# tldr pages book

Simplified and community-driven man pages

*Generated on Thu Oct 9 06:40:28 2025*

Website: https://tldr.sh

GitHub: https://github.com/tldr-pages/tldr

# Android

# am

**مدير الأنشطة في أندرويد.**

**لمزيد من التفاصيل**: https://developer.android.com/tools/adb#am.

- ابدأ نشاطا بتحديد اسم الحزمة/المكوّن بالكامل:

  ```
  am start -n {{com.android.settings/.Settings}}
  ```

- ابدأ فعلا (action) ومرّر بيانات (data) له:

  ```
  am start -a {{android.intent.action.VIEW}} -d {{tel:123}}
  ```

- ابدأ نشاطا يطابق فعلا (action) وفئة (category) معيّنة:

  ```
  am start -a {{android.intent.action.MAIN}} -c
  {{android.intent.category.HOME}}
  ```

- حوّل Intent إلى URI:

  ```
  am to-uri -a {{android.intent.action.VIEW}} -d {{tel:123}}
  ```

# Common

# $

**توسِّع متغيرات** Bash.

**لمزيد من التفاصيل**: https://gnu.org/software/bash/manual/bash.html#Shell-Variables.

- طباعة قيمة متغير:

  ```
  echo ${{VARIABLE}}
  ```

- طباعة حالة الخروج للأمر السابق:

  ```
  echo $?
  ```

- طباعة رقم عشوائي بين 0 و 32767:

  ```
  echo $RANDOM
  ```

- طباعة أحد سلاسل المطالبة (Prompt Strings):

  ```
  echo ${{PS0|PS1|PS2|PS3|PS4}}
  ```

- تنفيذ الأمر بعد توسيعه، وهو نفس تشغيله باستخدام backticks:

  ```
  $({{command}})
  ```

- عرض عدد الوسائط في السياق الحالي:

  ```
  echo $#
  ```

- طباعة جميع عناصر مصفوفة Bash:

  ```
  echo ${array[@]}
  ```

# %

**إدارة الوظائف** (Jobs).

**لمزيد من التفاصيل**: https://www.gnu.org/software/bash/manual/bash.html#Job-Control-Basics.

- استدعاء الوظيفة الحالية إلى المقدمة:

  ```
  %
  ```

- استدعاء الوظيفة السابقة إلى المقدمة:

  ```
  %-
  ```

- استدعاء الوظيفة ذات الرقم `N` إلى المقدمة:

  ```
  %{{N}}
  ```

- استدعاء الوظيفة التي يبدأ أمرها بـ `string` إلى المقدمة:

  ```
  %{{string}}
  ```

- استدعاء الوظيفة التي يحتوي أمرها على `string` إلى المقدمة:

  ```
  %?{{string}}
  ```

- استئناف وظيفة معلقة:

  ```
  %{{1}} &
  ```

# 7z

**أداة أرشفة الملفات بنسبة ضغط عالية.**

**لمزيد من التفاصيل**: https://manned.org/7z.

- إضافة ملف أو مجلد إلى أرشيف جديد أو موجود مسبقًا[a]:

```
7z a {{path/to/archive.7z}} {{path/to/file_or_directory}}
```

- تشفير أرشيف موجود (بما في ذلك أسماء الملفات):

```
7z a {{path/to/encrypted.7z}} -p{{password}} -mhe=on {{path/to/archive.7z}}
```

- ضغط أرشيف مع الحفاظ على هيكل المجلد الأصلي[x] فك:

```
7z x {{path/to/archive.7z}}
```

- ضغط أرشيف إلى مجلد محدد[x] فك:

```
7z x {{path/to/archive.7z}} -o{{path/to/output}}
```

- ضغط أرشيف إلى[x] فك stdout:

```
7z x {{path/to/archive.7z}} -so
```

- إنشاء أرشيف باستخدام نوع ضغط محدد:

```
7z a -t{{7z|bzip2|gzip|lzip|tar|zip}} {{path/to/archive}} {{path/to/file_or_directory}}
```

- عرض[l] محتويات أرشيف:

```
7z l {{path/to/archive.7z}}
```

- تحديد مستوى الضغط (مستوى أعلى يعني ضغط أكثر، ولكنه أبطأ):

```
7z a {{path/to/archive.7z}} -mx={{0|1|3|5|7|9}} {{path/to/file_or_directory}}
```

# arp

**عرض وإدارة ذاكرة التخزين المؤقت لـ** ARP **في النظام**.

**لمزيد من التفاصيل**: https://manned.org/arp.8.

- عرض جدول ARP الحالي:

  ```
  arp -a
  ```

- حذف إدخال معين:

  ```
  arp -d {{address}}
  ```

- إضافة إدخال جديد إلى جدول ARP:

  ```
  arp -s {{address}} {{mac_address}}
  ```

# cat

**طباعة وسلسلة الملفات.**

**لمزيد من التفاصيل**: https://manned.org/cat.1posix.

- طباعة محتوي ملف إلى `stdout`:

  ```
  cat {{path/to/file}}
  ```

- دمج عدة ملفات في ملف إخراج:

  ```
  cat {{path/to/file1 path/to/file2 ...}} > {{path/to/output_file}}
  ```

- إلحاق عدة ملفات بملف إخراج:

  ```
  cat {{path/to/file1 path/to/file2 ...}} >> {{path/to/output_file}}
  ```

- نسخ محتويات ملف إلى ملف إخراج دون استخدام الذاكرة المؤقتة:

  ```
  cat -u {{/dev/tty12}} > {{/dev/tty13}}
  ```

- كتابة `stdin` إلى ملف:

  ```
  cat - > {{path/to/file}}
  ```

# cd

**تغيير مجلد العمل الحالي**.

**لمزيد من التفاصيل**: https://www.gnu.org/software/bash/manual/bash.html#index-cd

- اللانتقال إلى المجلد المذكور:

```
cd {{path/to/directory}}
```

- اللانتقال إلى المجلد الأعلى للمجلد الحالي:

```
cd ..
```

- الانتقال إلى المجلد الرئيسي للمستخدم الحالي:

```
cd
```

- الانتقال إلى المجلد الرئيسي للمستخدم المذكور:

```
cd ~{{username}}
```

- الانتقال إلى المجلد الذي تم اختياره سابقًا:

```
cd -
```

- الانتقال إلى مجلد الجذر:

```
cd /
```

# chisel

**إنشاء أنفاق** TCP/UDP**، يتم نقلها عبر** HTTP**، وتأمينها عبر** SSH.

**يتضمن كل من العميل والخادم في نفس الملف التنفيذي** `chisel`.

**لمزيد من التفاصيل**: https://github.com/jpillora/chisel.

- تشغيل خادم Chisel:

```
chisel server
```

- تشغيل خادم Chisel يستمع إلى منفذ محدد:

```
chisel server {{[-p|--port]}} {{server_port}}
```

- تشغيل خادم Chisel يقبل الاتصالات المصادق عليها باستخدام اسم المستخدم وكلمة المرور:

```
chisel server --auth {{username}}:{{password}}
```

- الاتصال بخادم Chisel ومنفذ وإنشاء نفق لمنفذ محدد إلى خادم طرف ثالث:

```
chisel client {{server_ip}}:{{server_port}} {{local_port}}:
{{remote_server}}:{{remote_port}}
```

- الاتصال بخادم Chisel ومنفذ وإنشاء نفق لجهاز ومنفذ محددين إلى خادم طرف ثالث:

```
chisel client {{server_ip}}:{{server_port}} {{local_host}}:
{{local_port}}:{{remote_server}}:{{remote_port}}
```

- الاتصال بخادم Chisel باستخدام المصادقة باسم المستخدم وكلمة المرور:

```
chisel client --auth {{username}}:{{password}} {{server_ip}}:
{{server_port}} {{local_port}}:{{remote_server}}:
{{remote_port}}
```

- تهيئة خادم Chisel وكيل SOCKS5 في الوضع العكسي على منفذ محدد، مع تمكين وظيفة (على المنفذ 1080):

```
chisel server {{[-p|--port]}} {{server_port}} --reverse --
socks5
```

- الاتصال بخادم Chisel على عنوان IP ومنفذ محددين، وإنشاء نفق عكسي يتم تعيينه إلى وكيل SOCKS محلي:

```
chisel client {{server_ip}}:{{server_port}} R:socks
```

# cut

**استخراج حقول من `stdin` أو من الملفات.**

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/cut-invocation.html.

- حقل محدد من كل سطر[f]/حرف[c] طباعة نطاق:

```
{{command}} | cut --{{characters|fields}} {{1|1,10|1-10|1-|-10}}
```

- فاصل معين[d] حقل محدد من كل سطر باستخدام[f] طباعة نطاق:

```
{{command}} | cut --delimiter "{{,}}" --fields {{1}}
```

- حرف معين من كل سطر في ملف محدد[c] طباعة نطاق:

```
cut --characters {{1}} {{path/to/file}}
```

- بدلاً (`find . -print0` مثل الناتج من) `NUL` حقول محددة من أسطر منتهية بـ[f] طباعة من أسطر جديدة:

```
{{command}} | cut --zero-terminated --fields {{1}}
```

# dhclient

**عميل** DHCP.

**لمزيد من التفاصيل**: https://manned.org/dhclient.

- الحصول على عنوان IP لواجهة `eth0`:

  ```
  sudo dhclient {{eth0}}
  ```

- تحرير عنوان IP لواجهة `eth0`:

  ```
  sudo dhclient -r {{eth0}}
  ```

# doppler

**إدارة المتغيرات البيئية عبر بيئات مختلفة باستخدام** Doppler.

**تحتوي على وثائق استخدام خاصة بها `secrets` و `run` بعض الأوامر الفرعية مثل**.

**لمزيد من التفاصيل**: https://docs.doppler.com/docs/cli.

- إعداد Doppler CLI في الدليل الحالي:

  ```
  doppler setup
  ```

- إعداد مشروع Doppler والتكوين في الدليل الحالي:

  ```
  doppler setup
  ```

- تشغيل أمر مع حقن الأسرار في البيئة:

  ```
  doppler run --command {{command}}
  ```

- عرض قائمة المشاريع الخاصة بك:

  ```
  doppler projects
  ```

- عرض الأسرار للمشروع الحالي:

  ```
  doppler secrets
  ```

- فتح لوحة تحكم Doppler في المتصفح:

  ```
  doppler open
  ```

# du

**استخدام القرص: تقدير وتلخيص استخدام مساحة الملفات والمُجَلَّدات.**

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/du-invocation.html.

- عرض أحجام مُجَلَّد معين وجميع المُجَلَّدات الفرعية، بوحدة معينة (B/KiB/MiB):

```
du -{{b|k|m}} {{path/to/directory}}
```

- عرض أحجام مُجَلَّد معين وجميع المُجَلَّدات الفرعية بصيغة قابلة للقراءة (أي اختيار الوحدة المناسبة تلقائيًا لكل حجم):

```
du {{[-h|--human-readable]}} {{path/to/directory}}
```

- عرض حجم مُجَلَّد معين فقط، بوحدات قابلة للقراءة:

```
du {{[-sh|--summarize --human-readable]}} {{path/to/directory}}
```

- عرض الأحجام القابلة للقراءة لمُجَلَّد معين وجميع الملفات والمُجَلَّدات داخله:

```
du {{[-ah|--all --human-readable]}} {{path/to/directory}}
```

- عرض الأحجام القابلة للقراءة لمُجَلَّد معين والمُجَلَّدات الفرعية حتى عمق N مستويات:

```
du {{[-h|--human-readable]}} {{[-d|--max-depth]}} N {{path/to/directory}}
```

- عرض الحجم القابل للقراءة لجميع ملفات `.jpg` في المُجَلَّد الحالي، مع إظهار المجموع الكلي في النهاية:

```
du {{[-ch|--total --human-readable]}} {{./*.jpg}}
```

- عرض جميع الملفات والمُجَلَّدات (بما في ذلك المخفية) التي تتجاوز حدًا معينًا من الحجم (مفيد لمعرفة ما يستهلك المساحة):

```
du {{[-ah|--all --human-readable]}} {{[-t|--threshold]}} {{1G|1024M|1048576K}} .[^.]* *
```

# fastmod

**أداة للاستبدال الجزئي للنصوص في قاعدة الأكواد لديك.**

**التعبيرات النمطية يعالجها قفص من بضاعة رست وهو** regex.

**لمزيد من التفاصيل**: https://github.com/facebookincubator/fastmod.

- استبدال بالتعبيرات النمطية في كل ملفات المسار الحالي وأبنائه في الملفات غير المُتجاهلة بـ .gitignore أو .ignore:

```
fastmod {{regex_pattern}} {{replacement}}
```

- استبدال متجاهلا حالة الحرف في ملف أو في ملفات مسار:

```
fastmod --ignore-case {{regex_pattern}} {{replacement}} --
{{path/to/file path/to/directory ...}}
```

- استبدال بالتعبيرات النمطية مع تحديد المكان الذي يُستبدل فيه:

```
fastmod {{regex}} {{replacement}} --dir {{path/to/directory}}
--iglob {{'**/*.{js,json}'}}
```

