abstract class Veiculo {

    constructor(

        public *modelo*: *string*,

        public *marca*: *string*,

        public *cor*: *string*,

        public *numeroRodas*: *number*

    ) {}

    abstract clone(): Veiculo;

    represent(): *string* {

        return `Modelo: ${*this*.modelo}, Marca: ${*this*.marca}, Cor: ${*this*.cor}, Número de Rodas: ${*this*.numeroRodas}`;

    }

}

class Carro extends Veiculo {

    constructor(

*modelo*: *string*,

*marca*: *string*,

*cor*: *string*,

*numeroRodas*: *number*,

        public *numeroPortas*: *number*

    ) {

*super*(*modelo*, *marca*, *cor*, *numeroRodas*);

    }

    clone(): Carro {

        return **new** Carro(*this*.modelo, *this*.marca, *this*.cor, *this*.numeroRodas, *this*.numeroPortas);

    }

    represent(): *string* {

        return `${*super*.represent()}, Número de Portas: ${*this*.numeroPortas}`;

    }

}

class Moto extends Veiculo {

    constructor(

*modelo*: *string*,

*marca*: *string*,

*cor*: *string*,

*numeroRodas*: *number*,

        public *cilindradas*: *number*

    ) {

*super*(*modelo*, *marca*, *cor*, *numeroRodas*);

    }

    clone(): Moto {

        return **new** Moto(*this*.modelo, *this*.marca, *this*.cor, *this*.numeroRodas, *this*.cilindradas);

    }

    represent(): *string* {

        return `${*super*.represent()}, Cilindradas: ${*this*.cilindradas}`;

    }

}

class Aplicacao {

    criarVeiculos(): Veiculo[] {

        const veiculos: Veiculo[] = [];

        // Criação de carros

        veiculos.push(**new** Carro("Sedan", "Toyota", "Prata", 4, 4));

        veiculos.push(**new** Carro("SUV", "Honda", "Preto", 4, 5));

        // Criação de motos

        veiculos.push(**new** Moto("Esportiva", "Yamaha", "Vermelho", 2, 600));

        veiculos.push(**new** Moto("Cruiser", "Harley-Davidson", "Preto", 2, 1200));

        return veiculos;

    }

    cloneVeiculos(*veiculos*: Veiculo[]): Veiculo[] {

        const clonedVeiculos: Veiculo[] = [];

        for (const veiculo of *veiculos*) {

            clonedVeiculos.push(veiculo.clone());

        }

        return clonedVeiculos;

    }

    imprimirVeiculos(*veiculos*: Veiculo[]): *void* {

        for (const veiculo of *veiculos*) {

            console.log(veiculo.represent());

        }

    }

}

const app = **new** Aplicacao();

const veiculos = app.criarVeiculos();

const clonedVeiculos = app.cloneVeiculos(veiculos);

app.imprimirVeiculos(clonedVeiculos);