```
const __x=Name.create()
const __y=Name.create();
const __validate=Name.create();
Point = {
    //private members
    private __x: 0,
    private __y: 0,
    private __validate(x,y) {
      return typeof x == 'number' &&
               typeof y = 'number'},
    //public members
    new(x,y) {
        if (!this[__validate](x,y)) throw "invalid";
        return this <| {
                private __x: x,
                private __y: y
               }
     };
    add(anotherPoint) {
         return this.new(this[__x]+another[__x],
                         this[__y]+another[__y])
    }
}
```

```
const __x=Name.create();
const __y=Name.create();
const __validate=Name.create();
Point = {
    //private members
    [__x]: 0,
    [ __y]: 0,
    [__validate](x,y) {
      return typeof x == 'number' &&
               typeof y = 'number'},
    //public members
    new(x,y) {
        if (!this[__validate](x,y)) throw "invalid";
        return this <| {
                [__x]: x,
                [__y]: y
               }
     };
    add(anotherPoint) {
         return this.new(this[__x]+another[__x],
                         this[__y]+another[__y])
    }
}
```

```
const __x=Name.create();
const __y=Name.create();
const __validate=Name.create();
Point = {
    //private members
    @__x: 0,
    @__y: 0,
    @__validate(x,y) {
      return typeof x == 'number' &&
               typeof y = 'number'},
    //public members
    new(x,y) {
        if (!this@validate(x,y)) throw "invalid";
        return this <| {
                @__x: x,
                @__y: y
               }
     };
    add(anotherPoint) {
        return this.new(this@__x+another@__x,
                        this@__y+another@__y)
    }
}
```