Softex

Aluna: Vandilma Candido

Pesquise sobre padrões de projeto e escolha um para apresentar e descrever o seu funcionamento. Além disso, explique quais as vantagens e desvantagens comparados a outros e mostre suas referências.

**Prototype**

Aqui vamos abordar o padrão criacional **Prototype**, esse padrão serve para criar objetos a partir de outros existentes, criando uma cópia do objeto e podendo adicionar mais atributos nele.

É possível criar um método construtor da seguinte forma:

function Carro( modelo, ano, km ) {

this.modelo = modelo;

this.ano = ano;

this.km = km;

this.toString = function () {

return this.modelo + " andou " + this.km + " km desde " + this.ano;

}

// instanciando Carro

var opala = new Carro( "Chevrolet Opala", 1979, 130000 );

var belina = new Carro( "Ford Belina", 1982, 200000 );

// chamando o método toString() de cada Carro

console.log( opala.toString() );

console.log( belina.toString() );

Desvantagem:

Essa é uma forma muito ineficiente de se criar um método construtor, visto que toda vez que Carro for instanciado o método toString() será replicado, consumindo mais memória. Além do mais, usando esse padrão será muito mais difícil criar herança para essa classe.

Vantagem:

No JavaScript a ideia de Protótipos é muito forte, pois todos Objetos em JS possui propriedades internas. Que são chamadas de Prototype que é uma referência para outros objetos resumindo é a forma que o JS faz herança.