

使用 `create-react-app` 一步步地创建一个 TypeScript 项目，并引入 antd。

`antd` 基于最新稳定版本的 TypeScript (`>=4.0.0`)，请确保项目中使用匹配的版本。

安装和初始化

请确保电脑上已经安装了最新版的 [yarn](#) 或者 [npm](#)。

使用 yarn 创建 [cra-template-typescript](#) 项目。

```
$ yarn create react-app antd-demo-ts --template typescript
```

如果你使用的是 npm（接下来我们都会用 yarn 作为例子，如果你习惯用 npm 也没问题）。

```
$ npx create-react-app antd-demo-ts --template typescript
```

然后我们进入项目并启动。

```
$ cd antd-demo-ts
$ yarn start
```

此时浏览器会访问 <http://localhost:3000/>，看到 `Welcome to React` 的界面就算成功了。

引入 antd

```
$ yarn add antd
```

修改 `src/App.tsx`，引入 antd 的按钮组件。

```
import React, { FC } from 'react';
import { Button } from 'antd';
import './App.css';

const App: FC = () => (
  <div className="App">
    <Button type="primary">Button</Button>
  </div>
);

export default App;
```

修改 `src/App.css`，在文件顶部引入 antd 的样式。

```
@import '~antd/dist/antd.css';
```

重新启动 `yarn start`，现在你应该能看到页面上已经有了 antd 的蓝色按钮组件，接下来就可以继续选用其他组件开发应用了。其他开发流程你可以参考 create-react-app 的[官方文档](#)。

antd 使用 TypeScript 书写并提供了完整的定义，你可以享受组件属性输入建议和定义检查的功能。

```
const App = () => (

Button

);  
export default App
```

Button 属性选择器提示：
dashed
default
ghost
link
primary

注意不要安装 `@types/antd`。

高级配置

这个例子在实际开发中还有一些优化的空间，比如无法进行主题配置。

此时我们需要对 create-react-app 的默认配置进行自定义，这里我们使用 [craco](#)（一个对 create-react-app 进行自定义配置的社区解决方案）。

现在我们安装 craco 并修改 `package.json` 里的 `scripts` 属性。

```
$ yarn add @craco/craco
```

```
/* package.json */  
"scripts": {  
  - "start": "react-scripts start",  
  - "build": "react-scripts build",  
  - "test": "react-scripts test",  
  + "start": "craco start",  
  + "build": "craco build",  
  + "test": "craco test",  
}
```

然后在项目根目录创建一个 `craco.config.js` 用于修改默认配置。

```
/* craco.config.js */  
module.exports = {  
  // ...  
};
```

自定义主题

按照 [配置主题](#) 的要求，自定义主题需要用到类似 [less-loader](#) 提供的 less 变量覆盖功能。我们可以引入 [craco-antd](#) 来帮助加载 less 样式和修改变量。

首先把 `src/App.css` 文件修改为 `src/App.less`，然后修改样式引用为 less 文件。

```
/* src/App.tsx */  
- import './App.css';  
+ import './App.less';
```

```
/* src/App.less */  
- @import '~antd/dist/antd.css';
```

然后安装 `craco-antd` 并修改 `craco.config.js` 文件如下。

```
$ yarn add craco-antd
```

```
const CracoAntDesignPlugin = require('craco-antd');  
  
module.exports = {  
  plugins: [  
    {  
      plugin: CracoAntDesignPlugin,  
      options: {  
        customizeTheme: {  
          '@primary-color': '#1DA57A',  
        },  
      },  
    },  
  ],  
};
```

这里利用了 [less-loader](#) 的 `modifyVars` 来进行主题配置，变量和其他配置方式可以参考 [配置主题](#) 文档。修改后重启 `yarn start`，如果看到一个绿色的按钮就说明配置成功了。

antd 内建了深色主题和紧凑主题，你可以参照 [使用暗色主题和紧凑主题](#) 进行接入。

同样，你可以使用 [react-app-rewired](#) 和 [customize-cra](#) 来自定义 `create-react-app` 的 `webpack` 配置。

其他方案

如果你已经按照 [在 create-react-app 中使用](#) 初始化了环境，可以参考官方文档里的 [Adding TypeScript](#) 配置 TypeScript 开发环境。

- [Create React apps \(with Typescript and antd\) with no build configuration](#)
- [react-app-rewire-typescript](#)
- [ts-import-plugin](#)
- [Migrating from create-react-app-typescript to Create React App](#)