sekarang ini. Git merupakan salah satu Distributed Version Control System (DVCS), yang berarti salinan kode yang tersimpan di local repository adalah sebuah version control repository yang lengkap.

Jadi, antara local repository dan remote repository sama-sama berisi versi terakhir (terupdate).

Perintah di Git

Untuk melihat berbagai perintah (command) yang ada di Git, kamu bisa melihatnya di sini.



Tutorial Git

1. Buat akun di github.com
2. Buat sebuah repository baru seperti berikut

Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)


Owner Repository name *

 Hildayutami / 

Great repository names are short and memorable. Need inspiration? How about **psychic-spork**?


Description (optional)

☒  **Public**
Anyone can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

☐ **Initialize this repository with a README**
This will let you immediately clone the repository to your computer.



Create repository

3. Download dan install git bash: <https://git-scm.com/downloads>
4. Buatlah sebuah folder yang akan menjadi local repository yang kamu gunakan,. Nama folder bebas. *Catatan: dalam soal tutorial ini, nama folder yang digunakan adalah **“Lab”**, kalian boleh menggantinya.
5. Buka git bash, masuk ke dalam folder **Lab**
6. Buat sebuah folder baru dengan nama Tutorial 1
7. Jalankan perintah git init pada command prompt/shell. (ini akan menginisiasi Git pada direkori yang ditunjuk oleh Command Prompt, pastikan kalian tidak salah menginisiasi folder)

```
C:\Users\hld>E:
E:\>cd Lab
E:\Lab>git init
Initialized empty Git repository in E:/Lab/.git/
E:\Lab>
```

8. Jalankan perintah git status untuk memastikan bahwa inisiasi repository Git sudah berhasil dilakukan.
9. Atur username dan email yang akan diasosiasikan dengan pekerjaanmu di repository Git dengan perintah berikut: git config --global user.name "<nama-lengkapmu>" git config --global user.email "<masukkan-emailmu>" ini untuk memberi informasi tentang pengguna repo, masukan email dan nama kalian.

```
E:\Lab>git config --global user.name "Hildayutami"
E:\Lab>git config --global user.email "nur.hildayanti.u@gmail.com"
```

10. Setelah melakukan konfigurasi, Buatlah file Halo.txt di dalam folder "Tutorial 1".
11. Setelah itu, jalankan perintah git status pada command prompt/shell, maka nanti akan muncul daftar untracked files dari folder tersebut.

```
E:\Lab>git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        Tutorial 1/

nothing added to commit but untracked files present (use "git add" to track)
E:\Lab>
```

12. Jalankan perintah git add . untuk setiap untracked files.

```
E:\Lab>git add .
E:\Lab>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   Tutorial 1/hello.txt
```

13. Setelah itu, jalankan perintah git commit -m "<isi-dengan-pesanmu>". Hal tersebut berfungsi sebagai penanda terhadap perubahan yang terjadi di dalam files yang kamu buat sebelumnya.

```
E:\Lab>git commit -m "first commit"
[master (root-commit) 88df8a8] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Tutorial 1/hello.txt
```

14. Jalankan perintah git remote add origin “<isi-dengan-repository-kalian>” *misal repository saya adalah : <https://github.com/Hildayutami/TutorialLab.git>

```
E:\Lab>git remote add origin https://github.com/Hildayutami/TutorialLab.git
```

