Yarn Node package manager

CONCEPTS

Yarn 4 is set per project: yarn set version stable writes the version into package.json (packageManager), and Corepack picks it up.

package.json ranges you want
↓ resolve
yarn.lock exact versions
↓ fetch
Cache zipped packages
↓ link
Install PnP or node_modules

By default Yarn 4 uses Plug'n'Play and writes no node_modules; set nodeLinker: node-modules in .yarnrc.yml if a tool needs the folder.

INSTALL AND ADD

`yarn install`
Install all dependencies

`yarn add <pkg>`
Add a dependency

`yarn add -D <pkg>`
Add a dev dependency

`yarn add <pkg>@<version>`
Add a specific version or tag

`yarn install --immutable`
Fail if the lockfile would change

REMOVE AND UPDATE

`yarn remove <pkg>`
Remove a dependency

`yarn up <pkg>`
Upgrade a package

`yarn up "*"`
Upgrade everything

`yarn up -i`
Pick upgrades interactively

`yarn set version stable`
Pin the latest stable Yarn

RUN SCRIPTS

`yarn run <script>`
Run a package script

RUN SCRIPTS (cont.)

`yarn <script>`
Shorthand for yarn run

`yarn run <bin>`
Run a dependency's binary

`yarn dlx <pkg> [args]`
Fetch and run without installing

`yarn init`
Create a package.json

INSPECT

`yarn info <pkg>`
Show info about a package

`yarn why <pkg>`
Show why a package is installed

`yarn npm audit`
Check for known vulnerabilities

WORKSPACES

`yarn workspaces list`
List all workspaces

`yarn workspace <name> <cmd>`
Run a command in one workspace

`yarn workspace <name> add <pkg>`
Add a dependency to one workspace

`yarn workspaces foreach -A run <script>`
Run in every workspace

`yarn workspaces foreach -Apt run <script>`
Parallel, dependencies first

`yarn workspaces foreach --since=<ref> -A run <script>`
Only workspaces changed since a ref

Workspaces in package.json

```
{
  "private": true,
  "workspaces": ["packages/*"]
}
```

CONFIG, PATCHING AND PUBLISHING

`yarn config set <key> <value>`
Write a project setting

`yarn npm publish`
Publish the workspace