react

install and configure nodejs environment

```
# OS: linux (ubuntu)
# install node:

# installs nvm (Node Version Manager)
> curl -o https://raw.githubusercontent.com/nvm-sh/nvm/v0.40.0/install.sh | bash

# download and install Node.js (you may need to restart the terminal)
> nvm install 22

# verifies the right Node.js version is in the environment
> node -v # should print `v22.11.0`

# verifies the right npm version is in the environment
> npm -v # should print `10.9.0`


# OS: windows
# install node:

# installs fnm (Fast Node Manager)
> winget install Schniz.fnm

# configure fnm environment
> fnm env --use-on-cd | Out-String | Invoke-Expression

# download and install Node.js
> fnm use --install-if-missing 22

# verifies the right Node.js version is in the environment
> node -v # should print `v22.11.0`

# verifies the right npm version is in the environment
> npm -v # should print `10.9.0`
```

install yarn package manager

```
# install yarn on linux
> sudo npm install -g yarn

# install yarn on windows
> npm install -g yarn
```

create a new react application

```
# create a new application
# > npx create-react-app <application name>
> npx create-react-app app1

# run a react application
# NOTE: please make sure that you are in the application directory
# > cd app1
# > npm start
# > npm run start
> yarn start

# unit test the react application
# > npm test
# > npm run test
> yarn test

# build the application (compile the source into JS deployable package)
# > npm build
# > npm run build
> yarn build
```

react application directory structure

- build (directory)
  - contains the compiled version of the current application
  - the code inside this directory needs to get deployed on the server
  - by default this directory does exist
  - when yarn build command is fired, the whole react application source code gets compiled into the build directory

- node_modules (directory)
  - contains the application dependencies (node modules)
  - since the application is using react library, this directory contains the react and its dependencies
- public (directory)
  - contains the public files of the website
  - contains
    - index.html
      - is the only web page in the react application
      - remaining JS code gets compiled into JS modules and get imported inside the index.html file
      - this is the file which starts when the react application starts
    - favicon.ico
      - icon file which is rendered as tab icon
    - logo files (logo192 and logo512)
      - multiple resolutions of website logo
    - manifest.json
      - contains the application resources metadata
- .gitignore (file)
  - non-react file used to ignore the files or directories while committing the version in git repository
- package.json (file)
  - used to define the application dependencies
  - contains
    - two types of dependencies
      - dependencies: list of dependencies which will be needed for running the application and hence will be compiled while building the application
      - devDependencies: list of dependencies which will NOT be needed for running the application, but needed to develop the application and hence will not be compiled while building the application
    - scripts
      - the commands which can be executed using yarn or npm command
    - lint configuration
      - lint is the process for fixing the syntax issues
      - uses JS linter to find and fix the syntax issues
- package-lock.json (file)
  - contains the list of node modules installed in node_modules directory
  - contains the versions of all modules installed
- README.md (file)
  - contains the read me information about the project
- src (directory)
  - index.js (file)
    - startup file which loads the react application
    - which creates React Root element and loads the first component of the application
  - app.js (file)
    - default or first component, the application (index.js) will load
    - the default name of first component is: App
  - app.test.js (file)
    - contains unit tests for App component present in app.js file
  - App.css (file)
    - contains CSS rules for the first component
  - reportWebVitals.js
    - used to get the web vital report (monitoring)
  - setupTests.js
    - used to configure the unit testing of react application

react application startup

- execute command yarn start or npm run start
- react subsystem
  - compiles the application (JS code) into bundle.js
  - loads the script bundle.js into public/index.html file
    -