**Next js**

1) It is a library of react.

2) To install next app npx create-next-app hello-world.

3) npm install / yarn install.

4) npm run dev / yarn dev.

5) To know next version npm show next version.

**Next JS Routing**

1) Routing is based on page folder structure.

2) For nested routing we nested folder structure in page folder.

3) Dynamic route can be created by using square bracket to a page name.

4) Catch all route – add three dots inside bracket to create a catch all route , It is helpful when we need to define different url for same page or we want to pass multiple parameter that is dynamically change.

**404 page**

1. To create 404 page we just need to add 404.js page under page folder and we can change the text or template of this page.

Note – From next 13 or greater this version we make file like not-found.js  
These above are for global use and for different page different not found we can make it like that under folder for the specific page where we want to use it we make another folder name […not-found]/page.js.

**Navigation**

1. For link pages we use Link from next/link.
2. useRouter hook’s router.push method to navigate programmatically.
3. We can also use router.replace for return to desired page.

**Pre-rendering and Data Fetching**

There are 2 types of rendering

1)Client side rendering

2)Server side rendering

Difference between javascript and typescript

In JavaScript we don’t need to define variable data type , instead of this in typescript we need to define data type of the variable.

Server not understand typescript code directly, firstly it convert to JavaScript then it will run

**Difference between variable and state**

Variable not re rendering data but state can

We can make component inside another component and use it

To show parameter in page or we want to use it we need to define “use client” on the top of the page then we can use the parameters.

**Dynamic route**

To make dynamic route we make folder and name it like […foldername] then we create page under this folder and to get all segment we do it like that.  
{params.foldername[0]}, {params.foldername[1]} .. and so on.

**Middleware**

* Middleware is a part that is execute before the routing requests.
* We make file middleware.js under src if we have src otherwise we make it under root folder
* For specific route we do it like that

export *const* config = {

    matcher:["/blog","/blog/:path\*"]

}

* For single route

import { NextResponse } from "next/server";

export *function* middleware(*request*){

    if(*request*.nextUrl.pathname!="/login"){ for single route

        return NextResponse.redirect(new *URL*("/login",*request*.url))

    }

}

* For single we also do like that

export *const* config = {

    matcher:"/blog  
}

**Client Side Rendering & PreRendering**

* In network after load the page when we see then when it load all html code it means it is preRendering means server side rendering

**Styles**

1. Global CSS
2. Inline CSS
3. Module CSS

**Meta data or tags**

1. To generate meta tags we just add generateMetadata.

**Build**

The code that is written in next,react,angular js they are not run directly , for that we to generate a code that is understand by browser that is able to execute on browser is called build.

We can also make customize build .

Build types are :-

1. Production build
2. Development Build

**Production Build :-** Production build are those that is run on different url. It is fast . Code minified like js files is .min.js. to run production build npm run build then run npm start.

**Development Build :-** Development build are those that is run on different url. It is slow. Code is not minified because it will be complicated to debug. That’s why here we use not minified code.

When we do npm run dev  
then it create development build

**Export static HTML**

To export static html we need to run npm run build and before it we need to define output:”export” in nextconfig.js after that when we run the above command then it generate an out name folder under this some html files are included.

**Environment variable**

To know predefined environment variable just console it :-

console.log(process.env)

To make configuration file  
under src folder make a folder config under this folder make file named constants.js   
Now to use its variable just import this filer where you and call there variable.

**API Routes**