

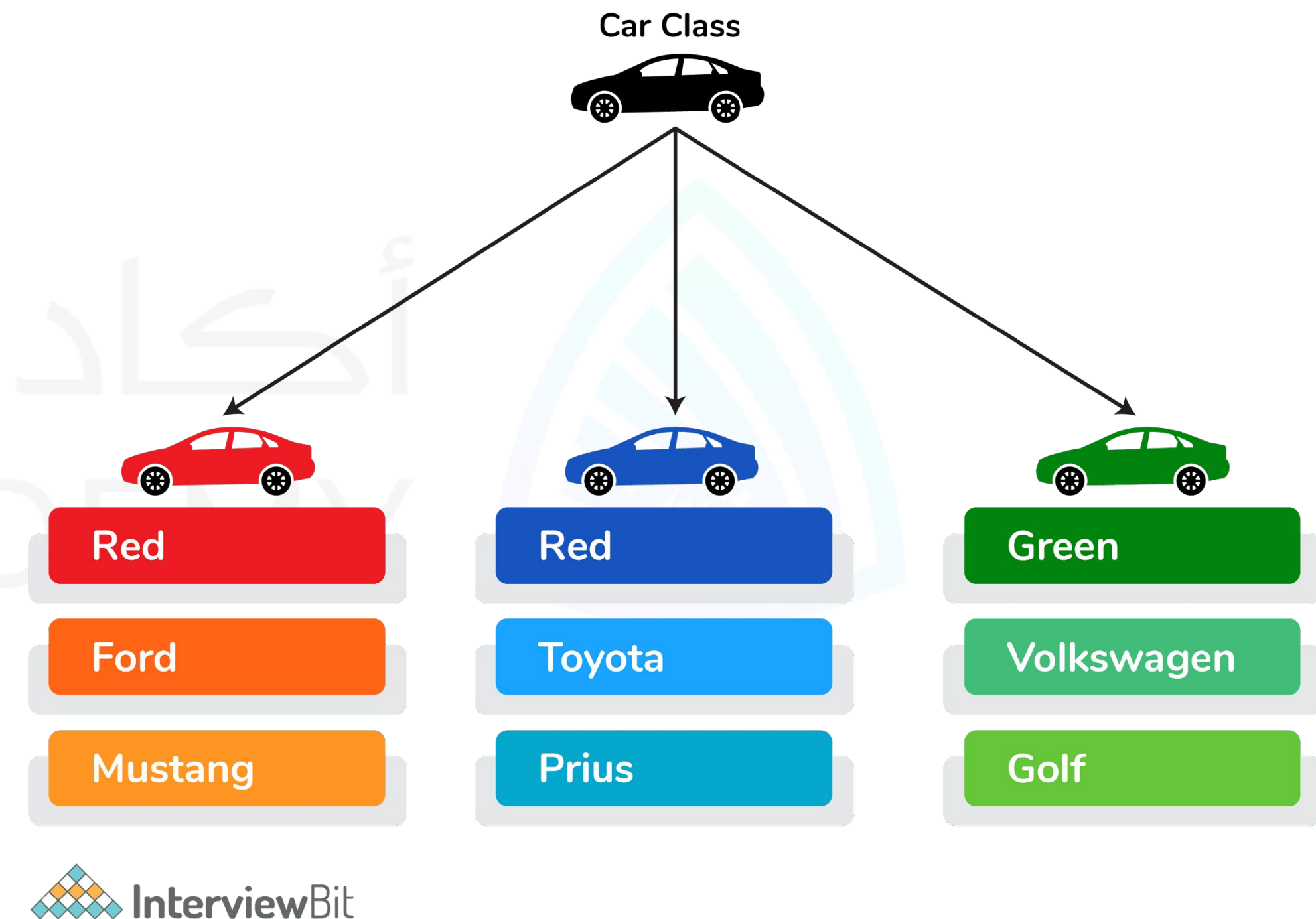
# JavaScript V: OOP

# Objective

- What is Object Oriented Programming
- JavaScript Objects
- Procedural Programming vs. OOP
- How do we Design OO Systems
- Creating Defined Objects
- Function meets Object

# What is Object Oriented Programming

- ❖ We as humans think in terms of objects.
- ❖ Object Oriented programming (OOP) is a programming paradigm that relies on the concept of **classes** and **objects**.



