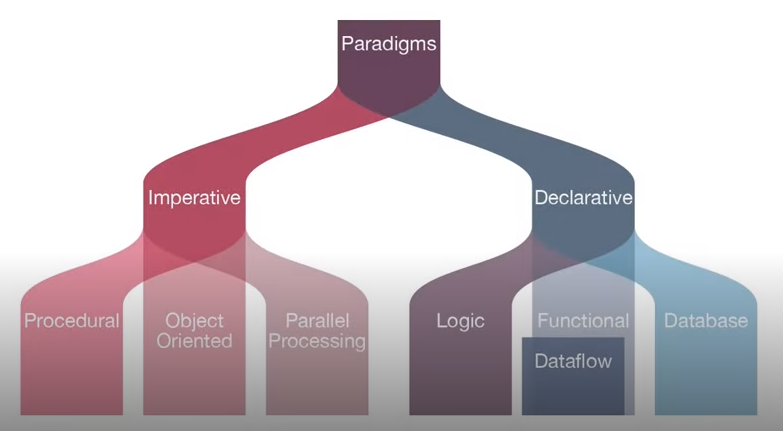
**Orientação a objetos**

**Paradigmas:**

* **imperativo:** sempre foca em como vai resolver os problemas
* **declarativo:** foca no que será feito
* **orientação a objeto** é um paradigma **imperativo**

****

* os programas são “objetos” que possuem uma série de propriedades

**Pilares:**

* herança
* polimorfismo
* encapsulamento
* abstração

**Abstração**

* temos um problema complexo e a abstração é cada vez tornar ele mais simples
* ex: carro é um objeto complexo. ao abstraí-lo, veremos que é um veículo de rodas. com outra abstração veremos que é apenas um veículo

**Herança**

* objetos filhos que herdam as propriedades e métodos do objeto pai

**Encapsulamento**

* cada classe tem propriedades e métodos independentes do restante do código

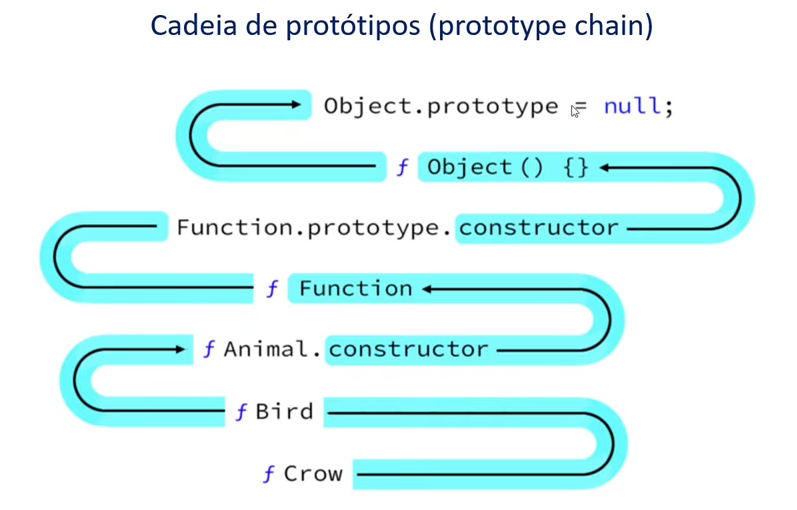
**Polimorfismo**

* objetos podem herdar a mesma classe pai, mas se comportam de forma diferente quando invocamos seus métodos
* Classes mais complexas herdam métodos de classes mais abstratas

**OOJS**

**Protótipos**

* todos os objetos JavaScript herdam propriedades e métodos de um prototype. O objeto Object.prototype está no topo desta cadeia.

****

**Classes**

* **não existem** nativamente no **JavaScript**
* são a **Syntatic Sugar:** uma sintaxe feita para facilitar a escrita
* todas as classes são objetos e a herança se dá por protótipos