15. Jalankan perintah git push -u origin master

```
E:\Lab>git push -u origin master
Logon failed, use ctrl+c to cancel basic credential prompt.
Username for 'https://github.com': nur.hildayanti.u@gmail.com
Password for 'https://nur.hildayanti.u@gmail.com@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Writing objects: 100% (4/4), 262 bytes | 262.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To https://github.com/Hildayutami/TutorialLab.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

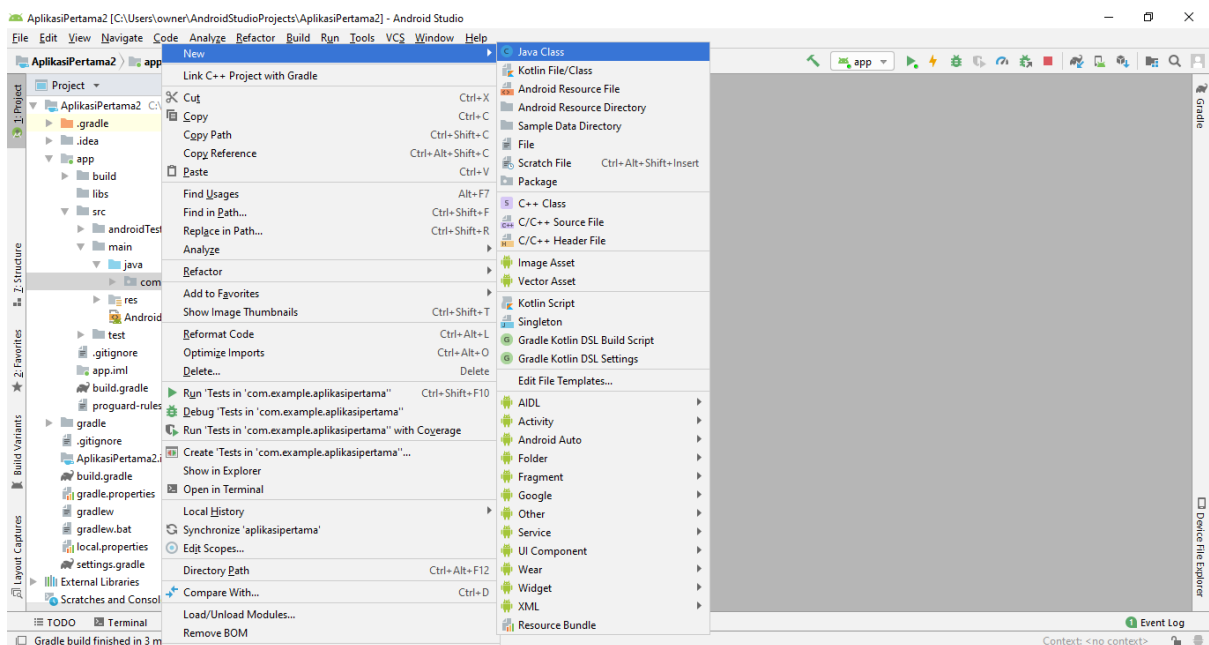
16. Cek repository kalian di github

The screenshot shows the GitHub interface for the repository 'Hildayutami / TutorialLab'. At the top, there are buttons for 'Unwatch', 'Star' (0), and 'Fork' (0). Below this is a navigation bar with links to 'Code', 'Issues' (0), 'Pull requests' (0), 'Projects' (0), 'Wiki', 'Security', 'Insights', and 'Settings'. The main content area displays repository statistics: '1 commit', '1 branch', '0 releases', and '1 contributor'. There are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and 'Clone or download'. A list of commits is shown, including 'Hildayutami first commit' (latest commit 88df8a8, 6 minutes ago) and 'Tutorial 1' (first commit, 6 minutes ago). At the bottom, there is a prompt to 'Add a README' to help people understand the project.

II. Aplikasi Android Pertama

Membuat Activity (File disimpan dalam folder yang telah di buat sebelumnya)

1. Buat Sebuah Class baru dengan cara Klik Kanan pada App -> New -> Class lalu beri nama activity.



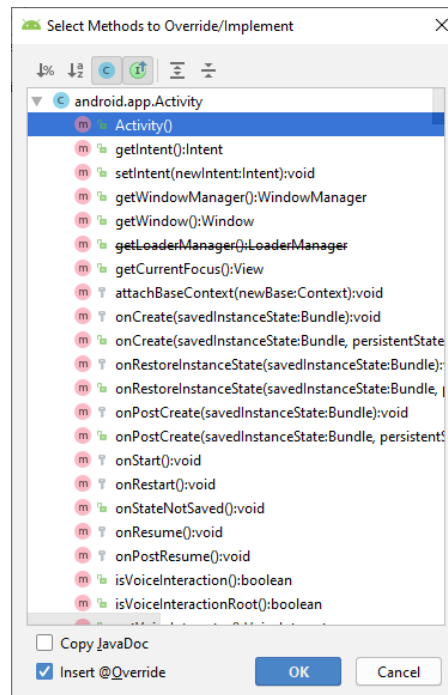
2. Turunkan Class Activity pada Class yang baru Anda buat dengan keyword **extends**.

