# Typescript in React

Watch video to get basic understanding: <https://www.youtube.com/watch?v=ydkQlJhodio>

My takeaways:

Can use typescript template when running npx create-react-app

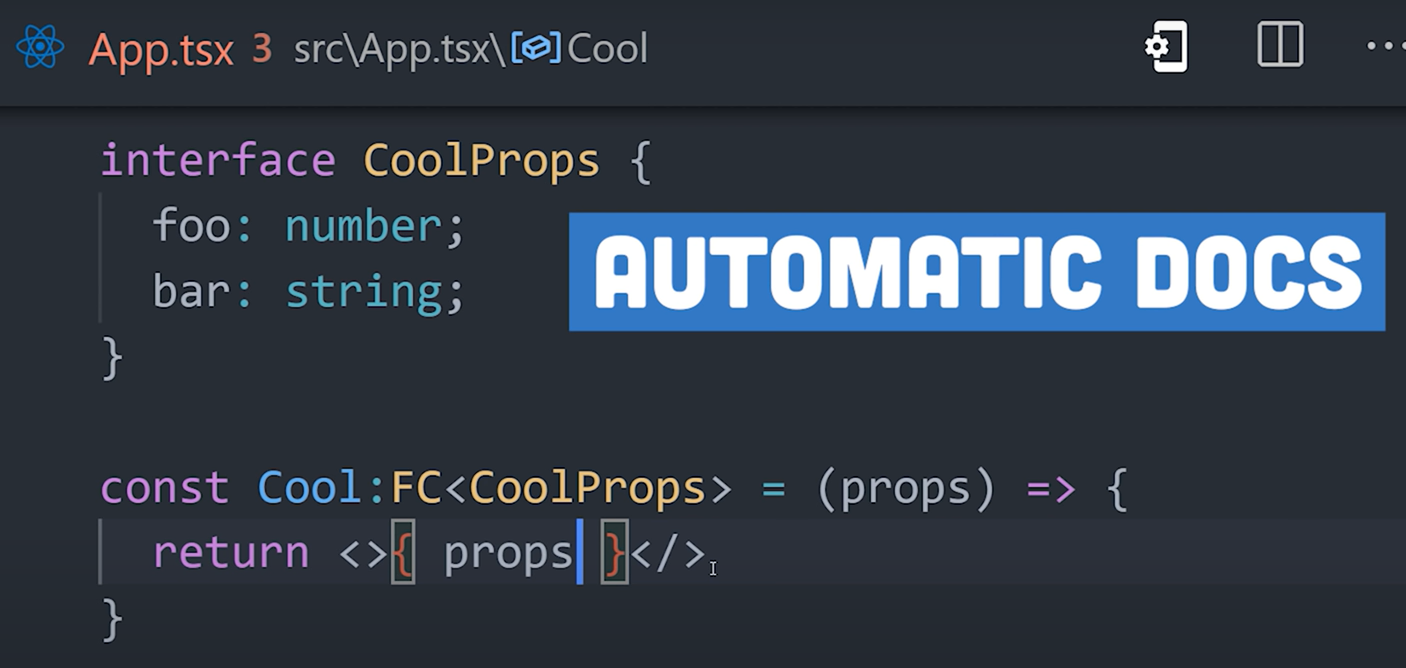
Can detect bad code, but can’t detect bad logic! (so still need tests)

In react:

Use FC for Function Component (it later turned out this isn’t needed anymore)

Give props an expected structure

Add ? before : to foo and bar to make them optional



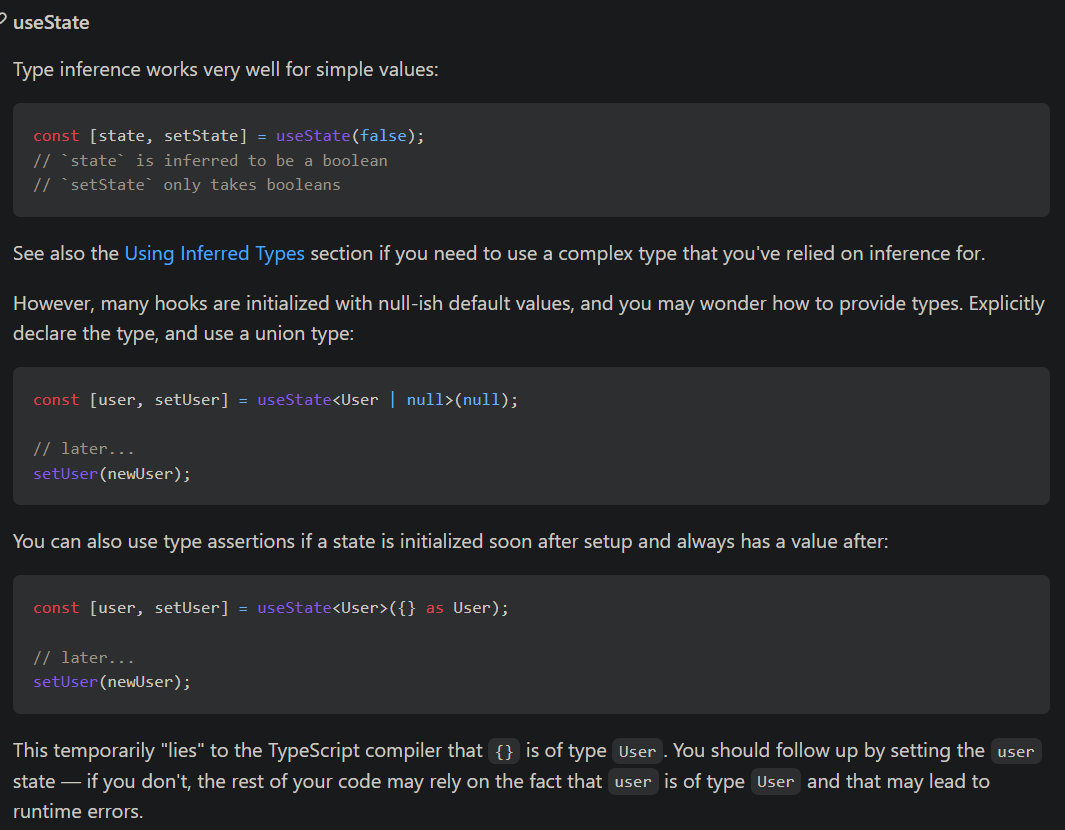
Researching useState and props syntax using <https://github.com/typescript-cheatsheets/react>

useState:

Type inference works well when setting simple state, providing you set it to the correct type in the first place!

If initialising with null, then you need to be explicit, use a union type and use angular brackets

e.g. const [user, setUser] = useState<User | null>(null);



Props

Define them at the top of the file for the component receiving the props – but this works cross file and will work wherever that component is called to ensure you are passing the correct props too!

## Arshi’s takeaway

Google for generics in TS

Typescript doesn’t just ensure the state is of the correct type, it checks that whatever you pass to the Set function to update state is the correct type too!

Can de-structure props and still use type annotation:

