បង្កើត blog react vite tailwindcss - Documentation

**Installing React.js with Vite and Tailwind CSS: A Quick and Efficient Guide**

<https://blog.100jsprojects.com/post/installing-react-js-with-vite-and-tailwind-css--a-quick-and-efficient-guide>

React tailwind responsive layout នៅក្នុង App.jsx ដាក់កូដដូចខាងក្រោម

import Navbar from "./components/Navbar";

function App() {

return (

<>

<div className="px-4 md:px-8 lg:px-16 xl:px-32 2xl:px-64">

{/\* NAVBAR \*/}

<Navbar />

{/\* BRADCRUMP \*/}

{/\* INTRODUCTION \*/}

{/\* FEATURE POSTS \*/}

{/\* POST LIST \*/}

</div>

</>

);

}

export default App;

React tailwind responsive Navbar: នៅក្នុង Navbar.jsx ដាក់កូដដូចខាងក្រោម

const Navbar = () => {

return (

<div className="w-full h-16 md:h-20 flex items-center justify-between">

{/\* LOGO \*/}

<div className="flex items-center gap-4 text-2xl font-bold">

<img src="/logo.png" className="w-8 h-8" alt="" />

<span>mkrablog.</span>

</div>

{/\* MOBILE MENU \*/}

<div className="md:hidden">M</div>

{/\* DESKTOP MENU \*/}

<div className="hidden md:flex">D</div>

</div>

);

};

export default Navbar;

React tailwind Hamburger Menu Design នៅក្នុង Navbar.jsx បង្កើត Hamburger button យើងចង់ បានពេលចុចលើ ☰ ចេញ X​ បិទបើក Menu ដោយប្រើ useState ReactHook

{/\* MOBILE MENU \*/}

<div className="md:hidden">

{/\* HAMBURGER BUTTON \*/}

<div

className="cursor-pointer text-4xl"

onClick={() => setopen((prev) => !prev)}

>

{open ? "x" : "☰"}

</div>

</div>

បង្កើត Menu Link List

{/\* HAMBURGER BUTTON \*/}

<div

className="cursor-pointer text-4xl"

onClick={() => setopen((prev) => !prev)}

>

{open ? "x" : "☰"}

</div>

{/\* MENU LINK LIST \*/}

<div

className={`w-full h-screen flex flex-col items-center justify-center absolute top-16 bg-[#e6e6ff] transition-all ease-in-out ${

open ? "-right-0" : "-right-[100%]"

}`}

>

Menu

</div>

បង្កើត Desktop Menu

{/\* DESKTOP MENU \*/}

<div className="hidden md:flex items-center gap-8 xl:gap-12 font-medium">

<a href="/">Home</a>

<a href="/">Trending</a>

<a href="/">Most Popular</a>

<a href="/">About</a>

<a href="/">

<button className="py-2 px-4 rounded-3xl bg-blue-800 text-white">

Login 👋

</button>

</a>

</div>

React Image Optimization

ពាក់ព័ន្ធបញ្ហារូបភាពឬវីដេអូដែលមានទំហំធំ ព្រោះប្រូជិករបស់យើងនឹងត្រូវផ្ទុករូបភាពឬវីដេអូច្រើន ដែលត្រូវ មានកន្លែងផ្ទុកត្រឹមត្រូវ យើងមិនគួរ Upload រក្សាទុកនៅក្នុង ទីតាំងប្រូជិកទេធ្វើប្រូជិករបស់យើងដំណើរការ យឺត ដូចនេះ យើងចំាបាច់ត្រូវប្រើ Cloud Media ឬអាចប្រើ CDN ប៉ុន្តែក្នុងប្រូជិកនេះយើងប្រើ imagekit.io ដោយដំឡើង npm install --save imagekitio-react ។ Signup ដោយប្រើ google account ។

React Router Dom

ចូលលីងនេះ <https://reactrouter.com/start/library/installation> ដើម្បីដំឡើង និងconfig ។ បង្កើត Page ដែលត្រូវការរួច កំណត់និងsetup នៅក្នុង main.jsx

import { StrictMode } from "react";

import { createRoot } from "react-dom/client";

import { BrowserRouter, Routes, Route } from "react-router";

import "./index.css";

import App from "./App.jsx";

import HomePage from "./pages/HomePage.jsx";

import AboutPage from "./pages/AboutPage.jsx";

import LoginPage from "./pages/LoginPage.jsx";

import RegisterPage from "./pages/RegisterPage.jsx";

import PostListPage from "./pages/PostListPage.jsx";

import SinglePostPage from "./pages/SinglePostPage.jsx";

import PublishPost from "./pages/PublishPost.jsx";

createRoot(document.getElementById("root")).render(

<StrictMode>

<BrowserRouter>

<Routes>

<Route index element={<HomePage />} />

<Route path="about" element={<AboutPage />} />

<Route path="postlist" element={<PostListPage />} />

<Route path="singlepost" element={<SinglePostPage />} />

<Route path="publishpost" element={<PublishPost />} />

{/\* <Route element={<AuthLayout />}> \*/}

<Route path="login" element={<LoginPage />} />

<Route path="register" element={<RegisterPage />} />

{/\* </Route> \*/}

</Routes>

</BrowserRouter>

</StrictMode>

);

បន្ទាប់មកដើម្បី page នីមួយៗបង្ហាញលក្ខណៈ Dynamic ដែលប្រើ Navbar ត្រូវបង្កើត Layouts ដោយបង្កើត file ឈ្មោះ MainLayout.jsx using <Outlet />

import { Outlet } from "react-router";

import Navbar from "../components/Navbar";

const MainLayout = () => {

return (

<div className="px-4 md:px-8 lg:px-16 xl:px-32 2xl:px-64">

<Navbar />

<Outlet />

</div>

);

