1. Use a pascal case or Upper -camel-case Example - SubmitForm
2. Single responsibility – use an abstract type of complex component.
3. Use the container pattern separates concerns. Data loading logic separate from presentation logic .
4. Use stateless functional components and they are predictable.

* Stateless components are simple functional component without having a local state but remember there is a hook in react to add state behavior in functional component as well.
* Stateful component can contains the state object and event handling function, user actions as well.
* Stateless component are pure in nature which does a very specific task.

1. Good pattern is the one consists of two or more components.
2. Anti-pattern is bad practice to use.

* When modifying prototype of object then react component will inherit from the other objects.

7. Destructing props

8. React lends itself to functional programming paradigm.

9. Use stateless component – to resuse across different projects.