TypeScript React Conversion

# Tổng quan:

Việc chuyển đổi TypeScript trong bất kỳ dự án React nào sử dụng JavaScript cũng có thể được chia thành 2 giai đoạn:

* Thêm trình biên dịch TypeScript (tsc) để biên dịch mã TS sang JS
* Chuyển đổi mã nguồn JS file sang mã nguồn TS file

# Add TypeScript compiler to build

* Cài đặt **TypeScript** (3.6.4 hoặc cao hơn)

npm install --save-dev typescript

* Lấy các tệp khai báo loại (file .d.ts) từ **@types** cho bất kỳ thư viện nào đang sử dụng. Đối với dự án này có **React, jest,** **react-router-dom,** **toastr** và **ReactDOM**.
* Nếu bạn đang sử dụng phiên bản cũ hơn của có **React, jest,** **react-router-dom,** **toastr** hay **ReactDOM** không tương thích với các file .**d.ts** mới nhất từ **​​@types**, bạn có thể chỉ định số phiên bản trong pack.json. (nếu không tương thích sẽ bị lỗi trong quá trình biên dịch)

1. "dependencies": {
2. "@types/jest": "24.0.19",
3. "@types/node": "12.11.1",
4. "@types/react": "16.9.7",
5. "@types/react-dom": "16.9.2",
6. "@types/react-router-dom": "^5.1.0",
7. "@types/toastr": "^2.1.37",
8. "axios": "^0.19.0",
9. "bootstrap": "^4.3.1",
10. "react": "^16.10.2",
11. "react-dom": "^16.10.2",
12. "react-router-dom": "^5.1.2",
13. "react-scripts": "3.2.0",
14. "toastr": "^2.1.4",
15. "typescript": "3.6.4"
16. },

# Configure TypeScript

Tiếp theo, cấu hình nguyên cảo bằng cách tạo ra một tập tin **tsconfig.json**

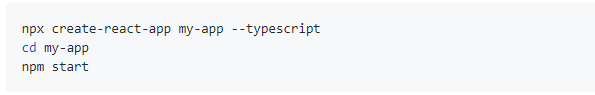
1. {
2. "compilerOptions": {
3. "target": "es5",
4. "lib": [
5. "dom",
6. "dom.iterable",
7. "esnext"
8. ],
9. "allowJs": **true**,
10. "skipLibCheck": **true**,
11. "esModuleInterop": **true**,
12. "allowSyntheticDefaultImports": **true**,
13. "strict": **true**,
14. "forceConsistentCasingInFileNames": **true**,
15. "module": "esnext",
16. "moduleResolution": "node",
17. "resolveJsonModule": **true**,
18. "isolatedModules": **true**,
19. "noEmit": **true**,
20. "jsx": "react"
21. },
22. "include": [
23. "src"
24. ]
25. }

# Set up build pipeline

## Sử dụng webpack.configure.js

Follow: [link](https://github.com/microsoft/TypeScript-React-Conversion-Guide#set-up-build-pipeline)

## Sử dụng React-create-app



# Transition from JS(X) to TS(X)

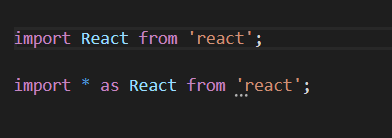
Các bước tối thiểu để chuyển đổi mã nguồn 1 file JS(x) sang TS(x)

1. Đổi đuôi file mở rộng từ .jsx sang .tsx
2. Xác định các đôi tượng trong file xử lý
3. Xác định phạm vi truy cập của các đối tượng
4. Thêm kiểu (type) cho các khai báo

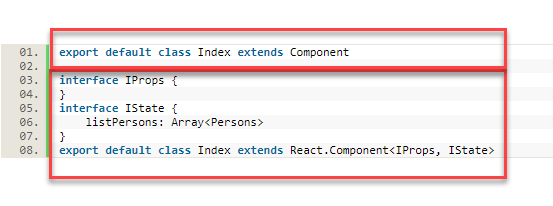
# Exemple Basic

Với file **index.component.js -> index.component.tsx**

* Ở dòng 1 import React from "react"; Thay đổi thành



* Ở dòng 5: export default class Index extends Component



* Càng nhiều thông tin loại được cung cấp cho **TypeScript**, việc kiểm tra loại của nó càng mạnh. Như một cách thực hành tốt nhất, chúng tôi khuyên bạn nên cung cấp **Kiểu** cho tất cả các khai báo. Chúng tôi sẽ lại sử dụng thành phần **Persons** làm ví dụ.

1. **export** **default** **class** Persons {
2. ID?: number;
3. FullName: string;
4. Address: string;
5. Age: number;
7. constructor(fullanme: string, address: string, age: number) {
8. **this**.FullName = fullanme;
9. **this**.Address = address;
10. **this**.Age = age;
11. }
12. }

