GIT CHEAT SHEET BY LIDSOL

Laboraqtorio de Investigación y Desarrollo de Software Libre

CREATE SSH KEY

```
# Generate a new ssh key
ssh-keygen -o -a 100 -t ed25519 -f /path/to/key -C "your@email.com"

# Change passphrase
ssh-keygen -p -f /path/to/key
```

CREATING A REPOSITORY

```
# Turn a directory into a git repository
git init

# Clone a repository
git clone <url>

# Clone only the latest commit
git clone --depth=1 <url>
```

COMMITTING CHANGES

```
# Commit changes
git commit -m "Message"

# Commit changes with a message on the editor
git commit
```

CHECKING THE STATUS

```
# Check the status of the repository
git status
```

LOGGING

```
# Show the history of the repository
git log

# Show differences between the working directory and the staging area
git diff

# Show differences between commits
git diff <commit> <commit>

# Show differences between the working directory and the last commit
git diff HEAD
```

MOVING AROUND

```
# Go to the last commit
git checkout HEAD

# Go to a specific commit
git checkout <commit>

# Go to the previous commit
git checkout HEAD^

# Go to the next commit
git checkout HEAD@{1}
```

STASHING CHANGES

```
# List branches
git branch -a

# Create a branch
git branch <branch>

# Switch to a branch
git checkout <branch>

# Create a branch and switch to it
git checkout -b <branch>

# Delete a branch
git branch -d <branch>

# Merge a branch
git merge <branch>
```

CONFIGURING GIT

```
# Configure name
git config --global user.name "Your Name"

# Configure email
git config --global user.email "your@email.com"

# Configure editor
git config --global core.editor "vim"

# Configure color
git config --global color.ui auto
```

STAGING FILES

```
# Add a file to the staging area
git add <file>

# Add all files to the staging area
git add .

# Remove a file from the staging area
git reset <file>

# Remove all files from the staging area
git reset
```

CHECKING THE STATUS

```
# Push changes to a remote repository
git push <remote> <branch>

# Force push changes to a remote repository
git push -f <remote> <branch>

# Pull changes from a remote repository
git pull <remote> <branch>

# Fetch changes from a remote repository
git fetch <remote>

# Merge changes from a remote repository
git merge <remote>/<branch>
```

REVERTING CHANGES

```
# Reverse a commit
git revert <commit>

# Reset the repository to a previous commit
git reset <commit>

# Reset the repository to a previous commit and keep the changes
git reset --soft <commit>

# Reset the repository to a previous commit and discard the changes
git reset --hard <commit>
```

STASHING CHANGES

```
# Stash changes
git stash

# Apply stashed changes
git stash apply

# List stashed changes
git stash list

# Drop stashed changes
git stash drop
```