# JavaScript Objects

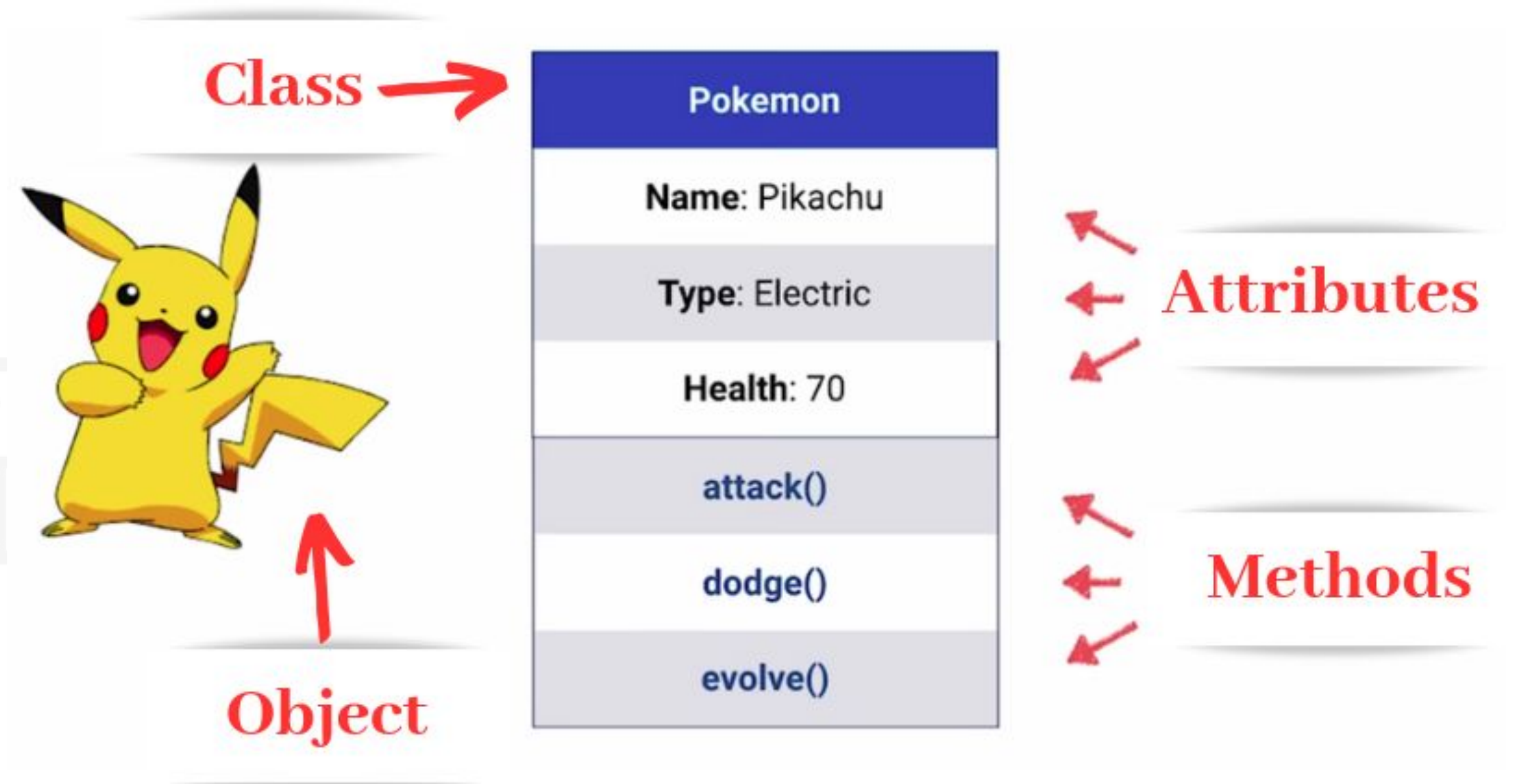
- ❖ JS is an **object-based** language
- ❖ OOP serves to allow the components of your code to be as modular as possible.
- ❖ Benefits:
  - ☐ Shorter development time
  - ☐ Easier debugging

# Procedural Programming vs. OOP

Procedural Programming	OOP
Waterfall sequence	System is designed in terms of objects which communicate with each other to accomplish a given task
Data are separate from the code	Data and code are encapsulated into a single module. Data is passed to modules using methods

# How do we Design OOP Systems?

- **Class:** blueprint from which objects are made and consists of data and the code that manipulates it
- **Attributes:** characteristics of an object
- **Methods:** a function related to an object





# Creating Defined Objects

There is 2 methods to create defined objects:

## ❖ Method 1: Using Object Literal

```
let car = {  
  brand: "Porsche",  
  model: "GT3",  
  year: 2004,  
  colour: "White",  
  howOld: function() { getFullYear() - this.year; }  
};
```

## ❖ Attributes:

Property	Value
Brand	Porsche
Model	GT3
Year	2004
Colour	White

# Creating Defined Objects

## ❖ Method 2: Using Object Constructor:

```
function carDescription(brand, model, year, colour) {  
  this.brand = brand;  
  this.model = model;  
  this.year = year;  
  this.colour = colour;  
};  
let car = new carDescription("Porsche", "GT3", 2012, "White");  
let car2 = new carDescription("Ford", "Fiesta", 2015, "Red");  
let car3 = new carDescription("Opel", "Corsa", 2014, "White");
```

- To change the colour: `car.colour = "Black";`
- To delete properties of an object: `delete car.colour;`



# Some Objects that you have Already Encountered

❖ Most of the built-in code that you have worked with so far has used objects.

