Chapter 02 - Assignment - Igniting our App

1. What is `NPM`?
   NPM is a package manager which helps to download/install packages in our React App.

2. What is `Parcel/Webpack`? Why do we need it?
   Parcel/Webpack is a bundler that performs multiple actions on our code to make that code ready for deployment into the live server. Bundler helps to bundle up the entire javascript code into a single file.

3. What is `.parcel-cache`?
   .parcel-cache helps to cache large-size assets such as media content so that on the next run document renders the web page in lesser time than the previous time.

4. What is `npx` ?
   Npx helps to execute a node package. For example, if we want to execute parcel node package then we need to write command as npx parcel <entry point file >

5. What is difference between `dependencies` vs `devDependencies`?
   Dependencies are node packages that would be used by the react for all the environments. devDevependencies are node packages mentioned in package.json that would be used specifically for development build.

6. What is Tree Shaking?
   Tree Shaking or Dead code elimination is the process by which bundler such as Webpack and Parcel removed unused code which are being added by using static/dynamic import and export with the help of a process called Scope Hoisting.

7. What is Hot Module Replacement?
   This is a way by which parcel updates the app on the run time and did not require refreshing the page.

8. List down your favourite 5 superpowers of Parcel and describe any 3 of them in your own words.

   A. Hot Reloading - This is a way by which parcel updates the code without refreshing the page.
   B. Lazy Mode - By using npm parcel <entry point> --lazy, parcel makes sure the web app does not load the entire page at once and load the pages once requested
   C. Cache - Parcel cache all the built files in the parcel-cache folder and update only the changed portion and not the entire app
   D. HTTPS
   E. File Watcher - An algorithm written in C++ to detect the changes made in files or even node modules which is using Parcel as a bundler
   F. Auto Install - Parcel install automatically the dev dependencies required in the parcel project. Based on the lock json parcel uses the package manager to download

files. For package-lock.json, parcel uses npm to install and download dependencies.
H. Minification - Parcel minifies html, css, javascript, SVGs out of the box.
I. Tree Shaking - Parcel helps to remove dead code for each module. This is applicable for static and dynamic imports as well.
J. Image Optimization

9. What is `.gitignore`? What should we add and not add into it?
   .gitignore is the file name which mentions the files that should not be pushed to git. We should not add files which are generated automatically such as node modules, dist folders.

10. What is the difference between `package.json` and `package-lock.json` ?
    Package.json is a json type file which lists all the app details and dependencies packages that are being used in the app for development environment and production environment as well. It contains the versions of the packages which are not exact.
    Package-lock.json shows all the specific versions of the packages that are being used in the web app and all the transitive dependencies packages as well.

11. Why should I not modify `package-lock.json`?
    Package-lock.json contains all the packages exact version details with its transitive dependencies. Modifying it might cause the web app not to work as expected.

12. What is `node_modules` ? Is it a good idea to push that on git?
    Node-modules is the auto generated folder which contains all the dependencies downloaded with exact version that has been mentioned in package-loack.json.
    It is not a good idea to push node-modules since the size is huge and node modeles will be generated once our application runs on the server.

13. What is the `dist` folder?
    Dist folder contains all the minified files that are required to render the web app.

14. What is `browserlists`?
    Browserlists help us to make sure to run the application for the mentioned browser details in the form of a query.

15. Read about dif bundlers: vite, webpack, parcel ✅
16. Read about: ^ - caret and ~ - tilda
    ^ - supports all versions without affecting the major versions. Eg - ^1.2.3 will support all the future patches <2.0.0
    ~ - supports all versions without affecting the minor versions. Eg - ^1.2.3 will support all the future patches <1.3.0

17. Read about Script types in html (MDN Docs)✅

# Project Assignment: - In your existing project ✅

- - intialize `npm` into your repo ✅
- - install `react` and `react-dom` ✅
- - remove CDN links of react ✅
- - install parcel ✅
- - ignite your app with parcel ✅
- - add scripts for “start” and “build” with parcel commands ✅
- - add `.gitignore` file ✅
- - add `browserlists` ✅
- - build a production version of your code using `parcel build` ✅

**Code added in https://github.com/sumitWebDev/ReactNotes**

# References

- Creating your own create-react-app
- Parcel Documentation
- Parcel on Production
- BrowsersList