

Something new:
Make a Class...

Mr. Neat
Java

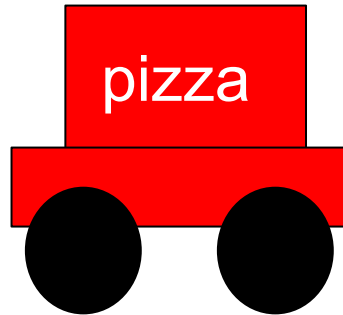
Ta - Da, now we are going to make a Class!

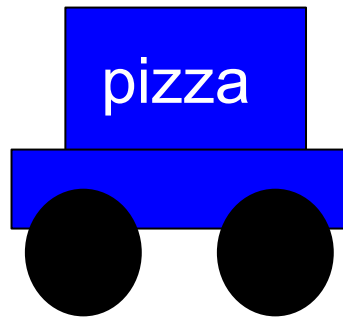
- the class definition has three parts:
 - 1) fields (also called global variables or instance variables)
 - 2) constructors
 - 3) methods

Constructors.....

just added another constructor to allow:

```
Car fred = new Car(17.0, 23.0, "pizza");
```

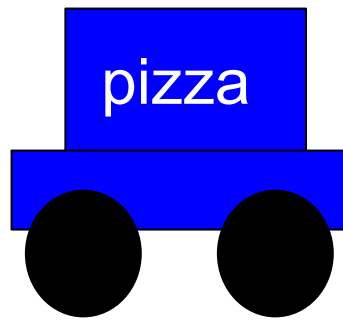




now want to do this:

```
Car fred = new Car(17.0, 23.0, "pizza", Color.BLUE);
```

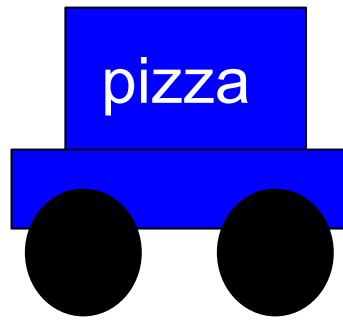
To do this we are going to **reuse** existing constructors
(any time you copy and paste code, there is a better way)



```
Car fred = new Car(17.0, 23.0, "pizza", Color.BLUE);
```

Need the keyword *this*

this provides a way to call another constructor in the same file. So if we wanted to call the existing constructor...



we can call:

```
this(x, y);
```

inside of the

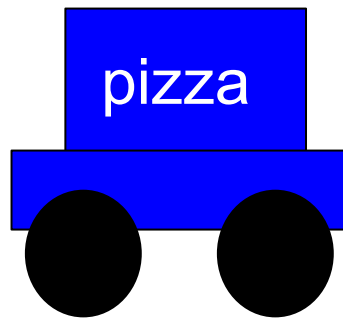
```
public Car(double xx, double yy, String mess)
```

and call

```
this(x,y,st)
```

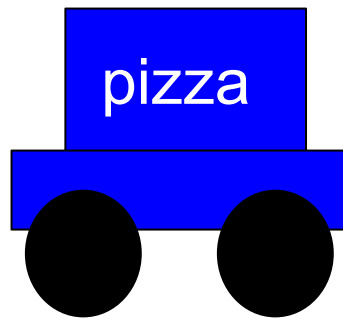
inside of the new constructor

```
public Car(double a, double b, String c, Color d)
```



note:

- *this(...)* must be the first line of the constructor
- can only call *this(...)* once in a given constructor



in starter.java

```
Car fred = new Car(17.0, 23.0, "pizza", Color.BLUE);
```

in Car.java

```
public Car(double xx, double yy, String mess, Color c)
{
    this(xx,yy,mess);
    // all the code to set the colors
}
```


Lab

- Add a third constructor to your Car class that allows the user to set the Color of the Car object
- test it by making a few Cars with different colors

Hint

- Don't forget to initialize the Color variable in the first two constructors.
- The first two constructors did not know about the Color variable cuz it didn't exist