# Constructors

A constructor creates a new instance of the class. It initializes all the variables and does any work necessary to prepare the class to be used. In the line

Car c = new Car();

Car() is the constructor. A constructor has the same name as the class.

If no constructor exists Java provides a generic one that takes no arguments (a noargs constructor), but it's better to write your own. You make a constructor by writing a method that has the same name as the class. Thus the Car constructor is called Car().

Constructors do not have return types. They do return an instance of their own class, but this is implicit, not explicit.

The following method is a constructor that initializes license plate to an empty string, speed to zero, and maximum speed to 120.0.

Car() {  
 this.licensePlate = "";   
 this.speed = 0.0;  
 this.maxSpeed = 120.0;  
 }

Better yet, you can create a constructor that accepts three arguments and use those to initialize the fields as below.

Car(String licensePlate, double speed, double maxSpeed) {  
  
 this.licensePlate = licensePlate;   
 this.speed = speed;  
 if (maxSpeed > 0) this.maxSpeed = maxSpeed;  
 else this.maxSpeed = 0.0;  
 if (speed > this.maxSpeed) this.speed = this.maxSpeed;  
 if (speed < 0) this.speed = 0.0;  
 else this.speed = speed;  
   
 }

Or perhaps you always want the initial speed to be zero, but require the maximum speed and license plate to be specified:

Car(String licensePlate, double maxSpeed) {  
  
 this.licensePlate = licensePlate;   
 this.speed = 0.0;  
 if (maxSpeed > 0) this.maxSpeed = maxSpeed;  
 else this.maxSpeed = 0.0;  
  
 }

[Previous](http://docs.google.com/18.html) | [Next](http://docs.google.com/20.html) | [Top](http://docs.google.com/index.html) | [Cafe au Lait](http://www.cafeaulait.org/)

Copyright 1997-2000 Elliotte Rusty Harold

[elharo@metalab.unc.edu](mailto:elharo@metalab.unc.edu)

Last Modified June 13, 2000