

More methods

We will add three more methods to the Person class. You will need the code for this class, which was written earlier. When you add a method to the class, the method declaration must be inside the braces for the class, but not inside the braces for another method.

Remember to recompile the Person class each time you modify it.

Getting older

The first method to add will be named `haveBirthday`, and should increase the person's age by 1. This method takes no parameters and returns nothing.

```
void haveBirthday() {  
    age++;  
}
```

The `age++` operation increases the field's value by 1. It doesn't matter whether we write `age` or `this.age` here, because there's no other variable named `age` in this method.

Try it out in the Eclipse:

```
Person alice = new Person("Alice", 19);  
alice.haveBirthday();  
alice.getAge()  
  
returns 20 (int)  
  
alice.haveBirthday();  
alice.getAge()  
returns 21 (int)
```

Greet a friend

Next we will write a method named `greet`, which will create a message saying "hello" to another person, greeting them by name. This method takes one parameter for the other person, and returns the message as a `String`.

```
String greet(Person other) {  
    return "Hello, " + other.getName() + "!";  
}
```

This method calls the other person's `getName` method, then joins their name together with the message. Try it out in with you IDE:

```
Person alice = new Person("Alice", 19);  
Person rob = new Person("Robert", 71);  
alice.greet(bob)
```

returns "Hello, Robert!" (String)

Notice that the greet method returns a String rather than printing a message to the console — the method specification doesn't say anything should be printed, so there is no `System.out.println` statement. A program should only print what it's supposed to print!

Years until retirement

The `yearsUntilRetirement` method will print a message saying how many years until the person reaches age 68, at which point they can claim their state pension. A different message should be printed if they are already 68 years old.

```
void yearsUntilRetirement() {  
    if(age < 68) {  
        int years = 68 - age;  
        System.out.println(name + " can retire in " + years + "  
years.");  
    } else {  
        System.out.println(name + " can retire already!");  
    }  
}
```

This method takes no parameters and returns nothing. Try it out in the Code Pad:

```
Person alice = new Person("Alice", 19);  
Person bob = new Person("Robert", 71);  
alice.yearsUntilRetirement();  
bob.yearsUntilRetirement();
```

Because the method prints to the console, you should see the output in the Terminal Window:

Alice can retire in 49 years.
Robert can retire already!