# New developer setup

Run this on your first day to check that your machine can build mdsh. It only checks and builds into a temporary directory; it doesn't install anything. Send the report to your onboarding buddy when something is missing:

```console
$ mdsh --pdf setup.pdf examples/runbooks/onboarding.md
```

## Tools

Each tool you need, the version you have, and the oldest that works.

```sh {#tools out=markdown}
ver() { "$@" 2>/dev/null | head -n 1 | grep -Eo '[0-9]+\.[0-9]+(\.[0-9]+)?' | head -n
1; }
row() {
  name="$1" need="$2"; shift 2
  if command -v "$name" >/dev/null 2>&1; then
    have="$(ver "$@")"
    printf '%s\t%s\t%s\tok\n' "$name" "${have:-unknown}" "$need"
  else
    printf '%s\tmissing\t%s\t**install it**\n' "$name" "$need"
  fi
}
ROWS="$(printf 'tool\tyou have\tneeds\tstatus')"
for t in "git 2.30 git --version" "go 1.24 go version" "make 3.81 make --version" "cur
l 7.0 curl --version"; do
  ROWS="$ROWS
$(row $t)"
done
MISSING="$(printf '%s\n' "$ROWS" | grep -c 'install it' || true)"
printf '%s\n' "$ROWS" | md_table
```

| tool | you have | needs | status |
|---|---|---|---|
| git | 2.43.0 | 2.30 | ok |
| go | 1.27.0 | 1.24 | ok |
| make | 4.3 | 3.81 | ok |
| curl | 8.5.0 | 7.0 | ok |

## Git

Commits need a name and an email. This only says whether they're set, so the report doesn't carry your address around.

```sh {#git-identity}
for key in user.name user.email; do
  if [ -n "$(git config --get "$key" 2>/dev/null)" ]; then
    echo "$key: set"
  else
    echo "$key: not set, run: git config --global $key ..."
```

```
  fi
done
```

```
user.name: set
user.email: set
```

## SSH key

GitHub over SSH needs a key. This lists the public keys you have, by type only.

```sh {#ssh}
found=0
for k in "$HOME"/.ssh/id_*.pub; do
  [ -e "$k" ] || continue
  echo "$(basename "$k"): $(cut -d' ' -f1 "$k")"
  found=1
done
[ "$found" = 1 ] || echo "no SSH key yet: ssh-keygen -t ed25519"
```

```
no SSH key yet: ssh-keygen -t ed25519
```

## Build

Build mdsh from this checkout into a temporary directory and run it.

```sh {#build timeout=5m}
cd "$(git rev-parse --show-toplevel 2>/dev/null || echo ../..)"
BIN="$(mktemp -d)"
go build -o "$BIN/mdsh" ./cmd/mdsh && "$BIN/mdsh" --version >/dev/null && echo "mdsh b
uilds and runs"
```

```
mdsh builds and runs
```

## Next steps

1. Read Getting started.
2. Run `go test -short ./...` before your first pull request.
3. Open `examples/hello.md` with `mdsh -i` and try a block of your own.