**Some Easy React Notes**

**前言：**

**React is essentially a JavaScript based framework/library. Essentially the structure still relies on HTML code to define the view elements, like typically inside the “return ()” function. Then CSS is still being used to define the styles of named classes - HTML elements. JavaScript on top of this handles the logics and motions of the website. Nothing really changed, but improved!**

1. React has two API: Functional Component (Hook) V.S. Class component

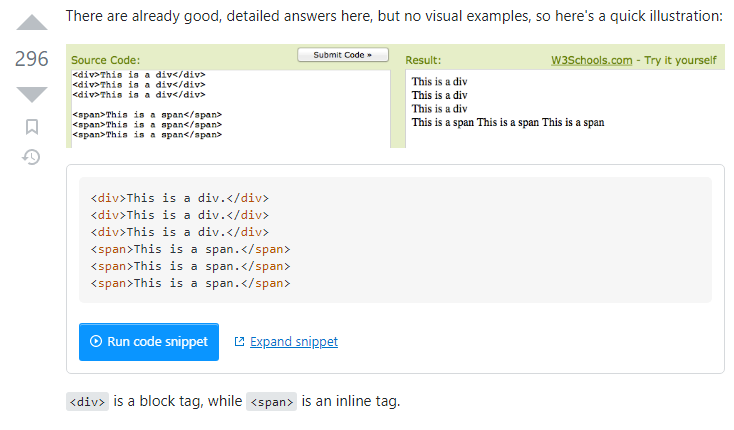
The former is recommended, as it is clean.

1. About useEffect(), useState(): They are about side effects.

Useful resources: <https://www.ruanyifeng.com/blog/2020/09/react-hooks-useeffect-tutorial.html>

使用useEffect()时，有一点需要注意。如果有多个副效应，应该调用多个useEffect()，而不应该合并写在一起。

1. In CSS, each section is titled with a className, essential a class, maybe. Then you define a class, cheat it as a view, same in Android Studio, then attributes of it go inside!
2. Span V.S. div in HTML



1. padding: <vertical size> <horizontal size>;

    padding: 18px 34px;

1. Navbar.Brand and navbar-brand are both bootstrap components.
2. Below, at the moment we treat “delta” as the input for the setInterval function, so every “delta” period, “tick()” will be invoked.

    useEffect(() => {

        let ticker = setInterval(() => {

            tick();

        }, delta);

1. href 的链接是一个view的id，即：#{id}。点击即到所绑定此id的view那里。
2. Below is saying, only show the <Col>…</Col> view when status message is {not None}/True.

{status.message && (

                                    <Col>

                                        <p

                                            className={

                                                status.success === true

                                                    ? "success"

                                                    : "danger"

                                            }

                                        >

                                            {status.message}

                                        </p>

                                    </Col>

                                )}

1. Below, when hovering .proj-imgbox, behaviour defined for .proj-txtx:

.proj-imgbox:hover .proj-txtx {

    top: 50%;

    opacity: 1;

}