

vue2项目的打包与在github上的部署

1. 配置 vue.config.js

首先，确保你的项目正确配置了 `publicPath`，这样在部署到 GitHub Pages 时，所有资源的路径会指向正确的位置。在项目根目录下创建（如果没有的话）或修改 `vue.config.js` 文件，添加如下配置：

```
module.exports = {
  publicPath: process.env.NODE_ENV === 'production'
    ? '/<REPO_NAME>/' // 这里填入你的仓库名，例如 'todolist'
    : '/'
}
```

`<REPO_NAME>` 是你在 GitHub 上创建的仓库名（如 `todolist`）。

这段代码会确保在生产环境（即部署到 GitHub Pages 时），所有的资源路径会被正确设置。

2. 修改 package.json (添加 GitHub Pages 配置)

在 `package.json` 中，添加一个 `homepage` 字段，指向你最终托管项目的 URL，这样 Vue CLI 在构建时会知道部署路径：

```
"homepage": "https://<USERNAME>.github.io/<REPO_NAME>/"
```

- `<USERNAME>` 替换为你的 GitHub 用户名。
- `<REPO_NAME>` 替换为你的仓库名。

3. 安装 gh-pages 部署工具

`gh-pages` 是一个方便将构建好的文件部署到 GitHub Pages 的工具。你需要安装它作为开发依赖。

```
npm install --save-dev gh-pages
```

4. 添加部署脚本

在 `package.json` 文件中，修改 `scripts` 部分，添加一个 `deploy` 脚本，用来将构建后的 `dist/` 文件夹部署到 `gh-pages` 分支。

```
"scripts": {
  "build": "vue-cli-service build",
  "deploy": "gh-pages -d dist"
}
```

- `build` 脚本会将项目构建到 `dist/` 目录。
- `deploy` 脚本会将 `dist/` 目录的内容推送到 `gh-pages` 分支。

5. 构建项目

在终端中运行以下命令，生成打包后的文件：

```
npm run build
```

这将会在 `dist/` 目录下生成你的 Vue 项目的生产环境版本。

6. 部署到 GitHub Pages

运行 `npm run deploy` 将构建后的文件推送到 GitHub Pages：

```
npm run deploy
```

这个命令会使用 `gh-pages` 将 `dist/` 目录中的所有文件推送到 `gh-pages` 分支。GitHub Pages 会自动托管这个分支中的内容。

7. 配置 GitHub Pages

1. 进入你的 GitHub 仓库页面。
2. 打开 **Settings** 页面。
3. 滚动到 **Pages** 部分，设置 **Source** 为 `gh-pages` 分支，并选择 `/ (root)` 作为路径。

设置完成后，GitHub Pages 会自动将 `gh-pages` 分支中的内容托管为静态网站。

8. 访问部署的项目

几分钟后，你就可以通过以下 URL 访问你托管的 Vue 项目：

```
https://<USERNAME>.github.io/<REPO_NAME>/
```

- `<USERNAME>` 替换为你的 GitHub 用户名。
- `<REPO_NAME>` 替换为你的仓库名。

9. 自定义域名 (可选)

如果你有自定义域名，可以在 `dist/` 目录下创建一个名为 `CNAME` 的文件，并将你的域名写入其中，例如：

```
echo 'www.example.com' > dist/CNAME
```

然后，重新运行 `npm run deploy` 推送 `CNAME` 文件到 `gh-pages` 分支。

总结

1. 配置 `vue.config.js` 中的 `publicPath`。
2. 在 `package.json` 中添加 `homepage` 字段，指向 GitHub Pages URL。
3. 安装 `gh-pages` 工具并添加部署脚本。
4. 执行 `npm run build` 构建项目。

5. 执行 `npm run deploy` 将项目部署到 GitHub Pages。
6. 在 GitHub 仓库设置中配置 GitHub Pages。
7. 访问部署的 URL，查看效果。