

## Домашнее задание для самопроверки

### урок “Интерфейсы, Дженерики”

**Цель:** Отработать навык написания типов, используя полученные знания из урока

#### **План выполнения:**

Необходимо дописать пропуски в коде, отмеченные символами

```
/* (1) */  
/* (2) */  
/* (3) */  
/* (4) */  
/* (5) */
```

так, чтоб приведенные в нижней части кода тестовые сценарии успешно выполнялись в песочнице, а так же не было ошибок в редакторе кода.

```
type ColorHexCode = string;  
  
interface VehicleParameters {  
    type: string;  
    weight: number;  
    colorHex: /* (1) */    // - Template Literal с первым символом # + ColorHexCode  
    vin: string;  
}  
  
interface IVehicle extends VehicleParameters {  
    getParametersForBoats: /* (2) */  
}  
  
class Vehicle<T extends VehicleParameters> implements IVehicle {  
    /* (3) */  
    constructor({type, /* (4) */ }: T) {  
        /* (5) */  
        this.vin = vin;  
    }  
}
```

```
getParametersForBoats() {  
    return this.type === 'boat' ? this : null  
}  
}  
  
/**  
 * Тестовые данные для проверки  
 */  
  
const boat = new Vehicle({  
    type: 'boat',  
    weight: 200,  
    colorHex: '#00ff00',  
    vin: '1234ABC5389DEF83958',  
})  
const parameters = boat.getParametersForBoats()  
console.log('parameters', parameters)  
/**  
 * console.log выше выведет следующее  
 */  
"parameters", {  
  "type": "boat",  
  "weight": 200,  
  "colorHex": "#00ff00",  
  "vin": "1234ABC5389DEF83958"  
}  
*  
*/  
  
const bike = new Vehicle({  
    type: 'bike',  
    weight: 12,  
    colorHex: '#ffff00',  
    vin: '5948ABC5389DEF832428',  
})  
const parameters2 = bike.getParametersForBoats()  
console.log('parameters2', parameters2) // console.log выше выведет null
```

**Рекомендации для выполнения:**

1. Для успешного написания второго пропуска смотрите, что возвращает одноименный метод класса Vehicle
2. this метода класса в данном случае равен this самого класса
3. Для успешного написания четвертого пропуска рекомендуется посмотреть уроки по деструктурирующему присваиванию объекта в JS/TS

<https://learn.javascript.ru/destructuring-assignment#destrukturizatsiya-obekta>