

// ES6 Classes

class Elf {

    constructor(name, weapon) {

        this.name = name;

        this.weapon = weapon

    }

    attack(){

        return 'attack with' + this.weapon

    }

}

// Beauty of Class, we are modelling real life

const peter = new Elf('Peter', 'stones') // peter is an instance of Elf

console.log(peter instanceof Elf) // true

// new keyword is instantiate the class

// this is syntatic sugar of JS, we dont really use classes like other programming language

// underneath the hood we are still using protypal inheritance

Inheritance

Core object from OOP.

Public vs Private

Anything that was private, in old time we use underscore \_ which actually didn’t do anything.

Other language actually has keyword of private so other cant access it

ES2020

Instead of constructor, we just do normal assigning (no var, let , const and this)

#age = 54 // # indicate this is private variable

Age = 54

OOP in ReactJS

4 Pillars of OOP

1. Encapsulation – The object
   1. Old time we have procedural programming, without structure
   2. Object, as a box can interact with each other, easier to maintain and reusable
2. Abstraction
3. Inheritance
   1. Avoid having to rewrite same code
   2. Save memory space
4. Polymorphism – Many forms
   1. Ability to call the same method on different object, and each object response in different way
   2. We simply doing method overwriting
   3. The ability to refine methods and customize their method to their object