```
Last login: Fri Jun 27 16:47:38 on console
sravan@PURVI ~ % brew install john

zsh: command not found: brew
sravan@PURVI ~ % /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"

==> Checking for `sudo` access (which may request your password)...
Password:
[==> This script will install:
/opt/homebrew/bin/brew
/opt/homebrew/share/doc/homebrew
/opt/homebrew/share/man/man1/brew.1
/opt/homebrew/share/zsh/site-functions/_brew
/opt/homebrew/etc/bash_completion.d/brew
/opt/homebrew
/etc/paths.d/homebrew
==> The following new directories will be created:
/opt/homebrew/bin
/opt/homebrew/etc
/opt/homebrew/include
/opt/homebrew/lib
/opt/homebrew/sbin
/opt/homebrew/share
/opt/homebrew/var
/opt/homebrew/opt
/opt/homebrew/share/zsh
/opt/homebrew/share/zsh/site-functions
/opt/homebrew/var/homebrew
/opt/homebrew/var/homebrew/linked
/opt/homebrew/Cellar
/opt/homebrew/Caskroom
/opt/homebrew/Frameworks
==> The Xcode Command Line Tools will be installed.

Press RETURN/ENTER to continue or any other key to abort:
==> /usr/bin/sudo /usr/bin/install -d -o root -g wheel -m 0755 /opt/homebrew
==> /usr/bin/sudo /bin/mkdir -p /opt/homebrew/bin /opt/homebrew/etc /opt/homebrew/include /opt/homebrew/lib /opt/homebrew/sbin /opt/homebrew/share /opt/homebrew/var /opt/homebrew/opt /opt/homebrew/share/z
sh /opt/homebrew/share/zsh/site-functions /opt/homebrew/var/homebrew /opt/homebrew/var/homebrew/linked /opt/homebrew/Cellar /opt/homebrew/Caskroom /opt/homebrew/Frameworks
==> /usr/bin/sudo /bin/chmod ug=rwx /opt/homebrew/bin /opt/homebrew/etc /opt/homebrew/include /opt/homebrew/lib /opt/homebrew/sbin /opt/homebrew/share /opt/homebrew/var /opt/homebrew/opt /opt/homebrew/sha
re/zsh /opt/homebrew/share/zsh/site-functions /opt/homebrew/var/homebrew /opt/homebrew/var/homebrew/linked /opt/homebrew/Cellar /opt/homebrew/Caskroom /opt/homebrew/Frameworks
==> /usr/bin/sudo /bin/chmod go-w /opt/homebrew/share/zsh /opt/homebrew/share/zsh/site-functions
==> /usr/bin/sudo /usr/sbin/chown sravan /opt/homebrew/bin /opt/homebrew/etc /opt/homebrew/include /opt/homebrew/lib /opt/homebrew/sbin /opt/homebrew/share /opt/homebrew/var /opt/homebrew/opt /opt/homebre
w/share/zsh /opt/homebrew/share/zsh/site-functions /opt/homebrew/var/homebrew /opt/homebrew/var/homebrew/linked /opt/homebrew/Cellar /opt/homebrew/Caskroom /opt/homebrew/Frameworks
==> /usr/bin/sudo /usr/bin/chgrp admin /opt/homebrew/bin /opt/homebrew/etc /opt/homebrew/include /opt/homebrew/lib /opt/homebrew/sbin /opt/homebrew/share /opt/homebrew/var /opt/homebrew/opt /opt/homebrew/
share/zsh /opt/homebrew/share/zsh/site-functions /opt/homebrew/var/homebrew /opt/homebrew/var/homebrew/linked /opt/homebrew/Cellar /opt/homebrew/Caskroom /opt/homebrew/Frameworks
==> /usr/bin/sudo /usr/sbin/chown -R sravan:admin /opt/homebrew
==> /usr/bin/sudo /bin/mkdir -p /Users/sravan/Library/Caches/Homebrew
==> /usr/bin/sudo /bin/chmod g+rwx /Users/sravan/Library/Caches/Homebrew
==> /usr/bin/sudo /usr/sbin/chown -R sravan /Users/sravan/Library/Caches/Homebrew
==> Searching online for the Command Line Tools
==> /usr/bin/sudo /usr/bin/touch /tmp/.com.apple.dt.CommandLineTools.installondemand.in-progress
Didn't get a response from the Apple Software Update server.
==> /usr/bin/sudo /bin/rm -f /tmp/.com.apple.dt.CommandLineTools.installondemand.in-progress
==> Installing the Command Line Tools (expect a GUI popup):
xcode-select: note: install requested for command line developer tools
Press any key when the installation has completed.