- استبدال بالنص مُطابقةً (وليس التعبيرات النمطية)، في ملفات امتداداتهم إما JSON أو js فحسب:

```
fastmod --fixed-strings {{exact_string}} {{replacement}} --
extensions {{json,js}}
```

- استبدال بجميع النصوص مُطابقةً، مباشرة دون مِحَثِّ تأكيد (prompt):

```
fastmod --accept-all --fixed-strings {{exact_string}}
{{replacement}}
```

- استبدال بجميع النصوص مُطابقةً، مباشرة دون تأكيد، مع طباعة الملفات المُستبدل فيها:

```
fastmod --accept-all --print-changed-files --fixed-strings
{{exact_string}} {{replacement}}
```

# ffmpeg

**أداة لتحويل الفيديو.**

**لمزيد من التفاصيل**: https://ffmpeg.org.

- استخراج الصوت من فيديو وحفظه بصيغة MP3:

```
ffmpeg -i {{path/to/video.mp4}} -vn {{path/to/sound.mp3}}
```

- تحويل ملف FLAC إلى تنسيق Red Book CD (44100kHz، 16bit):

```
ffmpeg -i {{path/to/input_audio.flac}} -ar 44100 -sample_fmt
s16 {{path/to/output_audio.wav}}
```

- حفظ فيديو كـ GIF 15 إلى 1000 بكسل ومعدل الإطارات مع ضبط ارتفاع الصورة إلى:

```
ffmpeg -i {{path/to/video.mp4}} {{[-vf|-filter:v]}}
'scale=-1:{{1000}}' -r {{15}} {{path/to/output.gif}}
```

- دمج صور مرقمة (`frame_1.jpg`، `frame_2.jpg`، في فيديو أو إلخ) GIF:

```
ffmpeg -i {{path/to/frame_%d.jpg}} -f image2 {{video.mpg|
video.gif}}
```

- قص فيديو من وقت بداية معين mm:ss إلى وقت نهاية mm2:ss2 (تجاهل علامة to- للقص حتى النهاية):

```
ffmpeg -i {{path/to/input_video.mp4}} -ss {{mm:ss}} -to
{{mm2:ss2}} {{[-c|-codec]}} copy {{path/to/output_video.mp4}}
```

- تحويل فيديو AVI إلى MP4. صوت AAC 128 بمعدلkbit، فيديو h264 بـ CRF 23:

```
ffmpeg -i {{path/to/input_video}}.avi {{[-c|-codec]}}:a aac -
b:a 128k {{[-c|-codec]}}:v libx264 -crf 23 {{path/to/
output_video}}.mp4
```

- إعادة تهيئة فيديو MKV إلى MP4 دون إعادة ترميز تيارات الصوت أو الفيديو:

```
ffmpeg -i {{path/to/input_video}}.mkv {{[-c|-codec]}} copy
{{path/to/output_video}}.mp4
```

- تحويل فيديو MP4 إلى ترميز VP9. استخدم قيمة CRF (النطاق للحصول على أفضل جودة، الموصى به 15-35) ويجب أن يكون b:v- صفرًا:

```
ffmpeg -i {{path/to/input_video}}.mp4 {{[-c|-codec]}}:v
libvpx-vp9 -crf {{30}} -b:v 0 {{[-c|-codec]}}:a libopus -vbr
on -threads {{number_of_threads}} {{path/to/
output_video}}.webm
```

# find

**البحث عن الملفات أو المُجَلَّدات داخل فروع مُجَلَّد، بشكل متكرر.**

**لمزيد من التفاصيل**: https://manned.org/find.

- البحث عن الملفات حسب الامتداد:

```
find {{root_path}} -name '{{*.ext}}'
```

- البحث عن الملفات المطابقة لأنماط مسار/اسم متعددة:

```
find {{root_path}} -path '{{*/path/*/*.ext}}' -or -name
'{{*pattern*}}'
```

- البحث عن المُجَلَّدات المطابقة لاسم معين، مع تجاهل حالة الأحرف سواء أكانت صغيرة او كبيرة:

```
find {{root_path}} -type d -iname '{{*lib*}}'
```

- البحث عن الملفات المطابقة لنمط معين، مع استثناء مسارات محددة:

```
find {{root_path}} -name '{{*.py}}' -not -path '{{*/site-
packages/*}}'
```

- البحث عن الملفات التي تطابق نطاق حجم معين، مع تقييد العمق التكراري إلى "1":

```
find {{root_path}} -maxdepth 1 -size {{+500k}} -size {{-10M}}
```

- تنفيذ أمر لكل ملف (استخدم {} داخل الأمر للوصول إلى اسم الملف):

```
find {{root_path}} -name '{{*.ext}}' -exec {{wc -l}} {} \;
```

- البحث عن جميع الملفات المعدلة اليوم وتمرير النتائج إلى أمر واحد كوسيطات:

```
find {{root_path}} -daystart -mtime {{-1}} -exec {{tar -cvf
archive.tar}} {} \+
```

- البحث عن الملفات أو المُجَلَّدات الفارغة وحذفها مع عرض التفاصيل:

```
find {{root_path}} -type {{f|d}} -empty -delete -print
```

# fossil ci

**هذا الأمر هو اسم مستعار لـ `fossil commit`.**

- إعرض التوثيقات للأمر الأصلي:

  ```
  tldr fossil commit
  ```

# fossil forget

**هذا الأمر هو اسم مستعار لـ `fossil rm`.**

**لمزيد من التفاصيل**: https://fossil-scm.org/home/help/forget.

- إعرض التوثيقات للأمر الأصلي:

  ```
  tldr fossil rm
  ```

# fossil new

**هذا الأمر هو اسم مستعار لـ `fossil init`.**

- إعرض التوثيقات للأمر الأصلي:

  ```
  tldr fossil init
  ```

# fossil rm

**هذا الأمر هو اسم مستعار لـ `fossil delete`.**

- إعرض التوثيقات للأمر الأصلي:

  ```
  tldr fossil delete
  ```

# ftp

**أدوات للتفاعل مع الخادم عبر بروتوكول نقل الملفات** (FTP).

**لمزيد من التفاصيل**: https://manned.org/ftp.

- الاتصال بخادم FTP:

  ```
  ftp {{ftp.example.com}}
  ```

- الاتصال بخادم FTP مع تحديد عنوان الـ IP والمنفذ:

  ```
  ftp {{ip_address}} {{port}}
  ```

- التبديل إلى وضع النقل الثنائي (الرسوميات، الملفات المضغوطة، إلخ):

  ```
  binary
  ```

- نقل عدة ملفات دون طلب تأكيد على كل ملف:

  ```
  prompt off
  ```

- تنزيل عدة ملفات (تعليمات الكلمة العامة glob):

  ```
  mget {{*.png}}
  ```

- رفع عدة ملفات (تعليمات الكلمة العامة glob):

  ```
  mput {{*.zip}}
  ```

- حذف عدة ملفات على الخادم:

  ```
  mdelete {{*.txt}}
  ```

- إعادة تسمية ملف على الخادم:

  ```
  rename {{original_filename}} {{new_filename}}
  ```

# fzf

**أداة بحث تقريبي** (fuzzy) **لسطر الأوامر**.

**مشابهة لـ** `sk`.

**لمزيد من التفاصيل**: https://github.com/junegunn/fzf#usage.

- تشغيل `fzf` على جميع الملفات داخل المجلد المحدد:

  ```
  find {{path/to/directory}} -type f | fzf
  ```

- تشغيل `fzf` للعمليات الجارية:

  ```
  ps aux | fzf
  ```

- تحديد ملفات متعددة باستخدام `<Shift Tab>` وكتابتها إلى ملف:

  ```
  find {{path/to/directory}} -type f | fzf {{[-m|--multi]}} >
  {{path/to/file}}
  ```

- تشغيل `fzf` مع نص بحثي محدد:

  ```
  fzf {{[-q|--query]}} "{{query}}"
  ```

- تشغيل `fzf` على إدخالات تبدأ بـ `core` وتنتهي بـ `go` أو `rb` أو `py`:

  ```
  fzf {{[-q|--query]}} "^core go$ | rb$ | py$"
  ```

- تشغيل `fzf` على إدخالات لا تطابق `pyc` وتحتوي على `travis`:

  ```
  fzf {{[-q|--query]}} '!pyc travis'
  ```

# g++

**ترجمة ملفات مصدر** C++.

**جزء من** GCC (**مجموعة مترجمات جنو**).

**لمزيد من التفاصيل**: https://gcc.gnu.org.

- ترجمة ملف مصدر إلى ملف تنفيذي ثنائي (Binary):

```
g++ {{path/to/source1.cpp path/to/source2.cpp ...}} {{-o|--output}} {{path/to/output_executable}}
```

- تفعيل عرض جميع الأخطاء والتحذيرات:

```
g++ {{path/to/source.cpp}} -Wall {{-o|--output}} {{output_executable}}
```

- عرض التحذيرات الشائعة، وإضافة رموز التصحيح إلى الإخراج، وتحسين الأداء دون التأثير على التصحيح:

```
g++ {{path/to/source.cpp}} -Wall {{-g|--debug}} -Og {{-o|--output}} {{path/to/output_executable}}
```

- اختيار معيار لغة C++ للترجمة (C++98/C++11/C++14/C++17):

```
g++ {{path/to/source.cpp}} -std={{c++98|c++11|c++14|c++17}} {{-o|--output}} {{path/to/output_executable}}
```

- تضمين مكتبات تقع في مسار مختلف عن ملف المصدر:

```
g++ {{path/to/source.cpp}} {{-o|--output}} {{path/to/output_executable}} -I{{path/to/header}} -L{{path/to/library}} -l{{library_name}}
```

- ترجمة وربط ملفات مصدر متعددة في ملف تنفيذي ثنائي (Binary):

```
g++ {{-c|--compile}} {{path/to/source1.cpp path/to/source2.cpp ...}} && g++ {{-o|--output}} {{path/to/output_executable}} {{path/to/source1.o path/to/source2.o ...}}
```

- تحسين البرنامج المترجم لزيادة الأداء:

```
g++ {{path/to/source.cpp}} -O{{1|2|3|fast}} {{-o|--output}} {{path/to/output_executable}}
```

- عرض الإصدار:

```
g++ --version
```

# gcc

**معالجة مسبقة وتجميع ملفات مصدر** C و C++، **ثم تجميعها وربطها معًا**.

**جزء من** GCC (**مجموعة مترجمات جنو**).

**لمزيد من التفاصيل**: https://gcc.gnu.org.

- ترجمة ملفات مصدر متعددة إلى ملف قابل للتنفيذ:

```
gcc {{path/to/source1.c path/to/source2.c ...}} {{-o|--output}} {{path/to/output_executable}}
```

- تفعيل عرض جميع الأخطاء والتحذيرات:

```
gcc {{path/to/source.c}} -Wall {{-o|--output}} {{output_executable}}
```

- عرض التحذيرات الشائعة، وإضافة رموز التصحيح إلى الإخراج، وتحسين الأداء دون التأثير على التصحيح:

```
gcc {{path/to/source.c}} -Wall {{-g|--debug}} -Og {{-o|--output}} {{path/to/output_executable}}
```

- تضمين مكتبات من مسار مختلف:

```
gcc {{path/to/source.c}} {{-o|--output}} {{path/to/output_executable}} -I{{path/to/header}} -L{{path/to/library}} -l{{library_name}}
```

- ترجمة الكود المصدري إلى تعليمات لغة التجميع:

```
gcc {{-S|--assemble}} {{path/to/source.c}}
```

- ترجمة الكود المصدري إلى ملف كائن دون ربط:

```
gcc {{-c|--compile}} {{path/to/source.c}}
```

- تحسين البرنامج المترجم لزيادة الأداء:

```
gcc {{path/to/source.c}} -O{{1|2|3|fast}} {{-o|--output}} {{path/to/output_executable}}
```

- عرض الإصدار:

```
gcc --version
```

# gh cs

**هذا الأمر هو اسم مستعار لـ `gh codespace`.**

- إعرض التوثيقات للأمر الأصلي:

  ```
  tldr gh codespace
  ```

# git add

**يضيف الملفات المعدلة إلى منطقة التجميع** (staging area).

**لمزيد من التفاصيل**: https://git-scm.com/docs/git-add.

- إضافة ملف إلى منطقة التجميع (staging area):

  ```
  git add {{path/to/file}}
  ```

- إضافة جميع الملفات (المتتبَّعة وغير المتتبَّعة Tracked & Untracked):

  ```
  git add {{[-A|--all]}}
  ```

- إضافة جميع الملفات بشكل متكرر (recursively) بدءًا من المُجَلَّد الحالي:

  ```
  git add .
  ```

- إضافة الملفات المتتبَّعة (Tracked) فقط:

  ```
  git add {{[-u|--update]}}
  ```

- إضافة الملفات المتجاهلة (Ignored) أيضًا:

  ```
  git add {{[-f|--force]}}
  ```

- إضافة أجزاء من الملفات بشكل تفاعلي (Interactive):

  ```
  git add {{[-p|--patch]}}
  ```

- إضافة أجزاء من ملف معين بشكل تفاعلي:

  ```
  git add {{[-p|--patch]}} {{path/to/file}}
  ```

