

Initialize React Project with Vite

[documentation](#) [documentation](#)

Create a New Project

[documentation](#) [documentation](#)

```
export VITE_PROJECT="vite-project"
```

```
yarn create vite -t react --no-interactive "$VITE_PROJECT"
```

Cleanup Vite Project Template

```
yarn install --immutable --immutable-cache  
yarn remove @types/react @types/react-dom  
yarn remove oxlint  
rm .oxlintrc.json  
sed -i '/"lint":/d' package.json  
sed -i 's@/src/main.jsx@/app/main.jsx@' index.html  
mv src app
```

Setup Custom Import Aliases

[documentation](#)

```
{  
  "imports": {  
    "#app/": "./app/",  
    "#assets/": "./assets/",  
    "#lib/": "./lib/",  
    "#routes/": "./routes/"  
  }  
}
```

```
}
```

Use Latest Version of Yarn

[documentation](#) [documentation](#)

```
yarn set version stable  
yarn install
```

Enable Node Modules Linking with Local Cache

```
# SPDX-FileCopyrightText: None  
#  
# SPDX-License-Identifier: 0BSD  
  
enableGlobalCache: false  
  
# [!] NPM Supply Chain Attack Mitigation.  
enableScripts: false  
  
enableTelemetry: false  
  
nodeLinker: node-modules
```

```
# SPDX-FileCopyrightText: None  
#  
# SPDX-License-Identifier: 0BSD  
  
.yarn/*  
!.yarn/cache/  
!.yarn/releases/  
yarn-debug.log*  
yarn-error.log*  
/node_modules/
```

Unify Chores via Makefile

```
# SPDX-FileCopyrightText: None  
# SPDX-License-Identifier: 0BSD
```

```
JQ := $(shell command -v jq 2>/dev/null)
ifeq ($(JQ),)
$(error jq is required but not installed)
endif

DISTNAME := $(shell jq -r .name package.json)
VERSION := $(shell jq -r .version package.json)
DISTDIR := dist

HOST := 127.27.0.1
PORT := 5173

.PHONY: dev
dev:
  yarn dev --host="$(HOST)" --port="$(PORT)"

.PHONY: install
install:
  yarn install --immutable --immutable-cache

.PHONY: update
update:
  yarn up

.PHONY: build
build:
  yarn build --outDir="$(DISTDIR)"

.PHONY: dist
dist: build
  tar --directory="$(DISTDIR)" --create --xz --file="$(DISTNAME)-$(VERSION).tar.xz" .

.PHONY: test
test:
  yarn test

.PHONY: clean
clean:
  -rm --recursive "$(DISTDIR)"
  -rm "$(DISTNAME)-$(VERSION).tar.xz"

.PHONY: distclean
distclean: clean
  git clean --force -Xd
```