❖ When you use code such as

```
let htmlSelect = document.getElementById('personList')
```

you have been using the document object

❖ Every time you create an array, you are actually creating an array object that is defined by an array class

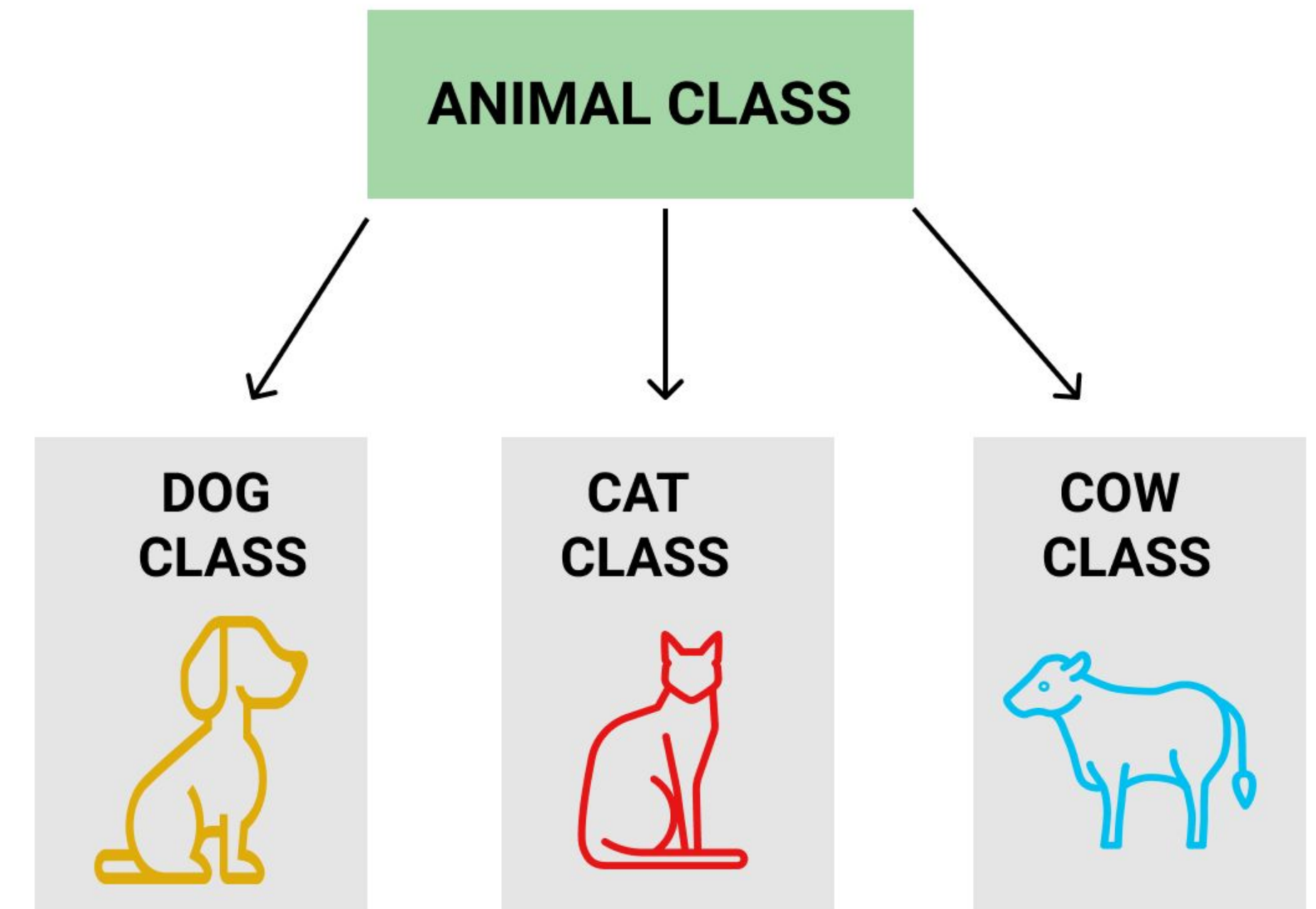
# Function meets Object

- ❖ Let's consider the declaration of an object with implementation through a **function**.

```
let loaded = {};  
loaded.testing = function(signal) {  
  alert("Hello World! " + signal);  
  loaded.signal = signal;  
}  
loaded.testing("This page has loaded!");
```

# Inheritance

- ❖ Inheritance is a mechanism in which one class acquires the property of another class.



# Summary

- What is Object Oriented Programming
- Procedural Programming vs. OOP
- How do we Design OO Systems
- JavaScript Objects
- Creating Defined Objects
- Function meets Object



## Resources

- JHD

أكاديمية طويق  
TUWAIQ ACADEMY