```

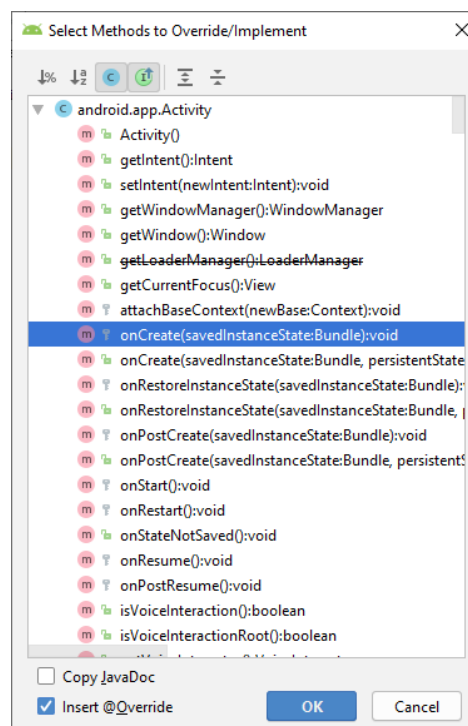
ActivityContoh.java x
1 package com.example.aplikasipertama;
2
3 import android.app.Activity;
4
5 public class ActivityContoh extends Activity {
6
7
  
```

**Biasakan menggunakan Ctrl + Space*

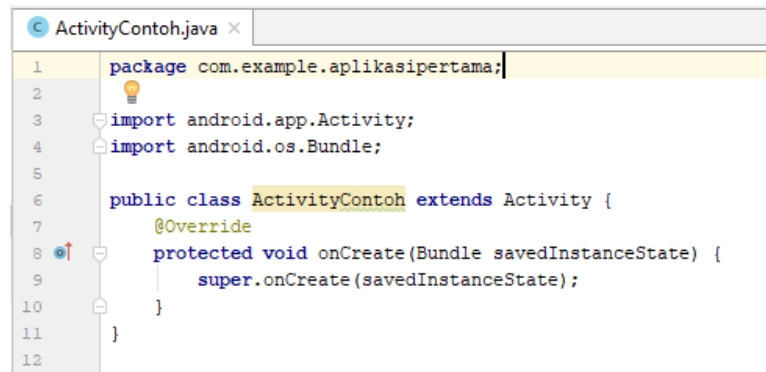
3. Tekan Ctrl + O untuk memunculkan method-method Override, sehingga akan muncul jendela baru seperti berikut:



4. Selanjutnya ketik huruf “**onc**” sehingga pilihan yang aktif akan menunjuk fungsi **onCreate** seperti gambar berikut:



5. Tekan Enter sehingga halaman kerja akan menjadi seperti berikut:

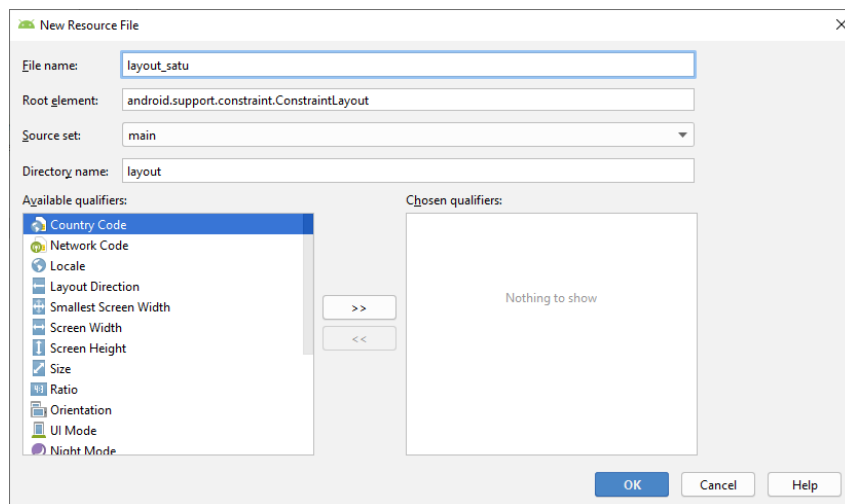


