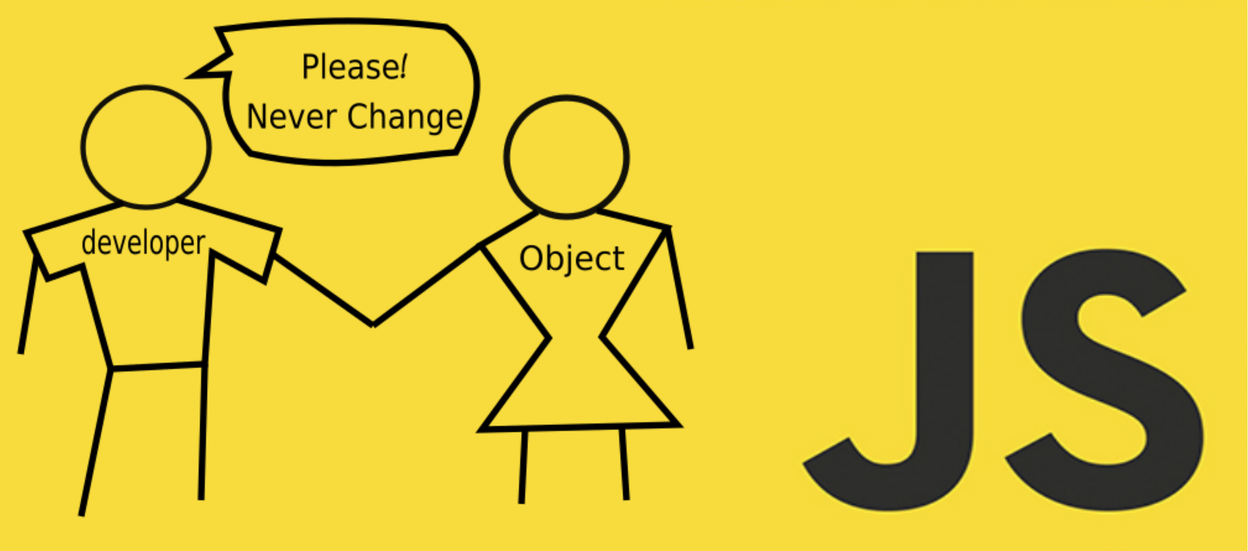
#javascript : Immutability in JavaScript

A basic understanding of Immutability would be:
Immutability means that it doesn't change, but rather create a new state from the old state.



Why is it so important?

- It's a principle used in functional programming as a paradigm.
- It's also important for performance reasons.
- Many core concepts of react framework related to state update are linked to immutability concepts in javascript.

What is mutability?

```
const Person = {
  name: 'Rizwan',
  email: 'rizwan@email.com'
}
const newPerson = Person
console.log(newPerson === Person) // true
```

```
The output is true, because both objects point to the same
memory location.
```

-> This is mutability because we assign a same memory reference to new variable newPerson

What is immutability?

```
const Person = {
  name: 'Rizwan',
  email: 'rizwan@email.com'
}

const newPerson = Object.assign({}, Person)
console.log(newPerson === Person) // false
```

The output is false because both objects are different and point to different memory locations.

We can use ES6 syntax called as rest parameter to create new object from existing ones:

```
const newPerson = {...Person}
```

Reference:
https://developer.mozilla.org/en-US/docs/Web/JavaScript

follow and add Rizwan Mushtaq Dhudhaal 👨‍💻 in your friend list