- إضافة ملف بشكل تفاعلي:

  ```
  git add {{[-i|--interactive]}}
  ```

# git fetch

**تنزيل الكائنات والمراجع من مستودع خارجي.**

**لمزيد من التفاصيل**: https://git-scm.com/docs/git-fetch.

- سحب آخر التعديلات من المستودع الخارجي الافتراضي (إذا كان مضبوطًا):

  ```
  git fetch
  ```

- سحب الفروع الجديدة من مستودع خارجي معين:

  ```
  git fetch {{remote_name}}
  ```

- سحب آخر التعديلات من جميع المستودعات الخارجية:

  ```
  git fetch --all
  ```

- سحب العلامات (tags) أيضًا من المستودع الخارجي:

  ```
  git fetch {{[-t|--tags]}}
  ```

- حذف المراجع المحلية للفروع الخارجية التي تم حذفها في المستودع الخارجي:

  ```
  git fetch {{[-p|--prune]}}
  ```

# git init

**يهيئ مستودع** Git **محلي جديد**.

**لمزيد من التفاصيل**: https://git-scm.com/docs/git-init.

- تهيئة مستودع محلي جديد:

  ```
  git init
  ```

- تهيئة مستودع مع تحديد اسم الفرع الابتدائي:

  ```
  git init {{[-b|--initial-branch]}} {{branch_name}}
  ```

- تهيئة مستودع يستخدم خوارزمية SHA256 (يتطلب إصدار Git 2.29 لتنسيق تجزئة الكائنات أو أحدث):

  ```
  git init --object-format sha256
  ```

- تهيئة مستودع خال، مناسب للاستخدام كمستودع خارجي عبر SSH:

  ```
  git init --bare
  ```

# git pull

**سحب فرع من مستودع خارجي ودمجه في المستودع المحلي.**

**لمزيد من التفاصيل**: https://git-scm.com/docs/git-pull.

- تنزيل التعديلات من المستودع الخارجي ودمجها:

  ```
  git pull
  ```

- تنزيل التعديلات من المستودع الخارجي مع إعادة ترتيب (rebase) التعديلات:

  ```
  git pull {{[-r|--rebase]}}
  ```

- تنزيل التعديلات من مستودع خارجي وفرع محددين، ثم دمجها في HEAD:

  ```
  git pull {{remote_name}} {{branch}}
  ```

# git push

**دفع التعديلات إلى مستودع** Git **خارجي**.

**لمزيد من التفاصيل**: https://git-scm.com/docs/git-push.

- دفع التعديلات المحلية في الفرع الحالي إلى نظيره في المستودع الخارجي:

  ```
  git push
  ```

- دفع التعديلات من فرع محلي معين إلى نظيره في المستودع الخارجي:

  ```
  git push {{remote_name}} {{local_branch}}
  ```

- دفع التعديلات من فرع محلي معين إلى نظيره في المستودع الخارجي، وتعيين الفرع الخارجي كهدف افتراضي لعمليات الدفع والسحب:

  ```
  git push {{[-u|--set-upstream]}} {{remote_name}} {{local_branch}}
  ```

- دفع التعديلات من فرع محلي معين إلى فرع خارجي محدد:

  ```
  git push {{remote_name}} {{local_branch}}:{{remote_branch}}
  ```

- دفع جميع الفروع المحلية إلى نظائرها في المستودع الخارجي:

  ```
  git push --all {{remote_name}}
  ```

- حذف فرع معين من المستودع الخارجي:

  ```
  git push {{remote_name}} {{[-d|--delete]}} {{remote_branch}}
  ```

- حذف الفروع البعيدة التي لا تمتلك نظيرًا محليًا بعد الآن:

  ```
  git push --prune {{remote_name}}
  ```

- نشر العلامات (tags) غير الموجودة في المستودع الخارجي:

  ```
  git push --tags
  ```

# git

**نظام تحكم في الإصدارات.**

**بعض الأوامر الفرعية مثل `commit` و `add` و `branch` و `checkout` و `push`، وغيرها، لديها وثائق استخدام خاصة بها.**

**لمزيد من التفاصيل**: https://git-scm.com/docs/git.

- تنفيذ أمر فرعي في Git:

  ```
  git {{subcommand}}
  ```

- تنفيذ أمر فرعي في Git على مسار مستودع مخصص:

  ```
  git -C {{path/to/repo}} {{subcommand}}
  ```

- تنفيذ أمر فرعي في Git مع ضبط إعداد معين:

  ```
  git -c '{{config.key}}={{value}}' {{subcommand}}
  ```

- عرض المساعدة:

  ```
  git --help
  ```

- عرض المساعدة لأمر فرعي محدد (مثل `log`، أو `push` أو `add` أو `clone`، إلخ):

  ```
  git help {{subcommand}}
  ```

- عرض رقم الإصدار:

  ```
  git --version
  ```

# gnmic sub

**هذا الأمر هو اسم مستعار لـ `gnmic subscribe`.**

- إعرض التوثيقات للأمر الأصلي:

```
tldr gnmic subscribe
```

# go

**إدارة كود مصدر** Go.

**تحتوي على توثيق استخدام خاص بها `build` بعض الأوامر الفرعية مثل**.

**لمزيد من التفاصيل**: https://go.dev/.

- تنزيل وتثبيت حزمة محددة بالمسار الخاص بها:

  ```
  go get {{package_path}}
  ```

- ترجمة وتشغيل ملف مصدر (يجب أن يحتوي على حزمة `main`):

  ```
  go run {{file}}.go
  ```

- ترجمة ملف مصدر إلى ملف تنفيذي باسم محدد:

  ```
  go build -o {{executable}} {{file}}.go
  ```

- ترجمة الحزمة الموجودة في المجلد الحالي:

  ```
  go build
  ```

- تنفيذ جميع اختبارات الحزمة الحالية (يجب أن تنتهي الملفات بـ `_test.go`):

  ```
  go test
  ```

- ترجمة وتثبيت الحزمة الحالية:

  ```
  go install
  ```

- تهيئة وحدة جديدة في المجلد الحالي:

  ```
  go mod init {{module_name}}
  ```

# gpg

**برنامج** GNU Privacy Guard.

**لمزيد من التفاصيل**: https://gnupg.org/documentation/manuals/gnupg/Invoking-GPG.html.

- عام وخاص بطريقة تفاعلية GPG إنشاء مفتاح:

```
gpg {{[--full-gen-key|--full-generate-key]}}
```

- توقيع الملف `doc.txt` دون تشفير (يتم حفظ الإخراج في ملف `doc.txt.asc`):

```
gpg --clearsign {{doc.txt}}
```

- تشفير وتوقيع الملف `doc.txt` للمستخدمين alice@example.com و bob@example.com (يتم حفظ الإخراج في ملف `doc.txt.gpg`):

```
gpg {{[-es|--encrypt --sign]}} {{[-r|--recipient]}}
{{alice@example.com}} {{[-r|--recipient]}}
{{bob@example.com}} {{doc.txt}}
```

- تشفير `doc.txt` في ملف (يتم حفظ الإخراج فقط باستخدام كلمة مرور `doc.txt.gpg`):

```
gpg {{[-c|--symmetric]}} {{doc.txt}}
```

- فك تشفير `doc.txt.gpg` (يتم عرض الإخراج على `stdout`):

```
gpg {{[-d|--decrypt]}} {{doc.txt.gpg}}
```

- استيراد مفتاح عام:

```
gpg --import {{public.gpg}}
```

- تصدير المفتاح العام للمستخدم alice@example.com (يتم عرض الإخراج على `stdout`):

```
gpg --export {{[-a|--armor]}} {{alice@example.com}}
```

- تصدير المفتاح الخاص للمستخدم alice@example.com (يتم عرض الإخراج على `stdout`):

```
gpg --export-secret-keys {{[-a|--armor]}}
{{alice@example.com}}
```

# grep

**البحث عن أنماط في الملفات باستخدام التعابير النمطية** (Regular Expressions).

**لمزيد من التفاصيل**: https://www.gnu.org/software/grep/manual/grep.html.

- البحث عن نمط داخل ملف:

```
grep "{{search_pattern}}" {{path/to/file}}
```

- البحث عن سلسلة نصية مطابقة تمامًا (تعطيل التعابير النمطية):

```
grep {{[-F|--fixed-strings]}} "{{exact_string}}" {{path/to/
file}}
```

- البحث عن نمط في جميع الملفات داخل دليل بشكل متكرر، مع عرض أرقام الأسطر المطابقة، وتجاهل الملفات الثنائية:

```
grep {{[-rnI|--recursive --line-number --binary-
files=without-match]}} "{{search_pattern}}" {{path/to/
directory}}
```

- في وضع عدم التمييز بين الأحرف الكبيرة والصغيرة، (يدعم استخدام التعابير النمطية الموسعة ?, +, {}, ( ), و |):

```
grep {{[-Ei|--extended-regexp --ignore-case]}}
"{{search_pattern}}" {{path/to/file}}
```

- طباعة 3 أسطر من السياق حول، قبل أو بعد كل تطابق:

```
grep {{--context|--before-context|--after-context}} 3
"{{search_pattern}}" {{path/to/file}}
```

- طباعة اسم الملف ورقم السطر لكل تطابق مع تمييز بالألوان:

```
grep {{[-Hn|--with-filename --line-number]}} --color=always
"{{search_pattern}}" {{path/to/file}}
```

- البحث عن الأسطر المطابقة لنمط معين، مع طباعة النص المطابق فقط:

```
grep {{[-o|--only-matching]}} "{{search_pattern}}" {{path/to/
file}}
```

- البحث في `stdin` عن الأسطر التي لا تطابق النمط:

```
cat {{path/to/file}} | grep {{[-v|--invert-match]}}
"{{search_pattern}}"
```

# host

**البحث في خادم أسماء النطاقات** (DNS).

**لمزيد من التفاصيل**: https://manned.org/host.

- البحث عن سجلات A و AAAA و MX لنطاق معين:

  ```
  host {{domain}}
  ```

- البحث عن نوع معين من السجلات (مثل CNAME أو TXT) لنطاق معين:

  ```
  host -t {{field}} {{domain}}
  ```

- البحث العكسي عن عنوان IP:

  ```
  host {{ip_address}}
  ```

- تحديد خادم DNS بديل للاستعلام:

  ```
  host {{domain}} {{8.8.8.8}}
  ```

# htop

**عرض معلومات ديناميكية في الوقت الحالي عن العمليات الجارية. نسخة محسنة من `top`.**

**لمزيد من التفاصيل**: https://htop.dev/.

- تشغيل `htop`:

  ```
  htop
  ```

- تشغيل `htop` لعرض العمليات المملوكة من قبل مستخدم معين:

  ```
  htop {{[-u|--user]}} {{username}}
  ```

- عرض العمليات بشكل هرمي في عرض متفرع لإظهار علاقات الأصل والفرع:

  ```
  htop {{[-t|--tree]}}
  ```

- لعرض `htop --sort help` المحدد (استخدم `sort_item` فرز العمليات حسب الخيارات المتاحة):

  ```
  htop {{[-s|--sort]}} {{sort_item}}
  ```

- تشغيل `htop` مع تأخير محدد بين التحديثات، بوحدات عشر الثانية (مثال: 50 = 5 ثواني):

  ```
  htop {{[-d|--delay]}} {{50}}
  ```

- عرض الأوامر التفاعلية أثناء تشغيل htop:

  ```
  <?>
  ```

- التبديل إلى علامة تبويب أخرى:

  ```
  <Tab>
  ```

- عرض المساعدة:

  ```
  htop {{[-h|--help]}}
  ```

# id

**يعرض معرف المستخدم الحالي ومعرف المجموعة.**

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/id-invocation.html.

