FORMATION REACT

MASTERING HOOKS AND STATE ARCHITECTURE

REACT IN A FEW WORDS...

REACT IN A FEW WORDS...

- A JavaScript library for building user interfaces
- Declarative and component based
- Statefulness and life cycles
- V for View
- JSX and components as functions

REACT HAS MANY CAVEATS !

REACT CAVEATS

- Compatibility and eco system, hard to stay up to date
- Internal complexity
- Freedom to make crap
- Performance ?
- Setup and good practices

HOOKS

HOOKS

- useEffect
- useLayoutEffect
- useState
- useCallback
- useMemo
- Custom hooks

CUSTOM HOOKS: WHEN TO USE THEM

- To clean a component body
- To reuse a stateful or « effectful » logic across components
- To add a layer of abstraction

STATE MANAGEMENT, A MULTIPLE CHOICE QUESTION

STATE

- Redux
- MobX
- NgRx/Vuex
- React
- Any state management pattern or library

WHY DO WE NEED STATE MANAGEMENT

- Interactions
- UX, data persistence
- DX and maintainability
- Control and visibility

WHY DO WE NEED GLOBAL STATE MANAGEMENT

- State colocation is hard
- Props drilling
- Global control and understanding
- Event driven over state driven

REACT-COURSE

ARCHITECTURE

- Numbered exercises
- TDD in place
- Locked variable names and implementation
- Pre-coded UI samples
- Local instructions and readme button

USEFUL COMMANDS

- yarn ex $number
- yarn test $number
- yarn pgm
- yarn chill
- yarn giveup
- yarn next