# Radare2 Cheatsheet

## Basic commands

| | |
|---|---|
| `s` | seek to a different address |
| `b` | change block size |
| `q` | quit |
| `pd/px/p8` | print disasm/hexdump/bytes |
| `pf` | print formatted |
| `f name=addr` | set flag to the given address |
| `-j` | enter the Javascript repl |
| `?*~str` | filter commands with the given string |

## Command prefixes

| | |
|---|---|
| `.` | interpret output |
| `:` | execute io command |
| `#` | comment |
| `` ` `` | avoid evaluation of special chars |
| `N` | repeat n times, being `N` a number |

## Command suffixes

| | |
|---|---|
| `j` | json |
| `q` | quiet (simplest output) |
| `*` | radare2 commands |
| `,` | table format |
| `?` | help for the command |

## Command modifiers

| | |
|---|---|
| `>` | redirect output to file or `$internalfile` |
| `@` | temporal seek |
| `@@` | repeat on every flag |

## Binary information

| | |
|---|---|
| `ie` | entrypoint |
| `is/ii/iE` | symbols/imports/exports |

## Patching

| | |
|---|---|
| `wx` | write hexpairs (wv for endian values) |
| `wa` | write assembly |
| `wo` | write in block |
| `wtf` | write to file (use wtff for @@) |

## Search

| | |
|---|---|
| `/ lib` | find 'lib' string |
| `/x 9090` | hexpairs |
| `/ad ret` | instructions with *ret* |
| `/m` | search for known magic headers |
| `w lob @@/ lib` | write lob on every lib |

## Analysis and xrefs

| | |
|---|---|
| `af` | analyze function |
| `aa` | analyze all program (aaa, aaaa, ...) |
| `afn/afvn` | rename a function/variable |
| `afl` | list functions |
| `axt` | list xrefs to given address |

## Disassembly

| | |
|---|---|
| `pdf/pdr` | disassemble function/recursive |
| `pdc` | pseudo-decompilation (see pdd/pdg) |
| `pd/pi` | print disassembly/instructions |
| `Cd` | define as data |
| `CC` | add a comment in code |

## Emulation (ESIL)

| | |
|---|---|
| `aeim` | initialize emulation registers + stack |
| `ae` | emulate ESIL expression |
| `aes` | step into (see ds, but also aesu?) |
| `aer` | for register manipulation (see dr) |

## Debugging
**(`r2 -d bin`)**

| | |
|---|---|
| `db` | set/manage breakpoints |
| `dbt` | backtrace |
| `ds/dso` | single step/step over |
| `dr` | get/set register values (drr for telescoped) |
| `doo/ood` | restart process |

## Visual mode
**(`V` for visual, `v` for panels, `!` to toggle)**

| | |
|---|---|
| `pP` | rotate modes (<tab> for submodes) |
| `s/S` | perform step/step over |
| `b` | browse (flags, config, classes, symbols, ...) |
| `.` | seek to entrypoint |
| `i/A` | insert mode for hexa/write assembly |
| `V_` | hud mode to seek flags while typing |
| `Vd1` | visual bit editor |
| `n/N` | seek to next/previous *scr.nkey* thing |
| `x/X` | view xrefs/refs |
| `hjkl` | move cur (`HJKL` for faster movement) |

## Graph
**(`VV` comand, `agfv`)**

| | |
|---|---|
| `agn/age/aggv` | custom handmade graphs |
| `t/f/u` | follow true/false branch, undo |

## Settings
**(`e` command)**

| | |
|---|---|
| `edit` | use *cfg.editor* with `~/.radare2rc` |
| `anal.hasnext` | consider code is sequential |
| `asm.bytes` | show/hide bytes in disasm |
| `asm.emu/emu.str` | emulation analysis/show strings |
| `bin.relocs.apply` | apply relocs |
| `scr.color=n` | enable colors, where n = 0,1,2,3 |
| `search.in` | define search boundaries |