- عرض معرف المستخدم الحالي (UID) ومعرف المجموعة (GID) والمجموعات التي ينتمي إليها:

  ```
  id
  ```

- عرض اسم المستخدم الحالي:

  ```
  id {{[-un|--user --name]}}
  ```

- عرض معرف المستخدم الحالي كرقم:

  ```
  id {{[-u|--user]}}
  ```

- عرض معرف المجموعة الأساسية الحالي:

  ```
  id {{[-gn|--group --name]}}
  ```

- عرض معرف المجموعة الأساسية الحالي كرقم:

  ```
  id {{[-g|--group]}}
  ```

- عرض معرف مستخدم آخر (UID) ومعرف المجموعة (GID) والمجموعات التي ينتمي إليها:

  ```
  id {{username}}
  ```

# ifconfig

**مُكوِّن واجهة الشبكة.**

**لمزيد من التفاصيل**: https://net-tools.sourceforge.io/man/ifconfig.8.html.

- عرض إعدادات الشبكة لواجهة محددة:

  ```
  ifconfig {{interface_name}}
  ```

- عرض تفاصيل جميع الواجهات، بما في ذلك الواجهات المعطلة:

  ```
  ifconfig -a
  ```

- تعطيل واجهة محددة:

  ```
  ifconfig {{interface_name}} down
  ```

- تمكين واجهة محددة:

  ```
  ifconfig {{interface_name}} up
  ```

- تعيين عنوان IP لواجهة محددة:

  ```
  ifconfig {{interface_name}} {{ip_address}}
  ```

# kill

**إرسال إشارة إلى عملية** (Process)**، عادةً لإيقافها**.

**أن يتم اعتراضها بواسطة** SIGSTOP **و** SIGKILL **يمكن لجميع الإشارات باستثناء العملية لإنهاء نظيف**.

**لمزيد من التفاصيل**: https://manned.org/kill.1posix.

- إنهاء برنامج باستخدام الإشارة الافتراضية SIGTERM (إنهاء):

```
kill {{process_id}}
```

- عرض أسماء الإشارات المتاحة (تُستخدم بدون بادئة `SIG`):

```
kill -l
```

- إنهاء برنامج باستخدام الإشارة SIGHUP (قطع الاتصال). العديد من الخدمات (Daemons) ستقوم بإعادة التحميل بدلاً من الإنهاء:

```
kill -{{1|HUP}} {{process_id}}
```

- إنهاء برنامج باستخدام الإشارة SIGINT (المقاطعة). يتم تنفيذ ذلك عادةً عند ضغط المستخدم على `<Ctrl c>`:

```
kill -{{2|INT}} {{process_id}}
```

- إرسال إشارة إلى نظام التشغيل لإنهاء البرنامج فورًا (لن تتاح له فرصة لمعالجة الإشارة):

```
kill -{{9|KILL}} {{process_id}}
```

- إرسال إشارة إلى نظام التشغيل لإيقاف برنامج مؤقتًا حتى يتم استلام إشارة SIGCONT ("متابعة"):

```
kill -{{17|STOP}} {{process_id}}
```

- إرسال إشارة `SIGUSR1` إلى جميع العمليات التي تملك معرف المجموعة المحدد (GID):

```
kill -{{SIGUSR1}} -{{group_id}}
```

# ls

**إدراج محتويات مجلد**.

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/ls-invocation.html.

- إدراج كل الملفات في أسطر منفصلة:

```
ls -1
```

- إدراج جميع الملفات بما فيها الملفات المخفية:

```
ls {{[-a|--all]}}
```

- إدراج جميع الملفات مع إضافة / لنهاية أسماء المللفات:

```
ls {{[-F|--classify]}}
```

- إدراج الملفات و معلوماتها لتشمل اللأذونات و الملكية و الحجم و تاريخ التغيير:

```
ls {{[-la|--all -l]}}
```

- إدراج اللملفات بصيغة طويلة مع حجم الملفات بوحدات مقروءة (KiB, MiB, GiB):

```
ls {{[-lh|-l --human-readable]}}
```

- صيغة طويلة للملفات مرتبة تنازليا حسب اللحجم:

```
ls {{-lSR|-lS --recursive}}
```

- صيغة طويلة للملفات مرتبة تنازليا حسب التاريخ الأقدم اولا:

```
ls {{[-ltr|-lt --reverse]}}
```

- إدراج المجلدات فقط:

```
ls {{[-d|--directory]}} */
```

# lsof

**يعرض قائمة الملفات المفتوحة والعمليات المرتبطة بها.**

**لعرض الملفات المفتوحة** (sudo **أو**) **ملاحظة: تتطلب بعض الأوامر صلاحيات الجذر من قبل الآخرين.**

**لمزيد من التفاصيل**: https://manned.org/lsof.

- العثور على العمليات التي فتحت ملفًا معينًا:

```
lsof {{path/to/file}}
```

- العثور على العملية التي فتحت منفذ إنترنت محدد:

```
lsof -i :{{port}}
```

- عرض معرف العملية (PID) فقط:

```
lsof -t {{path/to/file}}
```

- عرض الملفات المفتوحة بواسطة مستخدم معين:

```
lsof -u {{username}}
```

- عرض الملفات المفتوحة بواسطة أمر أو عملية معينة:

```
lsof -c {{process_or_command_name}}
```

- عرض الملفات المفتوحة بواسطة عملية معينة باستخدام PID:

```
lsof -p {{PID}}
```

- عرض الملفات المفتوحة داخل مُجَلَّد معين:

```
lsof +D {{path/to/directory}}
```

- العثور على العملية التي تستمع على منفذ محلي عبر TCP دون تحويل أرقام الشبكة IPv6 أو المنافذ:

```
lsof -i6TCP:{{port}} -sTCP:LISTEN -n -P
```

# mtr

**أداة** Matt's Traceroute: **تجمع بين** traceroute **و** ping.

**لمزيد من التفاصيل**: https://manned.org/mtr.

- مستمرة لجميع النقاط الوسيطة ping تتبع المسار إلى مضيف وإرسال حزم:

  ```
  mtr {{example.com}}
  ```

- تعطيل تعيين عناوين IP وأسماء المضيفين:

  ```
  mtr {{[-n|--no-dns]}} {{example.com}}
  ```

- لكل نقطة ping عرض المخرجات بعد إرسال 10 حزم:

  ```
  mtr {{[-w|--report-wide]}} {{example.com}}
  ```

- فرض استخدام IPv4 أو IPv6:

  ```
  mtr -4 {{example.com}}
  ```

- الانتظار لوقت محدد (بالثواني) قبل إرسال حزمة أخرى إلى نفس النقطة:

  ```
  mtr {{[-i|--interval]}} {{10}} {{example.com}}
  ```

- لكل نقطة (ASN) عرض رقم النظام المستقل:

  ```
  mtr {{[-z|--aslookup]}} {{example.com}}
  ```

- عرض كل من عنوان IP والاسم العكسي لـ DNS:

  ```
  mtr {{[-b|--show-ips]}} {{example.com}}
  ```

# newsboat

**هو قارئ خلاصة آر إس إس للطرفية أو الكونسول**.

**لمزيد من التفاصيل**: https://newsboat.org/.

- إستيراد روابط الخلاصات من ملف OPML:

  ```
  newsboat -i {{my-feeds.xml}}
  ```

- إضافة روابط الخلاصات يدوياً:

  ```
  echo {{http://example.com/path/to/feed}} >> "$
  {HOME}/.newsboat/urls"
  ```

- إبدأ newsboat وقم بتحديث كل الخلاصات عند بدء التشغيل:

  ```
  newsboat -r
  ```

- نفذ أمر أو عدة أوامر مفصولة بمسافات بدون الحاجة إلي فتح newsboat:

  ```
  newsboat -x {{reload print-unread ...}}
  ```

- انظر إختصارات لوحة المفاتيح (الإختصارت الأكثر شيوعاً مرئية في شريط الحالة):

  ```
  <?>
  ```

# node

**بيئة تشغيل** JavaScript **للخادم** (Node.js).

**لمزيد من التفاصيل**: https://nodejs.org.

- تشغيل ملف JavaScript:

  ```
  node {{path/to/file}}
  ```

- بدء REPL (وحدة تحكم تفاعلية):

  ```
  node
  ```

- تنفيذ الملف المحدد مع إعادة تشغيل العملية عند تغيير ملف مستورد (يتطلب Node.js إصدار 18.11+):

  ```
  node --watch {{path/to/file}}
  ```

- تنفيذ كود JavaScript من سطر الأوامر:

  ```
  node {{[-e|--eval]}} "{{code}}"
  ```

- تنفيذ وطباعة النتيجة، مفيد لطباعة إصدارات تبعيات node:

  ```
  node {{[-p|--print]}} "process.versions"
  ```

- تفعيل المصحح (inspector)، مع إيقاف التنفيذ حتى يتم الاتصال بمصحح الأخطاء (debugger) بالكامل بمجرد تحليل الكود المصدري:

  ```
  node --no-lazy --inspect-brk {{path/to/file}}
  ```

# openai

**أداة سطر الأوامر للوصول إلى واجهة برمجة تطبيقات** OpenAI.

**لمزيد من التفاصيل**: https://github.com/openai/openai-python.

- عرض قائمة النماذج المتاحة:

```
openai api models.list
```

- إنشاء إكمال نصي:

```
openai api completions.create --model {{ada}} --prompt
"{{Hello world}}"
```

- إنشاء إكمال محادثة:

```
openai api chat_completions.create --model {{gpt-3.5-turbo}}
--message {{user "Hello world"}}
```

- إنشاء صور باستخدام API الخاصة بـ DALL·E:

```
openai api image.create --prompt "{{كلبان يلعبان الشطرنج،
كرتوني}}" --num-images {{1}}
```

# openssl req

**أمر** OpenSSL **لإدارة طلبات توقيع الشهادات** PKCS#10.

**لمزيد من التفاصيل**: https://www.openssl.org/docs/manmaster/man1/openssl-req.html.

- إنشاء طلب توقيع شهادة لإرساله إلى جهة تصديق:

```
openssl req -new -sha256 -key {{filename.key}} -out
{{filename.csr}}
```

- إنشاء شهادة موقعة ذاتيًا وزوج مفاتيح مقابلة، وحفظهما في ملف:

```
openssl req -new -x509 -newkey {{rsa}}:{{4096}} -keyout
{{filename.key}} -out {{filename.cert}} -subj "{{/C=XX/
CN=foobar}}" -days {{365}}
```

# openssl x509

**أمر** OpenSSL **لإدارة شهادات** X.509.

**لمزيد من التفاصيل**: https://www.openssl.org/docs/manmaster/man1/openssl-x509.html.

- عرض معلومات الشهادة:

```
openssl x509 -in {{filename.crt}} -noout -text
```

- عرض تاريخ انتهاء صلاحية الشهادة:

```
openssl x509 -enddate -noout -in {{filename.pem}}
```

- تحويل الشهادة بين الترميز الثنائي DER والترميز النصي PEM:

```
openssl x509 -inform {{der}} -outform {{pem}} -in
{{original_certificate_file}} -out
{{converted_certificate_file}}
```

- تخزين المفتاح العام للشهادة في ملف:

```
openssl x509 -in {{certificate_file}} -noout -pubkey -out
{{output_file}}
```

# openssl

**مجموعة أدوات تشفير** OpenSSL.

**تحتوي على وثائق استخدام خاصة بها `req` بعض الأوامر الفرعية مثل**.

**لمزيد من التفاصيل**: https://www.openssl.org.

- عرض الإرشادات:

  ```
  openssl help
  ```

- عرض الإرشادات لأمر فرعي محدد:

  ```
  openssl help {{x509}}
  ```

- عرض الإصدار:

  ```
  openssl version
  ```

# pgrep

**البحث عن العمليات أو إرسال إشارات إليها باستخدام الاسم**.

**لمزيد من التفاصيل**: https://manned.org/pgrep.

- لأي عمليات جارية تتطابق مع اسم العملية عرض معرّفات العمليات (PIDs):

  ```
  pgrep {{process_name}}
  ```

- البحث عن العمليات مع الخيارات المستخدمة في سطر الأوامر:

  ```
  pgrep {{[-f|--full]}} "{{process_name}} {{parameter}}"
  ```

- البحث عن العمليات التي يتم تشغيلها بواسطة مستخدم معين:

  ```
  pgrep {{[-u|--euid]}} root {{process_name}}
  ```

# picotool

**إدارة لوحات** Raspberry Pi Pico.

**لمزيد من التفاصيل**: https://github.com/raspberrypi/picotool.

- عرض معلومات حول البرنامج المحمّل حاليًا على جهاز Pico:

  ```
  picotool info
  ```

- تحميل ملف ثنائي (binary) على جهاز Pico:

  ```
  picotool load {{path/to/binary}}
  ```

- تحويل ملف ELF أو BIN إلى تنسيق UF2:

  ```
  picotool uf2 convert {{path/to/elf_or_bin}} {{path/to/output}}
  ```

- إعادة تشغيل جهاز Pico:

  ```
  picotool reboot
  ```

- عرض جميع السجلات (registers) المعروفة:

  ```
  picotool otp list
  ```

- عرض إصدار الأداة:

  ```
  picotool version
  ```

- عرض المساعدة:

  ```
  picotool help
  ```

# pio init

**هذا الأمر هو اسم مستعار لـ `pio project`.**

- إعرض التوثيقات للأمر الأصلي:

  ```
  tldr pio project
  ```

# pip

**مدير الحزم الخاص بلغة بايثون.**

**بعض الأوامر الفرعية مثل `install` لديها توثيق خاص بها.**

**لمزيد من التفاصيل**: https://pip.pypa.io.

- تثبيت حزمة (راجع `pip install` لمزيد من خيارات التثبيت):

  ```
  pip install {{package}}
  ```

- تثبيت حزمة في مجلد المستخدم بدلاً من الموقع الافتراضي للنظام:

  ```
  pip install --user {{package}}
  ```

- تحديث حزمة مثبتة:

  ```
  pip install {{[-U|--upgrade]}} {{package}}
  ```

- إزالة تثبيت حزمة:

  ```
  pip uninstall {{package}}
  ```

- حفظ قائمة الحزم المثبتة في ملف:

  ```
  pip freeze > {{requirements.txt}}
  ```

- عرض معلومات عن حزمة مثبتة:

  ```
  pip show {{package}}
  ```

- تثبيت الحزم من ملف متطلبات:

  ```
  pip install {{[-r|--requirement]}} {{requirements.txt}}
  ```

