hledger Plain-text accounting

START HERE

`hledger add`
Enter transactions

`hledger bs`
Balance sheet

`hledger bal`
Balances per account

`hledger print`
Show journal entries

`hledger reg <acct>`
Postings with running total

`hledger check`
Find errors in the journal

`hledger is -M`
Monthly income statement

`hledger -f <file> bal`
Use a specific journal

Bank CSV downloaded → `import` → **Journal** plain text → `query + report` → **Reports** bal, reg, is, bs → `-O csv/json` → **Other tools** spreadsheet, scripts

CONCEPTS

Everything is read from a plain text journal; hledger never edits it except add and import. Set LEDGER_FILE to point at yours (default ~/.hledger.journal).

Account types come from top-level names (assets, liabilities, equity, revenues, expenses) or from account directives; bs and is rely on them.

JOURNAL FORMAT

A transaction

```
2026-10-22 * Grocer | weekly
shop  ; food:
    expenses:food          $45.20
    assets:bank:checking
# amount inferred
```

2026-10-22 * desc
Date, optional status (* cleared, ! pending), description
Postings are indented; two or more spaces separate account and amount.

; note tag:value
Comment with a tag
Tags are name:value pairs in comments, comma-separated.

= $100
Balance assertion after a posting
Fails the check if the account balance differs.

10 AAPL @ $150
Unit price (cost)

10 AAPL @@ $1500
Total price (cost)

(account) $5
Virtual posting, not balanced

account assets:bank
Declare an account

commodity $1,000.00
Declare commodity format

include *<file>*
Include another journal

~ monthly rent
Periodic transaction (budget, forecast)
Used with --forecast or balance --budget.

= expenses:food
Auto posting rule, applied with --auto

P 2026-10-01 EUR $1.10
Market price declaration

 # comment
Comment line

QUERIES

acct:*<regex>*
Match account names
A bare word is also an account match.

desc:*<regex>*
Match description

amt:'>100'
Match amount

date:2026-10
Match a date range

tag:*<name>*=*<value>*
Match a tag

status:*
Cleared only (also -C, -P, -U)

not:*<query>*
Negate a query

cur:*<regex>*
Match commodity

depth:2
Limit account depth

expr:'*<a>* **or** *<b>*'
Boolean query

BALANCE REPORTS

hledger bal
Account balances

hledger bal *<query>*
Balances for matching accounts

hledger bal -t
Show as indented tree

hledger bal -N
Hide the total line

hledger bal --depth 2
Roll up to depth 2

hledger bal -2
Same as --depth 2

hledger bal -M
One column per month

hledger bal -H
Historical balances, include earlier postings

hledger bal -T
Add a row total column

hledger bal -A
Add an average column

hledger bal -S
Sort by amount

hledger bal --cumulative
Running totals across periods

hledger bal --budget
Actual vs periodic budget goals

hledger bal -%
Percent of column total

hledger bal --invert
Flip signs

BALANCE REPORTS (cont.)

hledger bs
Balance sheet

hledger is
Income statement

hledger cf
Cash flow

REGISTER AND PRINT

hledger reg *<acct>*
Postings with running total

hledger reg -M *<acct>*
Monthly totals for an account

hledger areg *<acct>*
One line per transaction in an account

hledger print
Show transactions in journal format

hledger print -x
Show inferred amounts explicitly

hledger print -m *<desc>*
Most similar recent transaction

hledger print -O csv
Export transactions as CSV

TIME AND VALUATION

-p 'last month'
Report period
Also 'this year', '2026Q3', 'from 2026-01 to 2026-04'.

-b 2026-01-01 -e 2026-04-01
Start (inclusive) and end (exclusive)

-D / -W / -M / -Q / -Y
Daily to yearly interval

-B
Convert to cost

-V
Convert to market value at period end

--forecast
Include periodic transactions

INPUT AND ENTRY

hledger -f *<file> <cmd>*
Read a specific file (repeatable)

hledger add
Enter transactions interactively
Finish with . or Ctrl+D.

hledger import *<file.csv>*
Add new CSV records to the journal
Uses <file>.csv.rules and skips already imported dates.

INPUT AND ENTRY (cont.)

hledger import --dry-run *<file.csv>*
Preview what would be imported

hledger -f *<file.csv>* **--rules** *<rules>* **print**
Convert CSV with a rules file

CSV RULES

bank.csv.rules

```
skip 1
fields date, description,
amount
date-format %d/%m/%Y
currency $
account1 assets:bank:checking
if grocer
  account2 expenses:food
```

skip *<n>*
Skip header lines

fields *<a>*, *<b>*, ...
Name CSV columns

account2 *<acct>*
Set the other account

if *<regex>*
Conditional block of rules

CHECKS AND INFO

hledger check
Run basic checks

hledger check -s
Strict: also accounts and commodities declared

hledger accounts
List accounts

hledger payees
List payees

hledger tags
List tag names

hledger stats
Journal summary

hledger *<cmd>* **-h**
Options of a command

-O csv / -O json
Machine readable output

RECIPES

Monthly spending by category

```
hledger bal expenses --depth 2
-M -S -T
```

Import a bank export

```
hledger import --dry-run
bank.csv
hledger import bank.csv
hledger check -s
```

Net worth over time

```
hledger bal assets liabilities
-H -Y
hledger bs -M --tree
```

Export to a spreadsheet

```
hledger reg expenses:food -p
2026 -O csv -o food.csv
```