

10 Feb 2024

- create a github repo
- push the code
- npm init

For test we use jest

→ After performing the above steps we get a package.json, which is configuration file of npm

→ Bundler → It is a framework which bundles up the code to make it more optimize.

ex- web-pick (React)

② Parcel ← We are using Parcel

③ next

npm i parcel

npm i -D parcel

Dev dependency

Package.json

Package-lock.json

↳ It keeps the registry of approx versions of packages

↳ It keeps the registry of exact version used during development

↳ also has a hash value to verify it

exact

"^2.8.3"

↳ this will automatically update the minor version if their is any.

"~2.8.3"

tilde

↳ Now it can be upgraded to a major update if their comes any

* Error

The above statement was giving error so you used yarn to download ~~the~~ or install the parcel

npm parcel index.html

↳ executing a package

→ cdn links are not a good way to use react because it makes the network call which is time taken.

→ And also the version keep changing so it better to have a single version.

as a library is want to add many elements

→ Install React

npm i react

not

↳ now we are installing it as a web dependency.

Now we don't need the cdn links

But we need to import React from "react"

→ import ReactDOM from "react-dom/client"

⇒ Now import and export cannot work in normal JS file you need to specify that it is a module now

→ `<script type = "module" src = "/App.js"></script>`

→ [Parceljs.org/docs](https://parceljs.org/docs)

Parcel → ① Parcel uses HMR → Hot module Replacement
it updates of webpage real time

② → It also uses a File Watching Algorithm
build using C++

③ → It is also ~~cache~~ cache so that the next build took faster.
in the folder .parcel-cache

④ → It also do's image optimization.

⑤ → It also do minification when we do production code

⑥ → It also do compressing of code to make
~~the~~ the react fast.

⑦ → It also bundles up the code tightly together.

⑧ → Consistent Hashing

9) → Code splitting

10) → Differential ~~Build~~ Bundling ← Support older browsers

11) → Error handling

12) → Tree shaking algorithm ← Remove un-used code

Now to run code in production mode

→ `npx parcel build index.html`

↳ But before that remove in package.json remove "main": "App.js"
This was for node to see which was the entry point
but now no need of that

→ After creating the build Parcel compresses the whole code in mainly 3 files

- 1) Index.html
- 2) Index.css
- 3) Index.js

}

And the code is totally changed and compressed.

And this is stored in regenerative .dist folder

while pushing code on github

→ We do not push regenerative code like /dist, parcel-cache etc

→ ~~browserlist.dev/docs~~
Browserlist → This is framework that make sure our app works on all the specified versions of the browsers

→ In package.json

create → "browserlist": [

"last 2 versions"

]

↖ This means last 2 versions of every browser exists.

→ It creates a whole different bundle for old browser but that made the code large and slow.

⋮