```

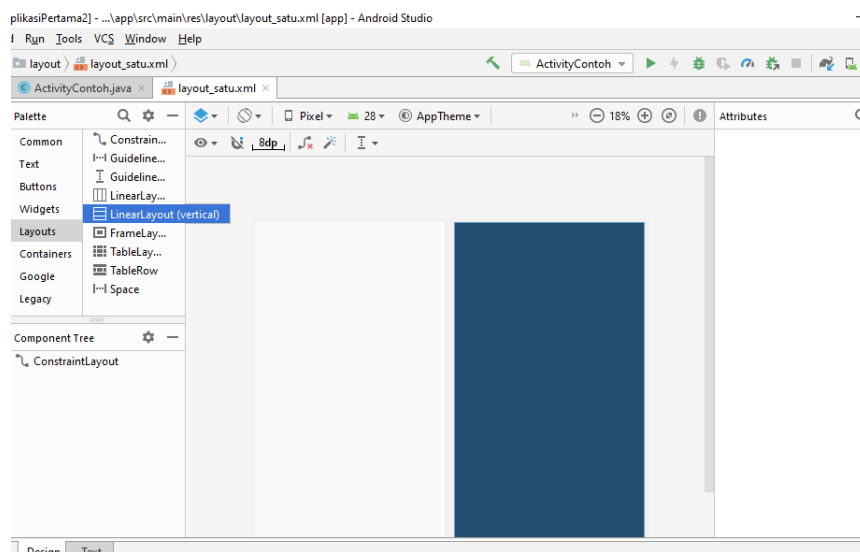
1 package com.example.aplikasipertama;
2
3 import android.app.Activity;
4 import android.os.Bundle;
5
6 public class ActivityContoh extends Activity {
7     @Override
8     protected void onCreate(Bundle savedInstanceState) {
9         super.onCreate(savedInstanceState);
10    }
11 }
12

```

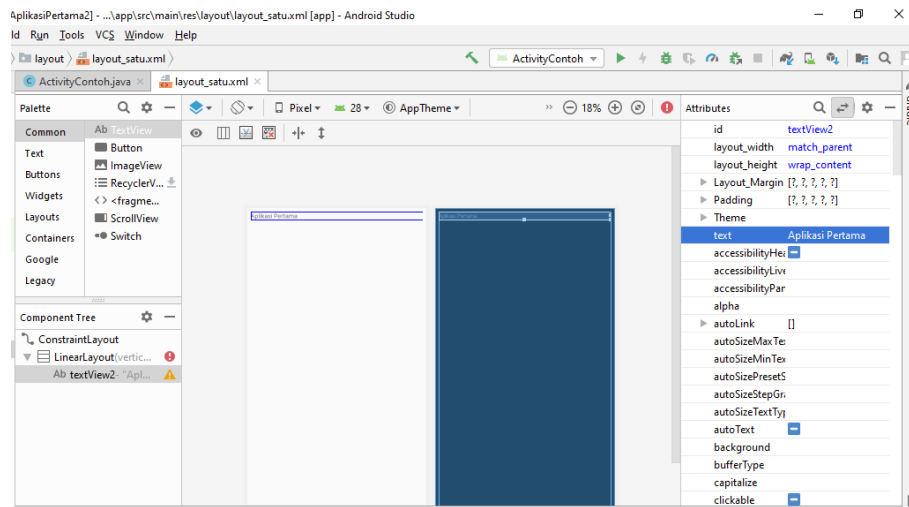
6. Selanjutnya pada sub-folder **layout**, lakukan klik kanan lalu pilih New->Layout resource file.



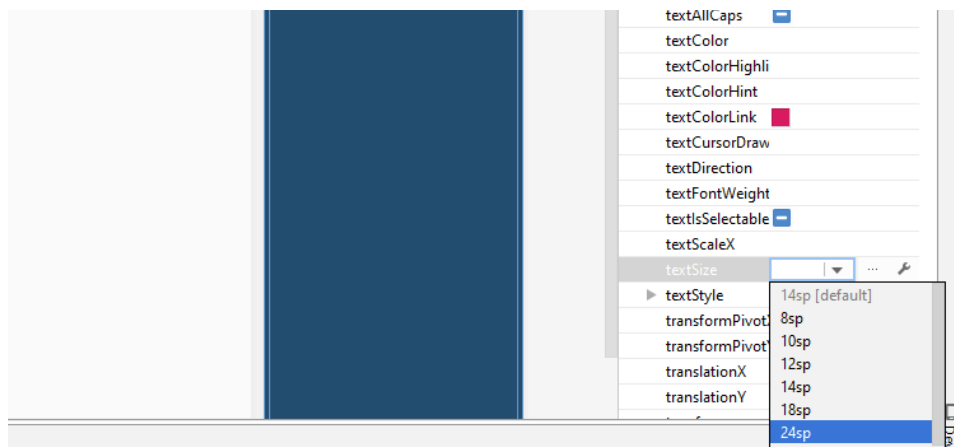
7. Buka bagian Design di layout, lalu drag and drop Linear Layout.



8. Lalu drag and drop sebuah TextView dan ubah properties bagian **text** menjadi Aplikasi Pertama seperti berikut:



9. Ubah ukuran text dibagian textSize seperti berikut:



10. Kembali ke Class ActivityContoh lalu ketikkan kodingan berikut setelah baris **super.onCreate(savedInstanceState)**.