# pkill

**إرسال إشارة إلى العملية حسب الاسم.**

**يُستخدم غالبًا لإيقاف العمليات.**

**لمزيد من التفاصيل**: https://manned.org/pkill.

- إنهاء جميع العمليات التي تطابق اسم محدد:

  ```
  pkill "{{process_name}}"
  ```

- إنهاء جميع العمليات التي تطابق الأمر الكامل بدلًا من مجرد اسم العملية:

  ```
  pkill {{[-f|--full]}} "{{command_name}}"
  ```

- فرض إنهاء العمليات المطابقة (لا يمكن حظره):

  ```
  pkill -9 "{{process_name}}"
  ```

- إرسال إشارة `SIGUSR1` إلى العمليات التي تطابق الاسم المحدد:

  ```
  pkill -USR1 "{{process_name}}"
  ```

- إنهاء العملية الرئيسية لـ `firefox` لإغلاق المتصفح:

  ```
  pkill {{[-o|--oldest]}} "{{firefox}}"
  ```

# ps

**معلومات عن العمليات قيد التشغيل.**

**لمزيد من التفاصيل**: https://manned.org/ps.

- عرض جميع العمليات قيد التشغيل:

  ```
  ps aux
  ```

- عرض جميع العمليات قيد التشغيل متضمنة سلسلة الأوامر الكاملة:

  ```
  ps auxww
  ```

- البحث عن عملية تطابق سلسلة معينة (الأقواس تمنع `grep` من مطابقة نفسه):

  ```
  ps aux | grep {{[s]tring}}
  ```

- عرض جميع عمليات المستخدم الحالي بتنسيق مفصل إضافي:

  ```
  ps {{[-u|--user]}} $(id {{[-u|--user]}}) -F
  ```

- عرض جميع عمليات المستخدم الحالي على شكل متفرع:

  ```
  ps {{[-u|--user]}} $(id {{[-u|--user]}}) f
  ```

- الحصول على معرّف العملية الأب (PPID) لعملية معينة:

  ```
  ps {{[-o|--format]}} ppid= {{[-p|--pid]}} {{pid}}
  ```

- فرز العمليات حسب استهلاك الذاكرة:

  ```
  ps --sort size
  ```

# pwd

**اطبع اسم الدليل الحالي**.

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/pwd-invocation.html.

- اطبع اسم الدليل الحالي:

  ```
  pwd
  ```

- اطبع اسم الدليل الحالي و حل جميع الروابط اللينة (وبمعنى آخر إظهار المسارالفعلي) :

  ```
  pwd {{[-P|--physical]}}
  ```

# rm

**يستخدم الأمر لحذف الملفات او المجلدات**

**أنظر أيضًا**: `rmdir`.

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/rm-invocation.html.

- حذف ملفات محددة:

```
rm {{path/to/file1 path/to/file2 ...}}
```

- حذف ملفات محددة وتجاهل الملفات الغير موجودة:

```
rm {{[-f|--force]}} {{path/to/file1 path/to/file2 ...}}
```

- حذف ملفات محددة مع واجهة تفاعلية قبل اي حذف للتأكد:

```
rm {{[-i|--interactive]}} {{path/to/file1 path/to/file2 ...}}
```

- حذف ملفات محددة مع عرض تفاصيل حول كل عملية حذف:

```
rm {{[-v|--verbose]}} {{path/to/file1 path/to/file2 ...}}
```

- حذف ملفات ومجلدات محددة بشكل تسلسلي:

```
rm {{[-r|--recursive]}} {{path/to/file_or_directory1 path/to/
file_or_directory2 ...}}
```

- حذف المجلدات الفارغة (هذه الطريقة تعتبر آمنة):

```
rm {{[-d|--dir]}} {{path/to/directory}}
```

# rmdir

**إزالة الدلائل بدون ملفات.**

**لمزيد من التفاصيل**: `rm`.

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/rmdir-invocation.html.

- إزالة أدلة محددة:

```
rmdir {{path/to/directory1 path/to/directory2 ...}}
```

- إزالة أدلة متداخلة محددة بشكل متكرر:

```
rmdir -p {{path/to/directory1 path/to/directory2 ...}}
```

# robo

**مشغل مهام** PHP.

**لمزيد من التفاصيل**: https://robo.li/getting-started.html.

- عرض قائمة الأوامر المتوفرة:

  ```
  robo list
  ```

- تشغيل أمر محدد:

  ```
  robo {{foo}}
  ```

- محاكاة تشغيل أمر محدد:

  ```
  robo --simulate {{foo}}
  ```

# shred

**الكتابة فوق الملفات لحذف البيانات بشكل آمن.**

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/shred-invocation.html.

- الكتابة فوق ملف:

```
shred {{path/to/file}}
```

- الكتابة فوق ملف وعرض التقدم على الشاشة:

```
shred {{-v|--verbose}} {{path/to/file}}
```

- الكتابة فوق ملف وترك أصفار بدلاً من البيانات العشوائية:

```
shred {{-z|--zero}} {{path/to/file}}
```

- الكتابة فوق ملف عددًا معينًا من المرات:

```
shred {{-n|--iterations}} {{25}} {{path/to/file}}
```

- الكتابة فوق ملف ثم إزالته:

```
shred --remove {{path/to/file}}
```

- الكتابة فوق ملف 100 مرة، إضافة كتابة نهائية بالأصفار، حذفه بعد الكتابة وعرض تقدم العملية بشكل مفصل:

```
shred {{-vzun|--verbose --zero -u --iterations}} 100 {{path/to/file}}
```

# stat

**عرض معلومات عن الملف ونظام الملفات.**

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/stat-invocation.html.

- عرض خصائص ملف معين مثل الحجم، الأذونات، تواريخ الإنشاء والوصول، وغيرها:

```
stat {{path/to/file}}
```

- عرض خصائص ملف معين مثل الحجم، الأذونات، تواريخ الإنشاء والوصول، وغيرها بدون تسميات:

```
stat {{-t|--terse}} {{path/to/file}}
```

- عرض معلومات عن نظام الملفات حيث يوجد ملف معين:

```
stat {{-f|--file-system}} {{path/to/file}}
```

- عرض أذونات الملف بصيغة ثُمانية فقط:

```
stat {{-c|--format}} "%a %n" {{path/to/file}}
```

- عرض مالك الملف والمجموعة التابعة له:

```
stat {{-c|--format}} "%U %G" {{path/to/file}}
```

- عرض حجم ملف معين بالبايت:

```
stat {{-c|--format}} "%s %n" {{path/to/file}}
```

# tail

**عرض الجزء الأخير من ملف**.

**انظر أيضًا**: **head**.

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/tail-invocation.html.

- عرض آخر 'عدد' من الأسطر في ملف:

```
tail {{[-n|--lines]}} {{count}} {{path/to/file}}
```

- طباعة محتوى ملف بدءًا من سطر معين:

```
tail {{[-n|--lines]}} +{{count}} {{path/to/file}}
```

- طباعة عدد معين من البايتات من نهاية ملف معين:

```
tail {{[-c|--bytes]}} {{count}} {{path/to/file}}
```

- طباعة آخر الأسطر من ملف معين والاستمرار في قراءته حتى الضغط على `<Ctrl c>`:

```
tail {{[-f|--follow]}} {{path/to/file}}
```

- الاستمرار في قراءة الملف حتى الضغط على `<Ctrl c>`، حتى لو كان غير متاح:

```
tail {{[-F|--retry --follow]}} {{path/to/file}}
```

- عرض آخر 'عدد' من الأسطر في 'ملف' وتحديث العرض كل 'عدد' من الثواني:

```
tail {{[-n|--lines]}} {{count}} {{[-s|--sleep-interval]}} {{seconds}} {{[-f|--follow]}} {{path/to/file}}
```

# tar

**أداة أرشفة.**

**غالبًا ما تُستخدم مع طريقة ضغط، مثل `gzip` أو `bzip2`.**

**لمزيد من التفاصيل**: https://www.gnu.org/software/tar.

- إنشاء أرشيف وكتابته إلى ملف:

```
tar cf {{path/to/target.tar}} {{path/to/file1 path/to/file2 ...}}
```

- إنشاء أرشيف وكتابته إلى ملف (gzipped):

```
tar czf {{path/to/target.tar.gz}} {{path/to/file1 path/to/file2 ...}}
```

- إنشاء أرشيف مضغوط من مجلد باستخدام المسارات النسبية:

```
tar czf {{path/to/target.tar.gz}} {{[-C|--directory]}} {{path/to/directory}} .
```

- فك ضغط ملف أرشيف مضغوط في المجلد الحالي:

```
tar xvf {{path/to/source.tar[.gz|.bz2|.xz]}}
```

- فك ضغط ملف أرشيف مضغوط في مجلد محدد:

```
tar xf {{path/to/source.tar[.gz|.bz2|.xz]}} {{[-C|--directory]}} {{path/to/directory}}
```

- إنشاء أرشيف مضغوط وكتابته إلى ملف، مع تحديد خوارزمية الضغط تلقائيًا بناءً على امتداد الملف:

```
tar caf {{path/to/target.tar.xz}} {{path/to/file1 path/to/file2 ...}}
```

- عرض محتويات ملف بالتفصيل:

```
tar tvf {{path/to/source.tar}}
```

- فك ضغط الملفات المطابقة لنمط معين من ملف أرشيف:

```
tar xf {{path/to/source.tar}} --wildcards "{{*.html}}"
```

# tcpdump

**عرض وتحليل حركة المرور على الشبكة.**

**لمزيد من التفاصيل**: https://www.tcpdump.org.

- عرض قائمة بواجهات الشبكة المتوفرة:

```
tcpdump {{[-D|--list-interfaces]}}
```

- التقاط حركة المرور لواجهة شبكة محددة:

```
sudo tcpdump {{[-i|--interface]}} {{eth0}}
```

- التقاط جميع حركة مرور TCP مع عرض المحتويات (ASCII) في وحدة التحكم:

```
tcpdump -A tcp
```

- التقاط حركة المرور من أو إلى مضيف محدد:

```
tcpdump host {{www.example.com}}
```

- التقاط حركة المرور من واجهة معينة مع مصدر، وجهة ومنفذ وجهة محددين:

```
sudo tcpdump {{[-i|--interface]}} {{eth0}} src
{{192.168.1.1}} and dst {{192.168.1.2}} and dst port {{80}}
```

- التقاط حركة المرور لشبكة محددة:

```
tcpdump net {{192.168.1.0/24}}
```

- التقاط جميع حركة المرور باستثناء حركة المرور على المنفذ 22 وحفظها في ملف:

```
tcpdump -w {{dumpfile.pcap}} port not {{22}}
```

- قراءة البيانات من ملف محدد:

```
tcpdump -r {{dumpfile.pcap}}
```

# tldr

**يعرض صفحات مساعدة مبسطة للأوامر في سطر الأوامر، مستمدة من مشروع** tldr-pages.

**ليست مطلوبة وفقًا للمواصفات، ولكن `--list` و `--language` ملاحظة: الخيارات معظم العملاء يدعمونها.**

**لمزيد من التفاصيل**: https://github.com/tldr-pages/tldr/blob/main/CLIENT-SPECIFICATION.md#command-line-interface.

- عرض صفحة tldr (تلميح: هذا ما أوصلك إلى هنا!) لأمر معين:

  ```
  tldr {{command}}
  ```

- عرض صفحة tldr لأمر فرعي معين:

  ```
  tldr {{command}} {{subcommand}}
  ```

- عرض صفحة tldr (إن وجدت، وإلا سيتم الرجوع إلى الإنجليزية) لأمر بلغة معينة:

  ```
  tldr {{[-L|--language]}} {{language_code}} {{command}}
  ```

- عرض صفحة tldr لأمر من نظام تشغيل معين:

  ```
  tldr {{[-p|--platform]}} {{android|common|freebsd|linux|osx|
  netbsd|openbsd|sunos|windows}} {{command}}
  ```

- تحديث ذاكرة التخزين المؤقت لصفحات tldr:

  ```
  tldr {{[-u|--update]}}
  ```

- عرض قائمة بجميع الصفحات المتاحة للمنصة الحالية وللأوامر الشائعة:

  ```
  tldr {{[-l|--list]}}
  ```

- عرض جميع الصفحات الفرعية المتاحة لأمر معين:

  ```
  tldr {{[-l|--list]}} | grep {{command}} | column
  ```

- عرض صفحة tldr لأمر عشوائي:

  ```
  tldr {{[-l|--list]}} | shuf {{[-n|--head-count]}} 1 | xargs
  tldr
  ```

# tlmgr arch

**هذا الأمر هو اسم مستعار لـ `tlmgr platform`.**

- إعرض التوثيقات للأمر الأصلي:

  ```
  tldr tlmgr platform
  ```

# tox

**أتمتة اختبارات بايثون عبر إصدارات بايثون متعددة.**

**استخدم** tox.ini **لضبط البيئات وأمر الاختبار.**

**لمزيد من التفاصيل**: https://github.com/tox-dev/tox.

- بدء الاختبارات على جميع بيئات الاختبار:

  ```
  tox
  ```

- إنشاء ملف الإعدادات `tox.ini`:

  ```
  tox-quickstart
  ```

- عرض قائمة جميع البيئات المتوفرة:

  ```
  tox --listenvs-all
  ```

- بدء الاختبارات على بيئة معينة (مثال: بايثون 3.6):

  ```
  tox -e {{py36}}
  ```

- إجبار إعادة إنشاء البيئة الافتراضية:

  ```
  tox --recreate -e {{py27}}
  ```

# tr

**ترجمة الأحرف: تنفيذ عمليات الاستبدال بناءً على أحرف مفردة ومجموعات أحرف.**

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/tr-invocation.html.

