

Class and Method Definitions

- A class as a blueprint

Class Name: Automobile

Data:

amount of fuel _____

speed _____

license plate _____

Methods (actions):

accelerate:

How: Press on gas pedal.

decelerate:

How: Press on brake pedal.

Class and Method Definitions

First Instantiation:

Object name: patsCar

```
amount of fuel: 10 gallons  
speed: 55 miles per hour  
license plate: "135 XJK"
```

Second Instantiation:

Object name: suesCar

```
amount of fuel: 14 gallons  
speed: 0 miles per hour  
license plate: "SUES CAR"
```

Third Instantiation:

Object name: ronsCar

```
amount of fuel: 2 gallons  
speed: 75 miles per hour  
license plate: "351 WLF"
```

Objects that are
instantiations of the
class **Automobile**

Class Files and Separate Compilation

- Each **Java** class definition usually in a file by itself
 - File begins with name of the class
 - Ends with **.java**
- Class can be compiled separately
- Helpful to keep all class files used by a program in the same directory

Information Hiding

- Programmer using a class method need not know details of implementation
 - Only needs to know what the method does
- Information hiding:
 - Designing a method so it can be used without knowing details
- Also referred to as abstraction
- Method design should separate what from how

The **public** and **private** Modifiers

- Type specified as **public**
 - Any other class can directly access that object by name
- Classes generally specified as **public**
- Instance variables usually not **public**
 - Instead specify as **private**

The Keyword **this**

- Referring to instance variables outside the class – must use
 - Name of an object of the class
 - Followed by a dot
 - Name of instance variable
- Inside the class,
 - Use name of variable alone
 - The object (unnamed) is understood to be there

Accessor and Mutator Methods

- When instance variables are private must provide methods to access values stored there
 - Typically named **getSomeValue**
 - Referred to as an accessor method
- Must also provide methods to change the values of the private instance variable
 - Typically named **setSomeValue**
 - Referred to as a mutator method

Encapsulation

- Consider example of driving a car
 - We see and use break pedal, accelerator pedal, steering wheel – know what they do
 - We do not see mechanical details of how they do their jobs