};

export default MainLayout;

បន្ទាប់នៅក្នុង Main.jsx

import { StrictMode } from "react";

import { createRoot } from "react-dom/client";

import { BrowserRouter, Routes, Route } from "react-router";

import "./index.css";

import HomePage from "./pages/HomePage.jsx";

import AboutPage from "./pages/AboutPage.jsx";

import LoginPage from "./pages/LoginPage.jsx";

import RegisterPage from "./pages/RegisterPage.jsx";

import PostListPage from "./pages/PostListPage.jsx";

import SinglePostPage from "./pages/SinglePostPage.jsx";

import PublishPost from "./pages/PublishPost.jsx";

import MainLayout from "./layouts/MainLayout.jsx";

createRoot(document.getElementById("root")).render(

<StrictMode>

<BrowserRouter>

<Routes>

<Route element={<MainLayout />}>

<Route index element={<HomePage />} />

<Route path="about" element={<AboutPage />} />

<Route path="postlist" element={<PostListPage />} />

<Route path="singlepost" element={<SinglePostPage />} />

<Route path="publishpost" element={<PublishPost />} />

{/\* <Route element={<AuthLayout />}> \*/}

<Route path="login" element={<LoginPage />} />

<Route path="register" element={<RegisterPage />} />

{/\* </Route> \*/}

</Route>

</Routes>

</BrowserRouter>

</StrictMode>

);

បន្ទាប់នៅក្នុង Navbar.jsx

import { useState } from "react";

import Image from "./Image";

import { Link } from "react-router";

const Navbar = () => {

const [open, setopen] = useState(false);

return (

<div className="w-full h-16 md:h-20 flex items-center justify-between">

{/\* LOGO \*/}

<Link to="/" className="flex items-center gap-4 text-2xl font-bold">

<Image src="logo.png" w={32} h={32} alt="mkrablog logo" />

<span>mkrablog.</span>

</Link>

{/\* MOBILE MENU \*/}

<div className="md:hidden">

{/\* HAMBURGER BUTTON \*/}

<div

className="cursor-pointer text-4xl"

onClick={() => setopen((prev) => !prev)}

>

{open ? "x" : "☰"}

</div>

{/\* MENU LINK LIST \*/}

<div

className={`w-full h-screen flex flex-col items-center justify-center gap-8 font-medium text-lg absolute top-16 bg-[#e6e6ff] transition-all ease-in-out ${

open ? "-right-0" : "-right-[100%]"

}`}

>

<Link to="/">Home</Link>

<Link to="/">Trending</Link>

<Link to="/">Most Popular</Link>

<Link to="/about">About</Link>

<Link to="/">

<button className="py-2 px-4 rounded-3xl bg-blue-800 text-white">

Login 👋

</button>

</Link>

</div>

</div>

{/\* DESKTOP MENU \*/}

<div className="hidden md:flex items-center gap-8 xl:gap-12 font-medium">

<Link to="/">Home</Link>

<Link to="/">Trending</Link>

<Link to="/">Most Popular</Link>

<Link to="/about">About</Link>

<Link to="/">

<button className="py-2 px-4 rounded-3xl bg-blue-800 text-white">

Login 👋

</button>

</Link>

</div>

</div>

);

};

export default Navbar;

បង្កើត User authentication ដោយប្រើ Clerk.com បង្កើត Clerk Account ដោយប្រើ google account បន្ទាប់មកអនុវត្តជំហានដែលបង្ហាញរហូតដល់ចប់ ។

HomePage design – Introduction section

HomePage design – MainCategories section

HomePage design – FeaturedPosts section

HomePage design – PostList section

សូមធ្វើការបញ្ជាក់ថា PostList រូបភាពនីមួយៗបើយើងInspect មើលទំហំគឺជាទំហំដើម ក្នុងករណីនេះបើ អ្នកប្រើប្រាស់ upload រូបភាពដែលមានទំហំធំនោះមិនល្អនោះទេក្នុងការប្រើទំហំដើម ដូចនេះយើងត្រូវតែ Resize វា។ ធ្វើដូចនេះបានយើងអាចប្រើ property របស់ imagekit.to នៅក្នុង Image.jsx component

transformation={[

{

width: w,

height: h,

},

]}

កូដក្នុង Image.jsx

import { IKImage } from "imagekitio-react";

const Image = ({ src, classname, w, h, alt }) => {

return (

<IKImage

urlEndpoint={import.meta.env.VITE\_IMAGEKIT\_URL\_ENDPOINT}

path={src}

className={classname}

loading="lazy"

lqip={{ active: true, quality: 20 }}

alt={alt}

width={w}

height={h}

transformation={[

{

width: w,

height: h,

},

]}

/>

);

};

export default Image;

ហើយកំណត់ទំហំ 735

<Image

src="postImg.jpeg"

classname="rounded-2xl object-cover"

w="735"

/>

ចុះធ្វើមិចយើងដឹងថាត្រូវកំណត់ទំហំនេះ បើយើងបង្រួមbrowser មកmdរូបភាពនឹងប្រើ maximum size យើងដឹងហើយថានៅក្នុងMainLayout.jsx យើងបានកំណត់px-4 ស្មើនឹង 16px ហើយទំហំscreen size md ធំបំផុត 767px បើ 767px-32px(16x2)=735px នេះហើយគឺជា maximum image size ។ បែបនេះធ្វើអោយ website យើងមានល្បឿន loading កាន់តែលឿន។