

# Workbook Git

## Basic Commands



Endy Muhardin  
[endy@artivisi.com](mailto:endy@artivisi.com)  
last updated : 18 Mar 2011

## Daftar Isi

Membuat Repository.....	3
Penjelasan.....	3
Panduan.....	3
Kesimpulan.....	5
Diff dan Patch.....	6
Penjelasan.....	6
Panduan.....	6
Kesimpulan.....	7
Menggunakan Interactive Staging.....	8
Penjelasan.....	8
Panduan.....	8
Kesimpulan.....	9
Memahami Staging Area.....	10
Penjelasan.....	10
Panduan.....	10
Kesimpulan.....	12
Autocommit.....	13
Penjelasan.....	13
Panduan.....	13
Kesimpulan.....	13
Melihat isi repository.....	14
Penjelasan.....	14
Panduan.....	14
Kesimpulan.....	14
Membatalkan perubahan.....	15
Penjelasan.....	15
Panduan.....	15
Kesimpulan.....	15

# Membuat Repository

## Penjelasan

Pada latihan ini, kita akan membuat repository Git untuk menyimpan histori perubahan.

## Panduan

No	Aktifitas	Hasil
1	Buat folder kosong, misalnya /home/andy/tmp/belajar-git	
2	Isi folder dengan file source code	
3	Inisialisasi repository git dengan perintah :  git init	Initialized empty Git repository in /home/andy/tmp/belajar-git/.git/
4	Ketik perintah ini untuk melihat posisi working folder  git status	# On branch master # # Initial commit # # Untracked files: # (use "git add <file>..." to include in what will be committed) # # BelajarGit/
5	Daftarkan file dan folder yang ada untuk disimpan dalam repository (staging)  git add .	
6	Lihat lagi posisi working folder  git status  Folder build, dist, dan private tidak ingin kita simpan ke repository, jadi harus ditulis di ignore list.	# On branch master # # Initial commit # # Changes to be committed: # (use "git rm --cached <file>..." to unstage) # # new file: BelajarGit/build.xml # new file: BelajarGit/manifest.mf # new file: BelajarGit/nbproject/build-impl.xml # new file: BelajarGit/nbproject/genfiles.properties # new file: BelajarGit/nbproject/private/private.properties # new file: BelajarGit/nbproject/project.properties # new file: BelajarGit/nbproject/project.xml
7	Buat file .gitignore untuk membuat perkecualian. Lokasinya sejajar dengan folder .git  Berikut isi file .gitignore  # Netbeans build dist	

	private	
8	Batalkan staging git rm --cached -r .	rm 'BelajarGit/build.xml' rm 'BelajarGit/build/built-jar.properties' rm 'BelajarGit/build/classes/belajar/git/Halo.class' rm 'BelajarGit/dist/BelajarGit.jar' rm 'BelajarGit/manifest.mf' rm 'BelajarGit/nbproject/build-impl.xml' rm 'BelajarGit/nbproject/genfiles.properties' rm 'BelajarGit/nbproject/private/private.properties' rm 'BelajarGit/nbproject/project.properties' rm 'BelajarGit/nbproject/project.xml' rm 'BelajarGit/src/belajar/git/Halo.java'
9	Lakukan staging lagi, kemudian lihat statusnya  git add .  git status  File dan folder yang ada di .gitignore tidak ikut di staging	# On branch master # # Initial commit # # Changes to be committed: # (use "git rm --cached <file>..." to unstage) # #     new file:   .gitignore #     new file:   BelajarGit/build.xml #     new file:   BelajarGit/manifest.mf #     new file:   BelajarGit/nbproject/build-impl.xml #     new file:   BelajarGit/nbproject/genfiles.properties #     new file:   BelajarGit/nbproject/project.properties #     new file:   BelajarGit/nbproject/project.xml #     new file:   BelajarGit/src/belajar/git/Halo.java
10	Simpan isi staging area ke repository git commit -m "commit pertama"	[master (root-commit) c70c58c] commit pertama 8 files changed, 1084 insertions(+), 0 deletions(-) create mode 100644 .gitignore create mode 100644 BelajarGit/build.xml create mode 100644 BelajarGit/manifest.mf create mode 100644 BelajarGit/nbproject/build-impl.xml create mode 100644 BelajarGit/nbproject/genfiles.properties create mode 100644 BelajarGit/nbproject/project.properties create mode 100644 BelajarGit/nbproject/project.xml create mode 100644 BelajarGit/src/belajar/git/Halo.java
11	Lihat hasilnya git log	commit c70c58c0e806c4967ee1bf2ae25687b737ad7f40 Author: Endy Muhardin <endy@artivisi.com> Date: Wed Jan 12 10:07:42 2011 +0700  commit pertama
12	Lihat log (versi singkat) git log --pretty=oneline	c70c58c0e806c4967ee1bf2ae25687b737ad7f40 commit pertama

13	Lihat log (tampilkan tanggal dan nama)  git log --pretty=format:'%ad - %an - %h - %s'	Wed Jan 12 10:07:42 2011 +0700 - Endy Muhardin - c70c58c - commit pertama
----	---	---

## Kesimpulan

Git memiliki 3 bagian utama, yaitu working folder, staging area, dan permanent area. Perintah add memasukkan perubahan dari working folder ke staging area. Perintah commit memasukkan perubahan dari staging area ke permanent area.

# Diff dan Patch

## Penjelasan

Konsep sentral dalam version control adalah diff dan patch. Latihan berikut akan membuat kita lebih paham tentang diff dan patch.

## Panduan

No	Aktifitas	Hasil
1	Buat file kalkulator.php  <pre>&lt;? function tambah(x,y){     return x+y; }  ?&gt;</pre>	
2	Buat file kalkulator2.php  <pre>&lt;? function tambah(x,y){     return x+y; }  function kurang(x,y){     return x-y; }  ?&gt;</pre>	
3	Hitung diff untuk mengubah kalkulator.php menjadi kalkulator2.php  <pre>diff -u kalkulator.php kalkulator2.php &gt; patch-kalkulator.txt</pre>	<pre>-- kalkulator.php      2011-01-12 13:36:25.993371206 +0700 +++ kalkulator2.php    2011-01-12 13:37:17.149371206 +0700 @@ -4,4 +4,8 @@      return x+y; }  +function kurang(x,y){ + return x-y; +} + ?&gt;</pre>
4	Copy kalkulator.php dan patch-kalkulator.txt ke folder berbeda	
5	Aplikasikan patch-kalkulator.txt ke kalkulator.php untuk mengubahnya menjadi kalkulator2.php  <pre>patch kalkulator.php patch-kalkulator.txt</pre>	<pre>patching file kalkulator.php  Isi file kalkulator.php menjadi sama dengan kalkulator2.php</pre>
6	Patch bisa diaplikasikan secara terbalik (reversed) untuk membatalkan perubahan	<pre>patching file kalkulator.php  Isi file kalkulator.php kembali ke kondisi</pre>

	<code>patch -R kalkulator.php patch-kalkulator.txt</code>	sebelum dilakukan patch.
--	---	--------------------------

## Kesimpulan

Diff adalah perangkat untuk membuat selisih antara dua file. Selisih ini dapat dipertukarkan dengan orang lain untuk menyelaraskan source code.

Untuk mengaplikasikan diff, kita gunakan aplikasi patch.

# Menggunakan Interactive Staging

## Penjelasan

Staging area digunakan untuk merapikan perubahan yang ingin kita commit. Jika kita menggunakan Subversion, semua perubahan yang ada dalam satu file akan disimpan pada saat commit. Git memungkinkan kita untuk memilih perubahan mana yang ingin disimpan.

## Panduan

No	Aktifitas	Hasil
1	Buat perubahan di 2 file	
2	Lihat status working folder  git status	# On branch master # Changed but not updated: # (use "git add <file>..." to update what will be committed) # (use "git checkout -- <file>..." to discard changes in working directory) # # modified: BelajarGit/src/belajar/git/Halo.java # # Untracked files: # (use "git add <file>..." to include in what will be committed) # #  BelajarGit/src/belajar/git/Kalkulator.java
3	Tampilkan menu untuk memilih mana yang akan dimasukkan ke staging area  git add -i	staged unstage path 1: unchanged +3/-1 BelajarGit/src/belajar/git/Halo.java  *** Commands *** 1: status 2: update 3: revert 4: add untracked 5: patch 6: diff 7: quit 8: help What now>
4	Tambahkan file Kalkulator.java  Tekan 4  Tekan 1 untuk memilih file Kalkulator.java  Tekan Enter untuk selesai	What now> a 1: BelajarGit/src/belajar/git/Kalkulator.java Add untracked>> 1 * 1: BelajarGit/src/belajar/git/Kalkulator.java Add untracked>> added one path
5	Lihat lagi statusnya  Tekan 1	*** Commands *** 1: status 2: update 3: revert 4: add untracked 5: patch 6: diff 7: quit 8: help What now> 1 staged unstage path 1: unchanged +3/-1 BelajarGit/src/belajar/git/Halo.java 2: +22/-0 nothing BelajarGit/src/belajar/git/Kalkulator.java
6	Keluar dari menu interactive staging	*** Commands ***



	Tekan 7	1: status            2: update            3: revert 4: add untracked 5: patch            6: diff    7: quit    8: help What now> 7 Bye.
7	Lihat status  git status	# On branch master # Changes to be committed: # (use "git reset HEAD <file>..." to unstage) # #        new file: BelajarGit/src/belajar/git/Kalkulator.java # # Changed but not updated: # (use "git add <file>..." to update what will be committed) # (use "git checkout -- <file>..." to discard changes in working directory) # #        modified: BelajarGit/src/belajar/git/Halo.java #
8	Commit, simpan hanya file baru Kalkulator.java  git commit -m "tambah satu file, fitur kalkulator"	[master 07a38f2] tambah satu file, fitur kalkulator 1 files changed, 22 insertions(+), 0 deletions(-) create mode 100644 BelajarGit/src/belajar/git/Kalkulator.java

## Kesimpulan

Dengan menggunakan menu interactive staging, kita dapat memilih perubahan mana yang akan dimasukkan ke staging area untuk selanjutnya dicommit.

# Memahami Staging Area

## Penjelasan

Staging area bukan hanya dapat digunakan untuk memilih file, tapi juga untuk memilih hunk (potongan kode) yang ingin dicommit.

## Panduan

No	Aktifitas	Hasil
1	Buat perubahan di file Kalkulator	
2	Jalankan menu interactive staging	<pre>staged  unstaged path 1:  unchanged  +3/-1 BelajarGit/src/belajar/git/Halo.java 2:  unchanged  +6/-2 BelajarGit/src/belajar/git/Kalkulator.java  *** Commands *** 1: status      2: update      3: revert 4: add untracked 5: patch       6: diff       7: quit   8: help What now&gt;</pre>
3	Pilih opsi 5, patch Kemudian pilih file yang mau diproses, yaitu nomer 2.	<pre>What now&gt; 5 staged  unstaged path 1:  unchanged  +3/-1 BelajarGit/src/belajar/git/Halo.java 2:  unchanged  +6/-2 BelajarGit/src/belajar/git/Kalkulator.java Patch update&gt;&gt; 2 staged  unstaged path 1:  unchanged  +3/-1 BelajarGit/src/belajar/git/Halo.java * 2:  unchanged  +6/-2 BelajarGit/src/belajar/git/Kalkulator.java Patch update&gt;&gt;</pre>
4	Git akan menampilkan potongan kode yang berubah. Pilihan action yang bisa kita lakukan ada di bawahnya. Pilih ? untuk menampilkan penjelasan	<pre>diff --git a/BelajarGit/src/belajar/git/Kalkulator.java b/BelajarGit/src/belajar/git/Kalkulator.java index fa1207f..94281d8 100644 --- a/BelajarGit/src/belajar/git/Kalkulator.java +++ b/BelajarGit/src/belajar/git/Kalkulator.java @@ -11,12 +11,16 @@ package belajar.git;  */ public class Kalkulator {     public int tambah(int a, int b){ -        return 0; +        return a + b;     }     public int kurang(int a, int b){ -        return 0; +        return a - b;     }     public int kali(int a, int b){ +        return a * b; +    } +    public double bagi(double a, double b){         return 0;     } }</pre>

		<pre> } + } Stage this hunk [y,n,q,a,d,/,s,e,]? ? y - stage this hunk n - do not stage this hunk q - quit, do not stage this hunk nor any of the remaining ones a - stage this and all the remaining hunks in the file d - do not stage this hunk nor any of the remaining hunks in the file g - select a hunk to go to /- search for a hunk matching the given regex j - leave this hunk undecided, see next undecided hunk J - leave this hunk undecided, see next hunk k - leave this hunk undecided, see previous undecided hunk K - leave this hunk undecided, see previous hunk s - split the current hunk into smaller hunks e - manually edit the current hunk ? - print help </pre>
5	<p>Kadangkala, potongan kodenya terlalu besar sehingga sulit dipilih. Gunakan perintah split [s] untuk memecah lagi potongan kodenya.</p> <p>Setelah dipecah, kita bisa pilih untuk mengcommit method tambah, kurang, kali, dan tidak mengikutsertakan method bagi yang belum selesai.</p>	<pre> Stage this hunk [y,n,q,a,d,/,s,e,]? s Split into 4 hunks. @@ -11,6 +11,6 @@ */ public class Kalkulator {     public int tambah(int a, int b){         return 0;     }     public int kurang(int a, int b){ Stage this hunk [y,n,q,a,d,/,j,j,g,e,]? y @@ -15,5 +15,5 @@     }     public int kurang(int a, int b){         return 0;     }     public int kali(int a, int b){ Stage this hunk [y,n,q,a,d,/,K,j,j,g,e,]? y @@ -18,4 +18,7 @@     }     public int kali(int a, int b){         return a * b;     }     public double bagi(double a, double b){         return 0;     } Stage this hunk [y,n,q,a,d,/,K,j,j,g,e,]? </pre>
6	<p>Setelah selesai, lihat statusnya</p> <p>git status</p> <p>Terlihat bahwa file Kalkulator terdaftar di dua tempat, to be committed dan not updated. Ini disebabkan karena ada sebagian isi file yang tidak masuk ke staging area.</p>	<pre> # On branch master # Changes to be committed: #   (use "git reset HEAD &lt;file&gt;..." to unstage) # #       modified: #       BelajarGit/src/belajar/git/Kalkulator.java # # Changed but not updated: #   (use "git add &lt;file&gt;..." to update what will #   be committed) </pre>

		<pre># (use "git checkout -- &lt;file&gt;..." to discard changes in working directory) # #       modified: BelajarGit/src/belajar/git/Halo.java #       modified: BelajarGit/src/belajar/git/Kalkulator.java</pre>
7	Commit file	<pre>[master 8a4fbcf] implement fitur tambah, kurang, kali 1 files changed, 3 insertions(+), 2 deletions(-)</pre>
8	Lihat hasilnya	

## Kesimpulan

Dengan menggunakan menu interactive staging, kita dapat memilih perubahan mana yang akan dimasukkan ke staging area untuk selanjutnya dicommit. Perubahan yang dipilih bisa detail per potongan kode (hunk dalam istilah git), bukan cuma file per file.

# Autocommit

## Penjelasan

Kadangkala penggunaan staging area terlalu merepotkan. Untuk itu, kita bisa langsung commit dari working folder.

## Panduan

No	Aktifitas	Hasil
1	Buat perubahan di file	
2	Lihat status  git status	# On branch master # Changed but not updated: # (use "git add <file>..." to update what will be committed) # (use "git checkout -- <file>..." to discard changes in working directory) # #       modified: BelajarGit/src/belajar/git/Halo.java #       modified: BelajarGit/src/belajar/git/Kalkulator.java # no changes added to commit (use "git add" and/or "git commit -a")
3	Commit langsung dari working folder  git commit -am "tambahan fitur pembagian"	[master d6a8455] tambahan fitur pembagian 2 files changed, 7 insertions(+), 2 deletions(-)
4	Lihat lognya  git log	

## Kesimpulan

Dengan menggunakan opsi -a, kita tidak perlu repot menambahkan dulu ke staging area sebelum commit.

# Melihat isi repository

## Penjelasan

Berbagai perintah untuk melihat isi repository

## Panduan

No	Aktifitas	Hasil
1	Informasi singkat history  git log --pretty=oneline --abbrev-commit	3b0c807 implement hello world 4f0a687 fitur pembagian 1f7282a fitur perkalian 382c959 fitur kurang 1fecded add fitur tambah 07a38f2 tambah satu file, fitur kalkulator c70c58c commit pertama
2	Melihat isi satu commit  git show 3b0c807	commit 3b0c80761f519d55499e7de0ee67993a1e4130d b Author: Endy Muhardin <endy@artivisi.com> Date: Wed Jan 12 11:29:54 2011 +0700  implement hello world  diff --git a/BelajarGit/src/belajar/git/Halo.java b/BelajarGit/src/belajar/git/H index 3e234c7..6f1ccae 100644 --- a/BelajarGit/src/belajar/git/Halo.java +++ b/BelajarGit/src/belajar/git/Halo.java @@ -10,5 +10,7 @@ package belajar.git; * @author endy */ public class Halo { - + public static void main(String[] args) { + System.out.println("Hello World"); + } }
3	Membandingkan dua commit  git diff 1f7282a..4f0a687	diff --git a/BelajarGit/src/belajar/git/Kalkulator.java b/BelajarGit/src/belajar index 24f9af4..e8ffc2d 100644 --- a/BelajarGit/src/belajar/git/Kalkulator.java +++ b/BelajarGit/src/belajar/git/Kalkulator.java @@ -19,4 +19,8 @@ public class Kalkulator { public int kali(int a, int b){ return a * b; } + public double bagi(double a, double b){ + return a/b; + } + } }

## Kesimpulan

Kita dapat melihat isi repository dengan menggunakan perintah git log dan git diff.

# Membatalkan perubahan

## Penjelasan

Berbagai perintah untuk membatalkan perubahan yang sudah dilakukan

## Panduan

No	Aktifitas	Hasil
1	Edit satu file	
2	Tambahkan ke staging area  git add Halo.java	
3	Batalkan staging  git reset HEAD Halo.java	Unstaged changes after reset: M BelajarGit/src/belajar/git/Halo.java
4	Langsung commit  git commit -am "ganti jadi bahasa indonesia"	[master e2e84fa] ganti jadi bahasa indonesia 1 files changed, 1 insertions(+), 1 deletions(-)
5	Kembalikan ke commit sebelumnya, dengan tetap menyimpan history undo  git revert e2e84f	Finished one revert.  [master 70002d2] Undo perubahan "ganti jadi bahasa indonesia" 1 files changed, 1 insertions(+), 1 deletions(-)
6	Perhatikan git log untuk melihat bahwa ada commit tambahan untuk melakukan undo terhadap commit e2e84f	70002d2 Undo perubahan "ganti jadi bahasa indonesia" e2e84fa ganti jadi bahasa indonesia d6a8455 tambahan fitur pembagian 8a4fbcf implement fitur tambah, kurang, kali 07a38f2 tambah satu file, fitur kalkulator c70c58c commit pertama
7	Kembalikan ke commit sebelumnya, tanpa menyimpan history undo.  git reset --hard d6a845	HEAD is now at d6a8455 tambahan fitur pembagian
8	Perhatikan git log untuk melihat bahwa commit setelah fitur pembagian telah hilang	d6a8455 tambahan fitur pembagian 8a4fbcf implement fitur tambah, kurang, kali 07a38f2 tambah satu file, fitur kalkulator c70c58c commit pertama

## Kesimpulan

Perubahan bisa dilakukan dengan menggunakan perintah reset dan revert. Perintah revert menambahkan satu commit yang membatalkan commit lain yang ingin direvert. Perintah reset memindahkan posisi HEAD ke commit yang dituju.