- استبدال جميع تكرارات حرف معين في ملف وطباعة النتيجة:

```
tr {{find_character}} {{replace_character}} < {{path/to/file}}
```

- استبدال جميع تكرارات حرف معين من ناتج أمر آخر:

```
echo {{text}} | tr {{find_character}} {{replace_character}}
```

- تعيين كل حرف من المجموعة الأولى إلى الحرف المقابل في المجموعة الثانية:

```
tr '{{abcd}}' '{{jkmn}}' < {{path/to/file}}
```

- حذف جميع تكرارات مجموعة الأحرف المحددة من المدخلات:

```
tr {{[-d|--delete]}} '{{input_characters}}' < {{path/to/file}}
```

- ضغط سلسلة من الأحرف المتطابقة إلى حرف واحد:

```
tr {{[-s|--squeeze-repeats]}} '{{input_characters}}' < {{path/to/file}}
```

- تحويل محتويات ملف إلى أحرف كبيرة (Upper-case):

```
tr "[:lower:]" "[:upper:]" < {{path/to/file}}
```

- إزالة الأحرف غير القابلة للطباعة من ملف:

```
tr {{[-cd|--complement --delete]}} "[:print:]" < {{path/to/file}}
```

# ulimit

**الحصول على وتعيين حدود الموارد لعمليات المستخدم**.

**لمزيد من التفاصيل**: https://www.gnu.org/software/bash/manual/bash.html#index-ulimit.

- عرض خصائص جميع حدود المستخدم:

```
ulimit -a
```

- عرض الحد الأقصى الثابت لعدد الملفات التي يمكن فتحها في نفس الوقت:

```
ulimit -H -n
```

- عرض الحد القابل للتغيير لعدد الملفات التي يمكن فتحها في نفس الوقت:

```
ulimit -S -n
```

- تعيين الحد الأقصى لعدد العمليات لكل مستخدم:

```
ulimit -u 30
```

# uptime

**يعرض مدة تشغيل النظام ومعلومات أخرى.**

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/uptime-invocation.html.

- طباعة الوقت الحالي، ومدة التشغيل، وعدد المستخدمين المسجلين، ومعلومات أخرى:

  ```
  uptime
  ```

- عرض مدة التشغيل فقط منذ بدء تشغيل النظام:

  ```
  uptime {{[-p|--pretty]}}
  ```

- طباعة تاريخ ووقت بدء تشغيل النظام:

  ```
  uptime {{[-s|--since]}}
  ```

- عرض إصدار الأداة:

  ```
  uptime {{[-V|--version]}}
  ```

# uv

**مدير حزم ومشاريع سريع للغة بايثون.**

**لديها توثيق خاص بها `python` و `tool` بعض الأوامر الفرعية مثل.**

**لمزيد من التفاصيل**: https://docs.astral.sh/uv/reference/cli.

- إنشاء مشروع بايثون جديد في المجلد الحالي:

  ```
  uv init
  ```

- إنشاء مشروع بايثون جديد في المسار المحدد:

  ```
  uv init {{path/to/directory}}
  ```

- إضافة تبعية جديدة إلى المشروع:

  ```
  uv add {{package}}
  ```

- إزالة تبعية من المشروع:

  ```
  uv remove {{package}}
  ```

- تشغيل سكربت داخل بيئة المشروع:

  ```
  uv run {{path/to/script.py}}
  ```

- تشغيل أمر داخل بيئة المشروع:

  ```
  uv run {{command}}
  ```

- تحديث بيئة المشروع من `pyproject.toml`:

  ```
  uv sync
  ```

- إنشاء ملف تأمين (lock file) لتبعيات المشروع:

  ```
  uv lock
  ```

# w

**اداة لعرض المستخدمين المتصلين حالياً و العمليات التي يتم تنفيذها**.

**لمزيد من التفاصيل**: https://manned.org/w.

- عرض معلومات عن جميع المستخدمين المتصلين حاليا:

  ```
  w
  ```

- عرض معلومات عن مستخدم محدد:

  ```
  w {{اسم_المستخدم}}
  ```

- عرض معلومات المستخدمين المتصلين بدون الترويسة:

  ```
  w {{[-h|--no-header]}}
  ```

- عرض المعلومات بدون تضمين أعمدة تسجيل الدخول و JCPU و PCPU:

  ```
  w {{[-s|--short]}}
  ```

# watch

**تنفيذ برنامج بشكل دوري ومراقبة المخرجات في وضع ملء الشاشة.**

**لمزيد من التفاصيل**: https://manned.org/watch.

- تشغيل أمر بشكل متكرر وعرض النتيجة:

  ```
  watch {{command}}
  ```

- إعادة تشغيل أمر كل 60 ثانية:

  ```
  watch {{[-n|--interval]}} 60 {{command}}
  ```

- مراقبة مساحة القرص مع تمييز الاختلافات عند ظهورها:

  ```
  watch {{[-d|--differences]}} {{df}}
  ```

- تشغيل سلسلة أوامر (بايبلاين) بشكل متكرر وعرض النتيجة:

  ```
  watch "{{command_1}} | {{command_2}} | {{command_3}}"
  ```

- إذا تغير الإخراج المرئي `watch` الخروج من:

  ```
  watch {{[-g|--chgexit]}} {{lsblk}}
  ```

- ترجمة رموز التحكم في الطرفية (Terminal):

  ```
  watch {{[-c|--color]}} {{ls --color=always}}
  ```

# wc

**عدّ الأسطر والكلمات والبايتات.**

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/wc-invocation.html.

- عدّ جميع الأسطر في ملف:

```
wc {{[-l|--lines]}} {{path/to/file}}
```

- عدّ جميع الكلمات في ملف:

```
wc {{[-w|--words]}} {{path/to/file}}
```

- عدّ جميع البايتات في ملف:

```
wc {{[-c|--bytes]}} {{path/to/file}}
```

- عدّ جميع الأحرف في ملف (مع أخذ الأحرف متعددة البايتات في الاعتبار مثل الحروف العربية):

```
wc {{[-m|--chars]}} {{path/to/file}}
```

- عدّ جميع الأسطر والكلمات والبايتات من `stdin`:

```
{{find .}} | wc
```

- حساب طول أطول سطر بعدد الأحرف:

```
wc {{[-L|--max-line-length]}} {{path/to/file}}
```

# who

**عرض المستخدمين المتصلين حاليًا والبيانات المتعلقة بهم (مثل العمليات ووقت الإقلاع)**.

**انظر أيضًا**: `whoami`.

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/who-invocation.html.

- عرض أسماء المستخدمين والخطوط الزمنية للجلسات النشطة حاليًا:

  ```
  who
  ```

- عرض جميع المعلومات المتاحة:

  ```
  who {{[-a|--all]}}
  ```

- عرض جميع المعلومات المتاحة مع عناوين الأعمدة:

  ```
  who {{[-aH|--all --heading]}}
  ```

# whois

**عميل سطر الأوامر لبروتوكول** WHOIS (RFC 3912).

**لمزيد من التفاصيل**: https://manned.org/whois.

- الحصول على معلومات حول اسم نطاق:

  ```
  whois {{example.com}}
  ```

- الحصول على معلومات حول عنوان IP:

  ```
  whois {{8.8.8.8}}
  ```

- الحصول على جهة الاتصال للإبلاغ عن إساءة استخدام عنوان IP:

  ```
  whois -b {{8.8.8.8}}
  ```

# XZ

**ضغط أو فك ضغط ملفات** XZ **و** LZMA.

**لمزيد من التفاصيل**: https://manned.org/xz.

- ضغط ملف باستخدام xz:

```
xz {{path/to/file}}
```

- فك ضغط ملف XZ:

```
xz {{[-d|--decompress]}} {{path/to/file.xz}}
```

- ضغط ملف باستخدام خوارزمية lzma:

```
xz {{[-F|--format]}} lzma {{path/to/file}}
```

- فك ضغط ملف LZMA:

```
xz {{[-d|--decompress]}} {{[-F|--format]}} lzma {{path/to/
file.lzma}}
```

- فك ضغط ملف وكتابته إلى المخرجات القياسية (stdout) (يتضمن `--keep`):

```
xz {{[-d|--decompress]}} {{[-c|--stdout]}} {{path/to/
file.xz}}
```

- ضغط ملف بدون حذف النسخة الأصلية:

```
xz {{[-k|--keep]}} {{path/to/file}}
```

- ضغط ملف باستخدام أسرع ضغط:

```
xz -0 {{path/to/file}}
```

- ضغط ملف باستخدام أكفأ ضغط:

```
xz -9 {{path/to/file}}
```

# yt-dlp

**تّفرع من** youtube-dl **مع ميزات إضافية وتحسينات**.

**لتحميل الفيديوهات من** Youtube **ومواقع أخرى**.

**أنظر أيضاً**: `ytfzf`.

**لمزيد من التفاصيل**: https://github.com/yt-dlp/yt-dlp.

- لتحميل فيديو أو قائمة تشغيل (مع الأعدادات الأفتراضية):

```
yt-dlp "{{https://www.youtube.com/watch?v=oHg5SJYRHA0}}"
```

- عرض جميع الصيغ المتوفرة للتحميل من الفيديو:

```
yt-dlp {{[-F|--list-formats]}} "{{https://www.youtube.com/
watch?v=oHg5SJYRHA0}}"
```

- تحميل فيديو أو قائمة تشغيل بأفضل جودة MP4 متاح (الأفتراضي: "bv*+ba/b"):

```
yt-dlp {{[-f|--format]}} "{{bv*[ext=mp4]+ba[ext=m4a]/
b[ext=mp4]}}" "{{https://www.youtube.com/watch?
v=oHg5SJYRHA0}}"
```

- أستخراج الصوت فقط من الفيديو (يلزم توفر ffmpeg أو ffprobe):

```
yt-dlp {{[-x|--extract-audio]}} "{{https://www.youtube.com/
watch?v=oHg5SJYRHA0}}"
```

- لتحديد صيغة الصوت وجودته من الصوت المستخرج (بين 0 (الأفضل) و 10 (الأسوء) الأفتراضي = 5):

```
yt-dlp {{[-x|--extract-audio]}} --audio-format {{mp3}} --
audio-quality {{0}} "{{https://www.youtube.com/watch?
v=oHg5SJYRHA0}}"
```

- تحميل فقط العناصر الثانية والرابعة والخامسة والسادسة والأخيرة من قائمة التشغيل (مع العلم أنّ العنصر الأول يُحسب برقم 1، وليس 0):

```
yt-dlp {{[-I|--playlist-items]}} 2,4:6,-1 "{{https://
youtube.com/playlist?list=PLbzoR-
pLrL6pTJfLQ3UwtB-3V4fimdqnA}}"
```

- تحميل كل قوائم التشغيل من قناة اليويتوب/المستخدم, مع حفظ كل قائمة تشغيل في مجلد منفصل:

```
yt-dlp {{[-o|--output]}} "{{%(uploader)s/%(playlist)s/%
(playlist_index)s - %(title)s.%(ext)s}}" "{{https://
www.youtube.com/user/TheLinuxFoundation/playlists}}"
```

- تحميل دورة من Udemy مع حفظ كل فصل في مجلد منفصل:

```
yt-dlp {{[-u|--username]}} {{user}} {{[-p|--password]}}
{{password}} {{[-P|--paths]}} "{{path/to/directory}}" {{[-
o|--output]}} "{{%(playlist)s/%(chapter_number)s - %
(chapter)s/%(title)s.%(ext)s}}" "{{https://www.udemy.com/
java-tutorial}}"
```

# zip

**أداة لضغط وحزم الملفات في أرشيف** Zip.

**انظر أيضًا**: `unzip`.

**لمزيد من التفاصيل**: https://manned.org/zip.

- إضافة ملفات/مجلدات إلى أرشيف محدد ([r]بشكل متكرر):

```
zip {{[-r|--recurse-paths]}} {{path/to/compressed.zip}} {{path/to/file_or_directory1 path/to/file_or_directory2 ...}}
```

- إزالة ملفات/مجلدات من أرشيف محدد ([d]حذف):

```
zip {{[-d|--delete]}} {{path/to/compressed.zip}} {{path/to/file_or_directory1 path/to/file_or_directory2 ...}}
```

- أرشفة ملفات/مجلدات مع [x]استثناء عناصر معينة:

```
zip {{[-r|--recurse-paths]}} {{path/to/compressed.zip}} {{path/to/file_or_directory1 path/to/file_or_directory2 ...}} {{[-x|--exclude]}} {{path/to/excluded_files_or_directories}}
```

