- Dịch 1 file typescript sang javascript: tsc filename.ts

- Chạy 1 file javascript: Có 2 cách

Cách 1: node filename.js

Cách 2: ts-node filename.ts (cần cài thư viện ts-node, dịch và chạy code trong 1 câu lệnh)

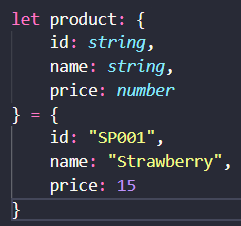
- type anotation chữ thường khác với chữ hoa, ví dụ :string khác với :String, 1 cái là dữ liệu nguyên thủy, 1 cái là dữ liệu k nguyên thủy.

- Nên dùng **viết thường**, k nên dùng viết hoa (<https://www.typescriptlang.org/docs/handbook/declaration-files/do-s-and-don-ts.html>)

- Đối với số, ngoài kiểu dữ liệu :number, còn có kiểu :bigint.

- Đối với array, nếu có nhiều kiểu dữ liệu thì có thể khai báo như mẫu sau courses: (string | number | boolean)[]

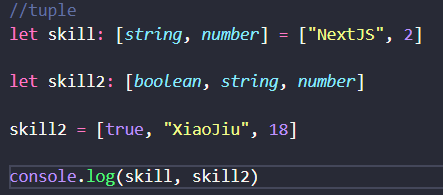
- Đối với object, có các kiểu anotation là :object, :{}. Nhưng ít dùng, thường dùng là kiểu



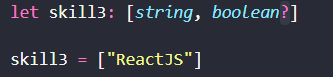
- Kiểu dữ liệu **tuple**, hoạt động tương tự array, nhưng có vài quy định bắt buộc như:

Phải khai báo trước số phần tử tuple

Phân biệt thứ tự từng phần tử tương ứng với kiểu dữ liệu.



- Tham số optional (có thể có hoặc k) trong kiểu dữ liệu tuple bằng cách thêm từ khóa ? ngay sau tên biến. Lưu ý: tham số Optional bắt buộc phải nằm cuối cùng.



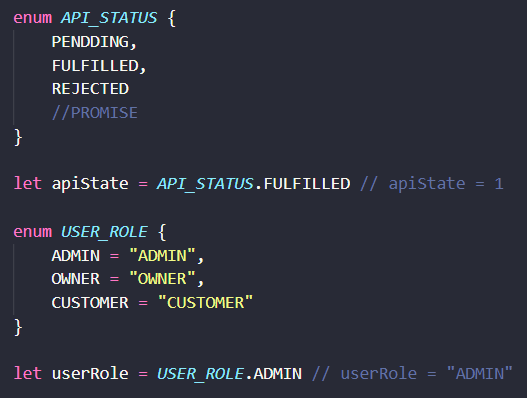
- **Enum** (enumerate: liệt kê, one by one) là 1 nhóm các giá trị hằng số. VD STATUS = [pending, success, failed]

- Từng giá trị enum sẽ được gán tương ứng với giá trị index.

- Có thể định nghĩa thêm các function (điểm hay hơn constant).

- Giúp thống nhất về syntax và logic của frontend & backend (java cũng có kiểu enum)

- Nếu không muốn lưu giá trị số, có thể định nghĩa giá trị.



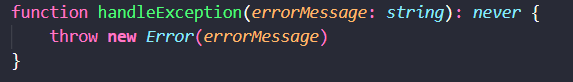
- Kiểu dữ liệu **any**, tương tự javascript, một biến sẽ có kiểu dữ liệu động tùy vào giá trị lưu trữ. Dùng khi không chắc dữ liệu trả về kiểu gì.

- Kiểu dữ liệu **void** thường dùng với function, trái ngược với kiểu any. Dùng khi function không cần return về giá trị gì.

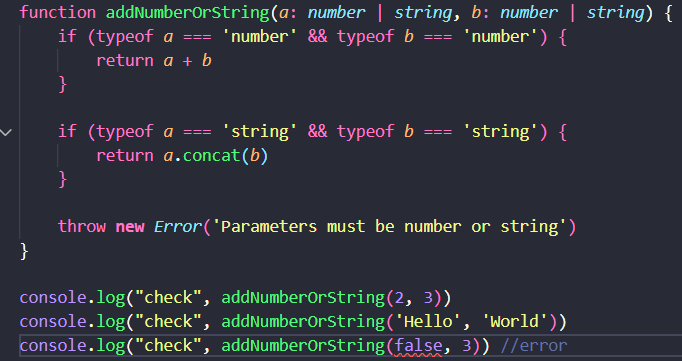
- Kiểu dữ liệu **never**, tương tự void, không bao giờ trả ra giá trị. Khi chắc chắn một điều gì đó không bao giờ xảy ra, thường dùng khi sử dụng với các thư viện.

- Thực chất kiểu void trong javascript vẫn trả về undefined, còn never thì không trả bất cứ giá trị gì.

- Kiểu never thường dùng với throw trong function.



- Kiểu dữ liệu **union**, cho phép 1 biến có nhiều hơn 1 kiểu dữ liệu.



- Kiểu **Aliases** cho phép tạo mới 1 kiểu type từ những type đã tồn tại (khắc phục nhược điểm của **union** nếu có nhiều kiểu dữ liệu sẽ gây rườm rà). Bắt đầu bằng một keyword “type” + tên định danh + các kiểu dữ liệu.