==> /usr/bin/sudo /usr/bin/xcode-select --switch /Library/Developer/CommandLineTools
xcode-select: error: invalid developer directory '/Library/Developer/CommandLineTools'
Failed during: /usr/bin/sudo /usr/bin/xcode-select --switch /Library/Developer/CommandLineTools
sravan@PURVI ~ % xcode-select --install

xcode-select: note: install requested for command line developer tools
sravan@PURVI ~ % xcode-select -p
```

```
sravan@PURVI ~ % xcode-select --install

xcode-select: note: install requested for command line developer tools
sravan@PURVI ~ % xcode-select -p

/Library/Developer/CommandLineTools
sravan@PURVI ~ % /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"

==> Checking for `sudo` access (which may request your password)...
Password:
==> This script will install:
/opt/homebrew/bin/brew
/opt/homebrew/share/doc/homebrew
/opt/homebrew/share/man/man1/brew.1
/opt/homebrew/share/zsh/site-functions/_brew
/opt/homebrew/etc/bash_completion.d/brew
/opt/homebrew
/etc/paths.d/homebrew

Press RETURN/ENTER to continue or any other key to abort:
==> /usr/bin/sudo /usr/sbin/chown -R sravan:admin /opt/homebrew
==> Downloading and installing Homebrew...
remote: Enumerating objects: 304650, done.
remote: Counting objects: 100% (140/140), done.
remote: Compressing objects: 100% (84/84), done.
remote: Total 304650 (delta 78), reused 101 (delta 56), pack-reused 304510 (from 4)
remote: Enumerating objects: 55, done.
remote: Counting objects: 100% (33/33), done.
remote: Total 55 (delta 33), reused 33 (delta 33), pack-reused 22 (from 1)
==> Updating Homebrew...
==> Downloading https://ghcr.io/v2/homebrew/portable-ruby/portable-ruby/blobs/sha256:fd162df7a06190ee800a9e6afd28f4466d33548821a480ba043cd927b44d60f7
######################################################################### 100.0%
==> Pouring portable-ruby-3.4.4.arm64_big_sur.bottle.tar.gz
==> /usr/bin/sudo /bin/mkdir -p /etc/paths.d
Password:
==> /usr/bin/sudo tee /etc/paths.d/homebrew
/opt/homebrew/bin
==> /usr/bin/sudo /usr/sbin/chown root:wheel /etc/paths.d/homebrew
==> Installation successful!

==> Homebrew has enabled anonymous aggregate formulae and cask analytics.
Read the analytics documentation (and how to opt-out) here:
  https://docs.brew.sh/Analytics
No analytics data has been sent yet (nor will any be during this install run).

==> Homebrew is run entirely by unpaid volunteers. Please consider donating:
  https://github.com/Homebrew/brew#donations

==> Next steps:
- Run these commands in your terminal to add Homebrew to your PATH:
    echo >> /Users/sravan/.zprofile
    echo 'eval "$(/opt/homebrew/bin/brew shellenv)"' >> /Users/sravan/.zprofile
    eval "$(/opt/homebrew/bin/brew shellenv)"
- Run brew help to get started
- Further documentation:
    https://docs.brew.sh

sravan@PURVI ~ % echo 'eval "$(/opt/homebrew/bin/brew shellenv)"' >> ~/.zprofile
eval "$(/opt/homebrew/bin/brew shellenv)"

sravan@PURVI ~ % brew install john

==> Downloading https://ghcr.io/v2/homebrew/core/john/manifests/1.9.0_1-1
```

```
################################################################## 100.0%
==> Fetching john
==> Downloading https://ghcr.io/v2/homebrew/core/john/blobs/sha256:838f0767e0ffb
################################################################## 100.0%
==> Pouring john--1.9.0_1.arm64_sequoia.bottle.1.tar.gz
🍺 /opt/homebrew/Cellar/john/1.9.0_1: 36 files, 19.6MB
==> Running `brew cleanup john`...
Disable this behaviour by setting HOMEBREW_NO_INSTALL_CLEANUP.
Hide these hints with HOMEBREW_NO_ENV_HINTS (see `man brew`).
sravan@PURVI ~ % john --version

Created directory: /Users/sravan/.john
Unknown option: "--version"
sravan@PURVI ~ % john

John the Ripper password cracker, version 1.9.0
Copyright (c) 1996-2019 by Solar Designer
Homepage: http://www.openwall.com/john/

Usage: john [OPTIONS] [PASSWORD-FILES]
--single                  "single crack" mode
--wordlist=FILE --stdin   wordlist mode, read words from FILE or stdin
--rules                   enable word mangling rules for wordlist mode
--incremental[=MODE]      "incremental" mode [using section MODE]
--external=MODE           external mode or word filter
--stdout[=LENGTH]         just output candidate passwords [cut at LENGTH]
--restore[=NAME]          restore an interrupted session [called NAME]
--session=NAME            give a new session the NAME
--status[=NAME]           print status of a session [called NAME]
--make-charset=FILE       make a charset, FILE will be overwritten
--show                    show cracked passwords
--test[=TIME]             run tests and benchmarks for TIME seconds each
--users=[-]LOGIN|UID[,..] [do not] load this (these) user(s) only
--groups=[-]GID[,..]      load users [not] of this (these) group(s) only
--shells=[-]SHELL[,..]    load users with[out] this (these) shell(s) only
--salts=[-]N              load salts with[out] at least N passwords only
--save-memory=LEVEL       enable memory saving, at LEVEL 1..3
--node=MIN[-MAX]/TOTAL    this node's number range out of TOTAL count
--fork=N                  fork N processes
--format=NAME             force hash type NAME: descrypt/bsdicrypt/md5crypt/
                          bcrypt/LM/AFS/tripcode/dummy
sravan@PURVI ~ % echo 'purvi:$(openssl passwd -1 password123)' > myhash.txt

sravan@PURVI ~ % cat myhash.txt

purvi:$(openssl passwd -1 password123)
sravan@PURVI ~ % echo "purvi:$(openssl passwd -1 password123)" > myhash.txt

sravan@PURVI ~ % cat myhash.txt

purvi:$1$k6igytd.$jmaUsCPvjVlcVJgBmGk.2/
sravan@PURVI ~ % john myhash.txt

Loaded 1 password hash (md5crypt [MD5 32/64 X2])
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:00:25 3/3 0g/s 10805p/s 10805c/s 10805C/s shikar..shikie
0g 0:00:01:04 3/3 0g/s 10991p/s 10991c/s 10991C/s brybill..brybile
0g 0:00:01:04 3/3 0g/s 10991p/s 10991c/s 10991C/s brybill..brybile
Session aborted
sravan@PURVI ~ % echo "password123" > mywordlist.txt

sravan@PURVI ~ % john --wordlist=mywordlist.txt myhash.txt
```

```
Unknown option: "--version"
sravan@PURVI ~ % john

John the Ripper password cracker, version 1.9.0
Copyright (c) 1996-2019 by Solar Designer
Homepage: http://www.openwall.com/john/

Usage: john [OPTIONS] [PASSWORD-FILES]
--single                 "single crack" mode
--wordlist=FILE --stdin   wordlist mode, read words from FILE or stdin
--rules                  enable word mangling rules for wordlist mode
--incremental[=MODE]     "incremental" mode [using section MODE]
--external=MODE          external mode or word filter
--stdout[=LENGTH]        just output candidate passwords [cut at LENGTH]
--restore[=NAME]         restore an interrupted session [called NAME]
--session=NAME           give a new session the NAME
--status[=NAME]          print status of a session [called NAME]
--make-charset=FILE      make a charset, FILE will be overwritten
--show                   show cracked passwords
--test[=TIME]            run tests and benchmarks for TIME seconds each
--users=[-]LOGIN|UID[,..] [do not] load this (these) user(s) only
--groups=[-]GID[,..]     load users [not] of this (these) group(s) only
--shells=[-]SHELL[,..]   load users with[out] this (these) shell(s) only
--salts=[-]N             load salts with[out] at least N passwords only
--save-memory=LEVEL      enable memory saving, at LEVEL 1..3
--node=MIN[-MAX]/TOTAL   this node's number range out of TOTAL count
--fork=N                 fork N processes
--format=NAME            force hash type NAME: descrypt/bsdicrypt/md5crypt/
                         bcrypt/LM/AFS/tripcode/dummy
sravan@PURVI ~ % echo 'purvi:$(openssl passwd -1 password123)' > myhash.txt

sravan@PURVI ~ % cat myhash.txt

purvi:$(openssl passwd -1 password123)
sravan@PURVI ~ % echo "purvi:$(openssl passwd -1 password123)" > myhash.txt

sravan@PURVI ~ % cat myhash.txt

purvi:$1$k6igytd.$jmaUsCPvjVlcVJgBmGk.2/
sravan@PURVI ~ % john myhash.txt

Loaded 1 password hash (md5crypt [MD5 32/64 X2])
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:00:25 3/3 0g/s 10805p/s 10805c/s 10805C/s shikar..shikie
0g 0:00:01:04 3/3 0g/s 10991p/s 10991c/s 10991C/s brybill..brybile
0g 0:00:01:04 3/3 0g/s 10991p/s 10991c/s 10991C/s brybill..brybile
Session aborted
sravan@PURVI ~ % echo "password123" > mywordlist.txt

sravan@PURVI ~ % john --wordlist=mywordlist.txt myhash.txt

Loaded 1 password hash (md5crypt [MD5 32/64 X2])
Press 'q' or Ctrl-C to abort, almost any other key for status
password123      (purvi)
1g 0:00:00:00 100% 100.0g/s 100.0p/s 100.0c/s 100.0C/s password123
Use the "--show" option to display all of the cracked passwords reliably
Session completed
sravan@PURVI ~ % john --show myhash.txt

purvi:password123

1 password hash cracked, 0 left
sravan@PURVI ~ %
```