- أرشفة ملفات/مجلدات مع مستوى ضغط محدد (0 - الأقل، 9 - الأعلى):

```
zip {{[-r|--recurse-paths]}} -{{0..9}} {{path/to/compressed.zip}} {{path/to/file_or_directory1 path/to/file_or_directory2 ...}}
```

- إنشاء أرشيف [e]مشفر باستخدام كلمة مرور محددة:

```
zip {{[-re|--recurse-paths --encrypt]}} {{path/to/compressed.zip}} {{path/to/file_or_directory1 path/to/file_or_directory2 ...}}
```

- أرشفة ملفات/مجلدات في أرشيف Zip مقسم[s] إلى أجزاء متعددة (مثل أجزاء بحجم 3 جيجابايت):

```
zip {{[-rs|--recurse-paths --split-size]}} {{3g}} {{path/to/compressed.zip}} {{path/to/file_or_directory1 path/to/file_or_directory2 ...}}
```

- طباعة محتويات أرشيف محدد:

```
zip {{[-sf|--split-size --freshen]}} {{path/to/compressed.zip}}
```

# Linux

# apt-get

**أداة إدارة الحزم لديبيان وأوبونتو.**

**ابحث عن الحزم باستخدام `apt-cache`.**

**لمزيد من التفاصيل**: https://manned.org/apt-get.8.

- تحديث قائمة الحزم الموجودة وإصداراتها (يوصى بتشغيله قبل أي أمر آخر `apt-get`):

  ```
  apt-get update
  ```

- تثبيت حزمة معينة، أو تحديثها إلى آخر إصدار متوفر:

  ```
  apt-get install {{package}}
  ```

- إزالة حزمة معينة:

  ```
  apt-get remove {{package}}
  ```

- إزالة حزمة معينة وملفات الإعدادات الخاصة بها:

  ```
  apt-get purge {{package}}
  ```

- تطوير جميع الحزم المثبتة إلى أجدد الإصدارات المتوفرة:

  ```
  apt-get upgrade
  ```

- تنظيف المستودع المحلي (.deb) - إزالة ملفات الحزم من التنزيلات المعطلة التي لم يعد من الممكن تنزيلها:

  ```
  apt-get autoclean
  ```

- إزالة جميع الحزم التي لم تعد مطلوبة:

  ```
  apt-get autoremove
  ```

- تطوير الحزم المثبتة (مثل `upgrade`)، ولكن تقوم بإزالة الحزم القديمة وتثبيت حزم إضافية لتلبية التوابع الجديدة:

  ```
  apt-get dist-upgrade
  ```

# apt

**أداة لإدارة الحزم للتوزيعات القائمة على ديبيان.**

**عند الاستخدام الفعال في إصدارات أوبونتو** 16.04 **وما بعده** `apt-get` **بديل لـ.**

**للاطّلاع على الأوامر المكافئة في مديري الحزم الآخرين، انظر**: https://wiki.archlinux.org/title/Pacman/Rosetta.

**لمزيد من التفاصيل**: https://manned.org/apt.8.

- تحديث قائمة الحزم الموجودة وإصداراتها (يوصى بتشغيله قبل أي أمر `apt` آخر):

  ```
  sudo apt update
  ```

- البحث عن حزمة بالاسم أو الوصف:

  ```
  apt search {{package}}
  ```

- البحث عن الحزم بالاسم فقط (يدعم استخدام wildcards مثل *):

  ```
  apt list {{package}}
  ```

- عرض معلومات تفصيلية حول حزمة معينة:

  ```
  apt show {{package}}
  ```

- تثبيت حزمة جديدة، أو تحديثها إلى آخر إصدار متوفر:

  ```
  sudo apt install {{package}}
  ```

- إزالة حزمة معينة (استخدم `purge` لحذف ملفات الإعدادات الخاصة بالحزمة):

  ```
  sudo apt remove {{package}}
  ```

- تحديث جميع الحزم المثبتة إلى أجدد الإصدارات المتوفرة:

  ```
  sudo apt upgrade
  ```

- إظهار قائمة جميع الحزم المثبتة:

  ```
  apt list {{[-i|--installed]}}
  ```

# blkid

**يعرض جميع الأقسام الخاصة بالقرص الصلب ومعرّفاتها الفريدة** (UUID).

**لمزيد من التفاصيل**: https://manned.org/blkid.

- عرض جميع الأقسام:

  ```
  sudo blkid
  ```

- عرض جميع الأقسام في جدول، بما في ذلك نقاط التحميل الحالية:

  ```
  sudo blkid {{-o|--output}} list
  ```

# df

**عرض نظرة عامة على استخدام مساحة القرص لنظام الملفات.**

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/df-invocation.html.

- عرض جميع أنظمة الملفات واستخدامها للقرص:

```
df
```

- عرض جميع أنظمة الملفات واستخدامها للقرص بصيغة قابلة للقراءة البشرية:

```
df {{[-h|--human-readable]}}
```

- عرض نظام الملفات واستخدامه للقرص الذي يحتوي على ملف أو مُجَلَّد معين:

```
df {{path/to/file_or_directory}}
```

- تضمين إحصائيات حول عدد العقد المتاحة (inodes):

```
df {{[-i|--inodes]}}
```

- عرض أنظمة الملفات مع استثناء أنواع محددة:

```
df {{[-x|--exclude-type]}} {{squashfs}} {{[-x|--exclude-type]}} {{tmpfs}}
```

- عرض أنواع أنظمة الملفات:

```
df {{[-T|--print-type]}}
```

# dialog

**عرض نافذة حوار في الطرفية** (Terminal).

**لمزيد من التفاصيل**: https://manned.org/dialog.

- عرض رسالة:

```
dialog --msgbox "{{Message}}" {{height}} {{width}}
```

- مطالبة المستخدم بإدخال نص:

```
dialog --inputbox "{{Enter text:}}" {{8}} {{40}}
2>{{output.txt}}
```

- مطالبة المستخدم بسؤال بنعم/لا:

```
dialog --yesno "{{Continue?}}" {{7}} {{40}}
```

# dockerd

**هي عملية مستمرة تعمل في الخلفية تبدأها لتتحكم في حاويات الدوكر.**

**لمزيد من التفاصيل**: https://docs.docker.com/reference/cli/dockerd/.

- قم بتشغيل دوكر في الخلفية:

  ```
  dockerd
  ```

- قم بتشغيل دوكر في الخلفية واجعله يستمع علي منفذ معين (يونكس وبروتوكول ضبط الإرسال):

  ```
  dockerd --host unix:///{{path/to/tmp.sock}} --host tcp://{{ip}}
  ```

- قم بتشغيل دوكر في الخلفية برقم عملية معين:

  ```
  dockerd --pidfile {{path/to/pid_file}}
  ```

- قم بتشغيل دوكر في وضع التصحيح واكتشاف الأخطاء:

  ```
  dockerd --debug
  ```

- قم بتشغيل دوكر وحدد له مستوي سجل معين:

  ```
  dockerd --log-level {{debug|info|warn|error|fatal}}
  ```

# fprintd

**خدمة إدارة بصمات الأصابع**.

**لمزيد من التفاصيل**: https://fprint.freedesktop.org/.

- عرض صفحة المساعدة لـ `fprintd`:

  ```
  man fprintd
  ```

# getcap

**أمر لعرض اسم وصلاحيات كل ملف محدد.**

**لمزيد من التفاصيل**: https://manned.org/getcap.

- الحصول على الصلاحيات للملفات المحددة:

```
getcap {{path/to/file1 path/to/file2 ...}}
```

- الحصول على الصلاحيات لجميع الملفات داخل المجلدات المحددة بشكل (Recursive) متكرر:

```
getcap -r {{path/to/directory1 path/to/directory2 ...}}
```

- عرض جميع الإدخالات التي تم البحث عنها حتى لو لم يتم تعيين أي صلاحيات:

```
getcap -v {{path/to/file1 path/to/file2 ...}}
```

# gpasswd

**إدارة `/etc/group` و `/etc/gshadow`.**

**لمزيد من التفاصيل**: https://manned.org/gpasswd.

- عرّف مديرين المجموعة المسماة:

  ```
  sudo gpasswd {{[-A|--administrators]}} {{user1,user2}} {{group}}
  ```

- عين أعضاء المجموعة المسماة:

  ```
  sudo gpasswd {{[-M|--members]}} {{user1,user2}} {{group}}
  ```

- إنشئ رقم سري للمجموعة المسماة:

  ```
  gpasswd {{group}}
  ```

- أضف عضو إلي المجموعة المسماة:

  ```
  gpasswd {{[-a|--add]}} {{user}} {{group}}
  ```

- إحذف عضو من المجموعة المسماة:

  ```
  gpasswd {{[-d|--delete]}} {{user}} {{group}}
  ```

# head

**عرض الجزء الأول من الملفات.**

**لمزيد من التفاصيل**: https://www.gnu.org/software/coreutils/manual/html_node/head-invocation.html.

- عرض أول عدد معين من الأسطر من ملف:

```
head {{[-n|--lines]}} {{count}} {{path/to/file}}
```

- عرض أول عدد معين من البايتات من ملف:

```
head {{[-c|--bytes]}} {{count}} {{path/to/file}}
```

- عرض كل شيء باستثناء آخر عدد معين من الأسطر من ملف:

```
head {{[-n|--lines]}} -{{count}} {{path/to/file}}
```

- عرض كل شيء باستثناء آخر عدد معين من البايتات من ملف:

```
head {{[-c|--bytes]}} -{{count}} {{path/to/file}}
```

# hostnamectl

**الحصول على اسم المضيف أو تعيينه لجهاز الكمبيوتر.**

**لمزيد من التفاصيل**: https://manned.org/hostnamectl.

- الحصول على اسم المضيف لجهاز الكمبيوتر:

```
hostnamectl
```

- تعيين اسم المضيف لجهاز الكمبيوتر:

```
sudo hostnamectl set-hostname "{{hostname}}"
```

- تعيين اسم مضيف مُنَظَّم لجهاز الكمبيوتر:

```
sudo hostnamectl set-hostname --static
"{{hostname.example.com}}" && sudo hostnamectl set-hostname
--pretty "{{hostname}}"
```

- إعادة تعيين اسم المضيف إلى قيمته الافتراضية:

```
sudo hostnamectl set-hostname --pretty ""
```

# ip route list

**أمر فرعي لعرض جدول التوجيه في نظام الشبكة.**

**لمزيد من التفاصيل**: https://manned.org/ip-route.

- عرض جدول التوجيه الرئيسي:

  ```
  ip {{[r|route]}} {{[l|list]}}
  ```

- عرض جدول التوجيه الرئيسي (نفس المثال الأول):

  ```
  ip {{[r|route]}} {{[l|list]}} {{[t|table]}} {{main|254}}
  ```

- عرض جدول التوجيه المحلي:

  ```
  ip {{[r|route]}} {{[l|list]}} {{[t|table]}} {{local|255}}
  ```

- عرض جميع جداول التوجيه:

  ```
  ip {{[r|route]}} {{[l|list]}} {{[t|table]}} {{all|unspec|0}}
  ```

- عرض المسارات الخاصة بواجهة معينة فقط:

  ```
  ip {{[r|route]}} {{[l|list]}} dev {{ethX}}
  ```

- عرض المسارات داخل نطاق معين:

  ```
  ip {{[r|route]}} {{[l|list]}} {{[s|scope]}} link
  ```

- عرض ذاكرة التخزين المؤقت للمسارات:

  ```
  ip {{[r|route]}} {{[l|list]}} {{[c|cache]}}
  ```

- عرض مسارات IPv6 أو IPv4 فقط:

  ```
  ip {{-6|-4}} {{[r|route]}}
  ```

# ip route show

**هذا الأمر هو اسم مستعار لـ `ip route list`.**

- إعرض التوثيقات للأمر الأصلي:

  ```
  tldr ip route list
  ```

# kill

**يرسل إشارة إلى عملية** (Process)**، وعادةً ما يكون ذلك متعلقًا بإيقاف العملية**.

**بواسطة العملية لتنفيذ** SIGSTOP**و** SIGKILL **يمكن اعتراض جميع الإشارات باستثناء خروج نظيف**.

**لمزيد من التفاصيل**: https://manned.org/kill.

- إنهاء برنامج باستخدام الإشارة الافتراضية SIGTERM (إنهاء):

```
kill {{process_id}}
```

- عرض قيم الإشارات وأسمائها المقابلة (يجب استخدامها بدون البادئة `SIG`). قد تختلف الخيارات المتاحة حسب تنفيذ `kill`:

```
kill {{-l|-L|--table}}
```

- إنهاء وظيفة (Job) تعمل في الخلفية:

```
kill %{{job_id}}
```

- إنهاء برنامج باستخدام الإشارة SIGHUP (قطع الاتصال). العديد من الخدمات (Daemons) ستقوم بإعادة التحميل بدلاً من الإنهاء:

```
kill -{{1|HUP}} {{process_id}}
```

- إنهاء برنامج باستخدام الإشارة SIGINT (المقاطعة). يتم تنفيذ ذلك عادةً عند ضغط المستخدم على `<Ctrl c>`:

```
kill -{{2|INT}} {{process_id}}
```

- إرسال إشارة إلى نظام التشغيل لإنهاء برنامج فورًا (لن تتاح له فرصة لمعالجة الإشارة):

```
kill -{{9|KILL}} {{process_id}}
```

- إرسال إشارة إلى نظام التشغيل لإيقاف برنامج مؤقتًا حتى يتم استقبال إشارة SIGCONT ("المتابعة"):

```
kill -{{17|STOP}} {{process_id}}
```

- إرسال إشارة `SIGUSR1` إلى جميع العمليات التي تملك معرف المجموعة (GID) المحدد:

```
kill -{{SIGUSR1}} -{{group_id}}
```

# pacman

**أداة مدير الحزم لنظام** Arch Linux.

**انظر أيضًا**: `pacman-sync`, `pacman-remove`, `pacman-query`, `pacman-upgrade`, `pacman-files`, `pacman-database`, `pacman-deptest`, `pacman-key`, `pacman-mirrors`.

**لأوامر مكافئة في مديري الحزم الآخرين، انظر** https://wiki.archlinux.org/title/Pacman/Rosetta.

**لمزيد من التفاصيل**: https://manned.org/pacman.8.

- مزامنة وتحديث جميع الحزم:

  ```
  sudo pacman -Syu
  ```

- تثبيت حزمة جديدة:

  ```
  sudo pacman -S {{package}}
  ```

- إزالة حزمة والتبعيات الخاصة بها:

  ```
  sudo pacman -Rs {{package}}
  ```

- البحث في قاعدة بيانات الحزم عن تعبير نمطي (Regular Expresssion) أو كلمة مفتاحية:

  ```
  pacman -Ss "{{search_pattern}}"
  ```

- البحث في قاعدة البيانات عن الحزم التي تحتوي على ملف محدد:

  ```
  pacman -F "{{file_name}}"
  ```

- عرض الحزم المثبتة بشكل صريح (تم تثبيتها يدويًا بواسطة المستخدم) مع إصداراتها:

  ```
  pacman -Qe
  ```

- عرض الحزم اليتيمة (المثبتة كـ تبعيات ولكنها غير مطلوبة من قبل أي حزمة):

  ```
  pacman -Qtdq
  ```

- تفريغ ذاكرة التخزين المؤقت بالكامل لـ `pacman`:

  ```
  sudo pacman -Scc
  ```

# rkhunter

**يبحث عن برامج روتكيت والبرمجيات الخبيثة.**

**لمزيد من التفاصيل**: https://manned.org/rkhunter.

- فحص النظام للبحث عن برامج روتكيت والبرمجيات الخبيثة:

  ```
  sudo rkhunter --check
  ```

- تحديث rkhunter:

  ```
  sudo rkhunter --update
  ```

- عرض جميع الاختبارات المتاحة:

  ```
  sudo rkhunter --list
  ```

- عرض الإصدار:

  ```
  sudo rkhunter --versioncheck
  ```

- عرض المساعدة:

  ```
  sudo rkhunter --help
  ```

# setcap

**تعيين الصلاحيات لملف محدد.**

**انظر أيضًا**: **`getcap`**.

**لمزيد من التفاصيل**: https://manned.org/setcap.

- تعيين الصلاحية `cap_net_raw` (لاستخدام مآخذ RAW و PACKET) لملف معين:

```
setcap '{{cap_net_raw}}' {{path/to/file}}
```

- تعيين عدة صلاحيات على ملف (`ep` تعني "مفعلة ومسموح بها" خلف الصلاحية):

```
setcap '{{cap_dac_read_search,cap_sys_tty_config+ep}}' {{path/to/file}}
```

- إزالة جميع الصلاحيات من ملف:

```
setcap -r {{path/to/file}}
```

- التحقق من أن الصلاحيات المحددة مرتبطة حاليًا بالملف المحدد:

```
setcap -v '{{cap_net_raw}}' {{path/to/file}}
```

- لتعيين صلاحيات الملف لاستخدامها `-n root_uid` يمكن استخدام المعامل الاختياري فقط في نطاق مستخدم معين مع هذا المعرّف الجذري:

```
setcap -n {{root_uid}} '{{cap_net_admin}}' {{path/to/file}}
```

# SS

**أداة للتحقيق في المقابس** (Sockets) **و المنافذ**.

**لمزيد من التفاصيل**: https://manned.org/ss.8.

- عرض جميع مقابس TCP/UDP/RAW/UNIX:

```
ss {{[-a|--all]}} {{-t|-u|-w|-x}}
```

- تصفية مقابس TCP حسب الحالات، تضمين/استبعاد:

```
ss {{state|exclude}} {{bucket|big|connected|
synchronized|...}}
```

- عرض جميع مقابس TCP المتصلة بمنفذ HTTPS (443) المحلي:

```
ss {{[-t|--tcp]}} src :{{443}}
```

- عرض جميع مقابس TCP 8080 التي تستمع على المنفذ المحلي:

```
ss {{[-lt|--listening --tcp]}} src :{{8080}}
```

- عرض جميع مقابس TCP المتصلة بمنفذ SSH الخارجي مع العمليات:

```
ss {{[-pt|--processes --tcp]}} dst :{{ssh}}
```

- عرض جميع مقابس UDP المتصلة بمنفذ مصدر ومنفذ وجهة محددين:

```
ss {{[-u|--udp]}} 'sport == :{{source_port}} and dport == :
{{destination_port}}'
```

- عرض جميع مقابس TCP IPv4 192.168.0.0/16 المتصلة محليًا على الشبكة الفرعية:

```
ss {{[-4t|--ipv4 --tcp]}} src {{192.168/16}}
```

- إنهاء اتصال مقبس IPv4 أو IPv6 8080 والمنفذ 192.168.1.17 مع عنوان الوجهة:

```
ss {{[-K|--kill]}} dst {{192.168.1.17}} dport = {{8080}}
```

# systemctl

**التحكم في مدير نظام** systemd **والخدمات**.

**لمزيد من التفاصيل**: https://www.freedesktop.org/software/systemd/man/systemctl.html.

- عرض جميع الخدمات قيد التشغيل:

```
systemctl status
```

- عرض الوحدات الفاشلة:

```
systemctl --failed
```

- بدء/إيقاف/إعادة تشغيل/إعادة تحميل/عرض حالة خدمة:

```
systemctl {{start|stop|restart|reload|status}} {{unit}}
```

- تمكين/تعطيل وحدة ليتم تشغيلها عند بدء تشغيل النظام:

```
systemctl {{enable|disable}} {{unit}}
```

- إعادة تحميل systemd والبحث عن وحدات جديدة أو متغيرة:

```
systemctl daemon-reload
```

- التحقق مما إذا كانت الوحدة نشطة/مُمكّنة/فاشلة:

```
systemctl {{is-active|is-enabled|is-failed}} {{unit}}
```

- عرض جميع وحدات الخدمة/المقبس/التركيب التلقائي مع التصفية حسب الحالة (قيد التشغيل/فاشلة):

```
systemctl list-units {{[-t|--type]}} {{service|socket|automount}} --state {{failed|running}}
```

- عرض محتويات ومسار ملف الوحدة:

```
systemctl cat {{unit}}
```

# tor

**تمكين الاتصال المجهول عبر شبكة تور.**

**لمزيد من التفاصيل**: https://manned.org/tor.

- الاتصال بشبكة تور:

  ```
  tor
  ```

- عرض إعدادات تور:

  ```
  tor --config
  ```

- التحقق من حالة تور:

  ```
  tor --status
  ```

- التشغيل كعميل فقط:

  ```
  tor --client
  ```

- التشغيل كنقطة تمرير:

  ```
  tor --relay
  ```

- التشغيل كجسر:

  ```
  tor --bridge
  ```

- التشغيل كخدمة مخفية:

  ```
  tor --hidden-service
  ```

# ufw

**جدار حماية بسيط.**

**تهدف إلى تسهيل تكوين جدار الحماية `iptables` واجهة أمامية لـ.**

**لمزيد من التفاصيل**: https://wiki.ubuntu.com/UncomplicatedFirewall.

- تفعيل ufw:

  ```
  ufw enable
  ```

- تعطيل ufw:

  ```
  ufw disable
  ```

- عرض قواعد ufw مع أرقامها:

  ```
  ufw status numbered
  ```

- السماح بحركة المرور الواردة على المنفذ 5432 مع تعليق يحدد الخدمة:

  ```
  ufw allow {{5432}} comment "{{Service}}"
  ```

- السماح بحركة مرور TCP إلى أي عنوان على هذا الجهاز 192.168.0.4 فقط من العنوان، على المنفذ 22:

  ```
  ufw allow proto {{tcp}} from {{192.168.0.4}} to {{any}} port {{22}}
  ```

- منع حركة المرور على المنفذ 80 على هذا الجهاز:

  ```
  ufw deny {{80}}
  ```

- منع جميع حركة مرور UDP 8412:8500 إلى المنافذ في النطاق:

  ```
  ufw deny proto {{udp}} from {{any}} to {{any}} port {{8412:8500}}
  ```

- حذف قاعدة معينة. يمكن الحصول على رقم القاعدة من أمر `ufw status numbered`:

  ```
  ufw delete {{rule_number}}
  ```

# vnstat

**مراقبة حركة مرور الشبكة عن طريق وحدة تحكم.**

**لمزيد من التفاصيل**: https://manned.org/vnstat.

- عرض ملخص حركة المرور لجميع الواجهات:

  ```
  vnstat
  ```

- عرض ملخص حركة المرور لواجهة شبكة محددة:

  ```
  vnstat -i {{network_interface}}
  ```

- عرض إحصائيات مباشرة لواجهة شبكة محددة:

  ```
  vnstat -l -i {{network_interface}}
  ```

- عرض إحصائيات حركة المرور على أساس كل ساعة خلال آخر 24 ساعة باستخدام رسم بياني شريطي:

  ```
  vnstat -hg
  ```

- قياس وعرض متوسط حركة المرور لمدة 30 ثانية:

  ```
  vnstat -tr {{30}}
  ```

# xclip

**أداة معالجة لحافظة** x11، **تشبه إلي حد ما** `xsel`.

**بالإضافة إلي حافظة النظام،** x **تتعامل مع الحافظة الأولية والثانوية لـ** (`<Ctrl c>`/ `<Ctrl v>`).

**لمزيد من التفاصيل**: https://manned.org/xclip.

- إنسخ ناتج الخرج من أمر إلي حافظة x11 الأولية:

```
echo 123 | xclip
```

- إنسخ ناتج الخرج من أمر إلي الحافظة المختارة:

```
echo 123 | xclip -selection {{primary|secondary|clipboard}}
```

- إنسخ ناتج الخرج من أمر إلي حافظة النظام، باستخدام الصيغة المختصرة:

```
echo 123 | xclip -sel clip
```

- إنسخ محتوي ملف إلي حافظة النظام:

```
xclip -sel clip {{input_file.txt}}
```

- إنسخ محتوي صورة بصيغة PNG عن (يمكن أن تستخدم في أي برنامج إلي حافظة النظام طريق لصق):

```
xclip -sel clip -t image/png {{input_file.png}}
```

- إنسخ إدخال المستخدم في الطرفية أو الكونسول إلي حافظة النظام:

```
xclip -i
```

- إلصق محتوي حافظة x11 الأولية إلي الطرفية أو الكونسول:

```
xclip -o
```

- إلصق محتوي حافظة النظام إلي الطرفية أو الكونسول:

```
xclip -o -sel clip
```

# Windows

# del

**حذف ملف واحد او مجموعه من الملفات.**

**وهو الاسم المستعار للامر `Remove-Item`.**

**هذه الوثائق تستند إلى إصدار سطر الأوامر (`cmd`) من `del`.**

**لمزيد من التفاصيل**: https://learn.microsoft.com/windows-server/administration/windows-commands/del.

- اعرض التوثيقات للأمر الأصلي:

  ```
  tldr remove-item
  ```

- حذف ملف او أكثر او حذف التطابق مع أنماط:

  ```
  del {{file_pattern1 file_pattern2 ...}}
  ```

- التأكيد قبل الحذف:

  ```
  del {{file_pattern}} /p
  ```

- حذف ملفات القراءة فقط:

  ```
  del {{file_pattern}} /f
  ```

- حذف الملفات الموجودة في المجلد الحالي وأيضًا الملفات الفرعية:

  ```
  del {{file_pattern}} /s
  ```

- لا تطلب التأكيد عند حذف الملفات بناءً على محدد عام:

  ```
  del {{file_pattern}} /q
  ```

- عرض المساعدة وقائمة السمات المتاحة:

  ```
  del /?
  ```

- حذف ملف اعتمادا على محدد معين:

  ```
  del {{file_pattern}} /a {{attribute}}
  ```

# iwr

**هذا الأمر هو اسم مستعار لـ `invoke-webrequest`.**

- إعرض التوثيقات للأمر الأصلي:

  ```
  tldr invoke-webrequest
  ```

# pwsh where

**هذا الأمر هو اسم مستعار لـ `Where-Object`.**

- إعرض التوثيقات للأمر الأصلي:

  ```
  tldr Where-Object
  ```

# sls

**هذا الأمر هو اسم مستعار لـ `Select-String`.**

- إعرض التوثيقات للأمر الأصلي:

  ```
  tldr select-string
  ```