```

ActivityContoh.java
1  package com.example.aplikasipertama;
2
3  import android.app.Activity;
4  import android.os.Bundle;
5
6  public class ActivityContoh extends Activity {
7      @Override
8      protected void onCreate(Bundle savedInstanceState) {
9          super.onCreate(savedInstanceState);
10         setContentView(R.layout.layout_satu);
11     }
12 }

```


11. Deklarasikan Activity (Class) yang baru dibuat ke file Manifest. Buka file AndroidManifest.xml lalu ketikkan baris berikut:

```


<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="AplikasiPertama"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportRtl="true"
    android:theme="@style/AppTheme">
    <activity android:name=".ActivityContoh">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

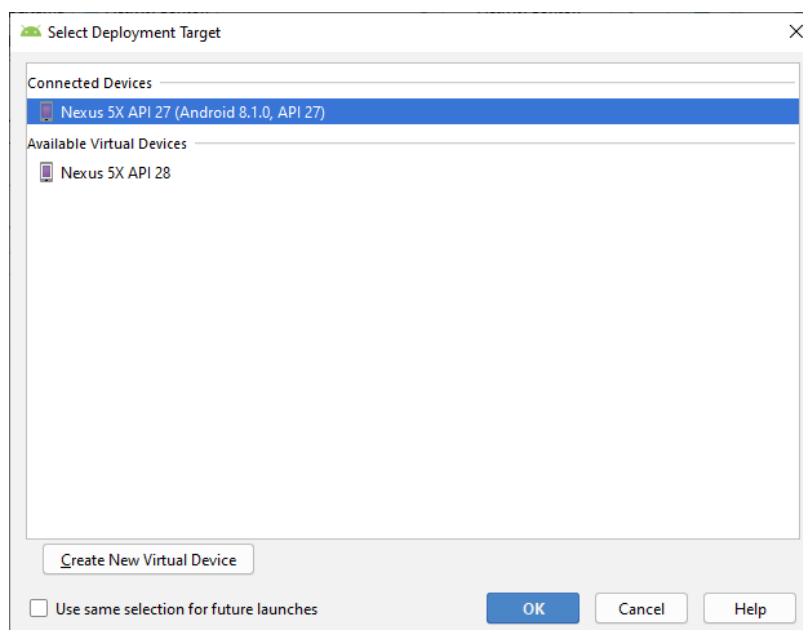
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
    <activity android:name=".MainActivity">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

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

```

****Ingat! Usahakan seminimal mungkin mengetik manual selama bisa memanfaatkan Ctrl + Space***

12. Jalankan program dengan menekan icon  atau menekan Ctrl + Shift + F10. Akan muncul layar baru untuk memilih Emulator.



13. Akan muncul tampilan seperti berikut:

