Repository on server:
`http://server/project`

`hg serve -p port`

`mkdir project`
`hg init project`
Create repository

`sshd`

Repository on server:
`ssh://server/project`

`hg clone repository [project]`

`hg clone repository [project]`

Repository:
`/path/project`

`hg clone project myProj`

Copied Repository:
`/path/myProj`

`hg branch branch`
Name current changeset as a new branch

`hg merge branch`
Merge-in changesets from another branch

`hg rollback`
Revert to before last commit/import/push/pull

`hg incoming [repository]`
List changsets from source

`hg pull [-u] [repository]`
Check-in new changesets from source or other repository (-u = update as well)

`hg log [-r REV[:toREV]] [-p]`
Show changesets (`-p` = with diff)

`hg tip = hg log -r tip`
Show the latest changeset

`hg update [-C] [Rev|Tag]`
Update working dir with pulled changeset (-C = discard uncommitted changes)

update over branch: failed

`hg merge`
Merge change across branch and notify merge conflicts

`hg resolve [-a] [files]`
Try to remerge unresolved files (-a = try all unresolved)

`hg heads [-r Rev]`
Show all branch heads or heads of specified revision

Updated repository:
`/path/project`

Modification

`hg parents [file]`
Show parent revision of current working dir
If file is given, show revision of last change

`hg revert [-r REV] [files]`
Revert changes in file/dir to an earlier state default to the parent revision

`hg copy oldFile newFile`
Make copy of file for modification and mark future changes at old as needed in new file

`hg rename oldFile newFile`
Copy and remove

`hg remove files`

`hg add [files]`

`hg addremove`
Automatic add and remove

New Revision

Modified copy

`hg backout --merge REV`
Create new changeset to reverse the effect of a particular changeset

`hg status`
Check what changed in working dir

`hg diff [-r RevA [-r RevB]] [files]`
Show diff of files between changesets (default current against tip)

`hg commit [-A] [-u 'name <email>'`
Create a changeset in local repository

diff file

`hg export Rev > diff`

`hg import diff`

New Revision

`hg unbundle file`
Apply the changegroup file

changegroup in bzip2

`hg bundle [--base RevA] [-r RevB] [-a] file`
Create a changegroup file from RevA to RevB (-a = all since null)

`hg tag [-r Rev] Tag`
Give a name to a revision (default = parent revision)

`hg tag --remove Tag`
Remove a name assignment

`hg tags`
List all names and their revision

`hg outgoing [repository]`
List changesets to destination

`hg push [-r Rev] [repository]`
Push changsets to destination

Repository on server

mercurial Cheatsheet

Copyright © 2009 Adrian Sai-wah Tam
Licensed under Creative Commons
(cc) BY-SA