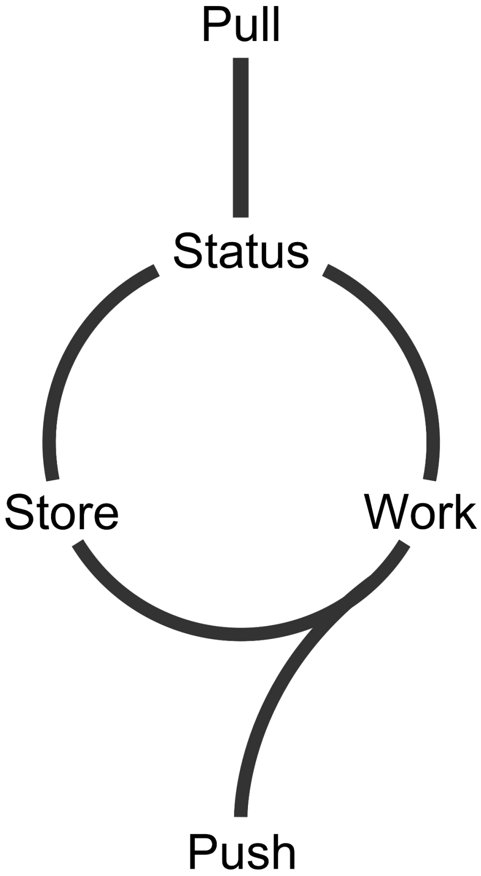
Git for beginners - a visual workflow cheat sheet

Status

pull	{from} {to}	Pull changes {from} remote {to} local branch
status		Current branch, staged & unstaged changes
log		Consult previous commits
branch		Overview existing and current branches

Work

switch	-c {new}	Switch to and (-c)reate {new} branch
diff	[file]	Compare unstaged changes of [file]
restore	[file]	Restore [file] to latest commit (changes lost)
stash | pop		Take out - | bring back - unstaged changes



Git for beginners - a visual workflow cheat sheet

Legends:
[Hello] Path to file or directory named `Hello`
{John} Branch named `john`
(awesome) Commit tag called `awesome`

Workflow example

```
# Start of day
> cd /dir/to/repo
# Update working branch from remote
> git pull origin dev
# New feature = new branch
> git switch -c fix_syntax
...Work on script.py and script.sh
# Test and review changes
> git diff script.py
# Stage and commit changes
> git add script.py script.sh
> git commit
# merge into dev branch
> git switch dev
> git merge fix_syntax
> git branch -d fix_syntax
# Push changes to remote
> git push origin dev
```

Store

add	[file or dir]	Stage changes to [file] or all files in [dir]
reset	[file]	Unstage yet keep, changes to [file]
commit		Store staged changes
reset	--hard	Remove all staged and unstaged changes
push	{to} {from}	Push changes {to} remote {from} local branch

Maintenance

merge	{old}	Merge {old} branch into current branch
branch	-d {old}	Delete {old} branch for cleanup after merge
tag	(tag) #cmt	At a (tag) to latest commit or specific #cmt
rebase	-i #cmt	Clean up in latest or specified commits
commit	--amend	Add staged changes to latest commit