// MEMBUAT OBJECK DENGAN CLASS

class Color {

    constructor(r,g,b, name){

        this.r = r;

        this.g = g;

        this.b = b;

        this.name = name;

    }

    colorName (){

        console.log(`Name color : ${this.name}`)

    }

    rgb(){

        const {r,g,b} = this;

        return `rgb(${r}, ${g}, ${b})`;

    }

    rgba(a = 1){

        const {r,g,b} = this;

        return `rgb(${r}, ${g}, ${b}, ${a})`;

    }

    hex() {

        const {r,g,b} = this;

        return '#' + ((1 << 24) + (r << 16) + (g << 8) + b).toString(16).slice(1);

    }

}

const skyColor = new Color(123, 111, 23, 'sky');

console.log(skyColor.hex());

console.log(skyColor.rgb());

console.log(skyColor.rgba());

console.log(skyColor.colorName());

document.body.style.backgroundColor = skyColor.rgba(0.6)

// PENGGUNAAN EXTENDS OBJECT

class Peliharaan {

    // parent object

    constructor(name, age){

        this.name = name;

        this.age = age;

    }

    makan() {

        return `${this.name}, umur: ${this.age} lagi makan`

    }

}

class Kucing extends Peliharaan {

    //child object

    meong(){

        return 'Meeeong'

    }

}

class Anjing extends Peliharaan {

    gonggong(){

        return "guk guk guk";

    }

}

//

const kucing = new Kucing('mbul', 3);

const anjing = new Anjing('buldog', 4);

console.log(kucing.name)

console.log(kucing.makan())

console.log(kucing.meong())

console.log(anjing.name)

console.log(anjing.makan())

console.log(anjing.gonggong())