MODULE 4: JAVA CLASS METHODS

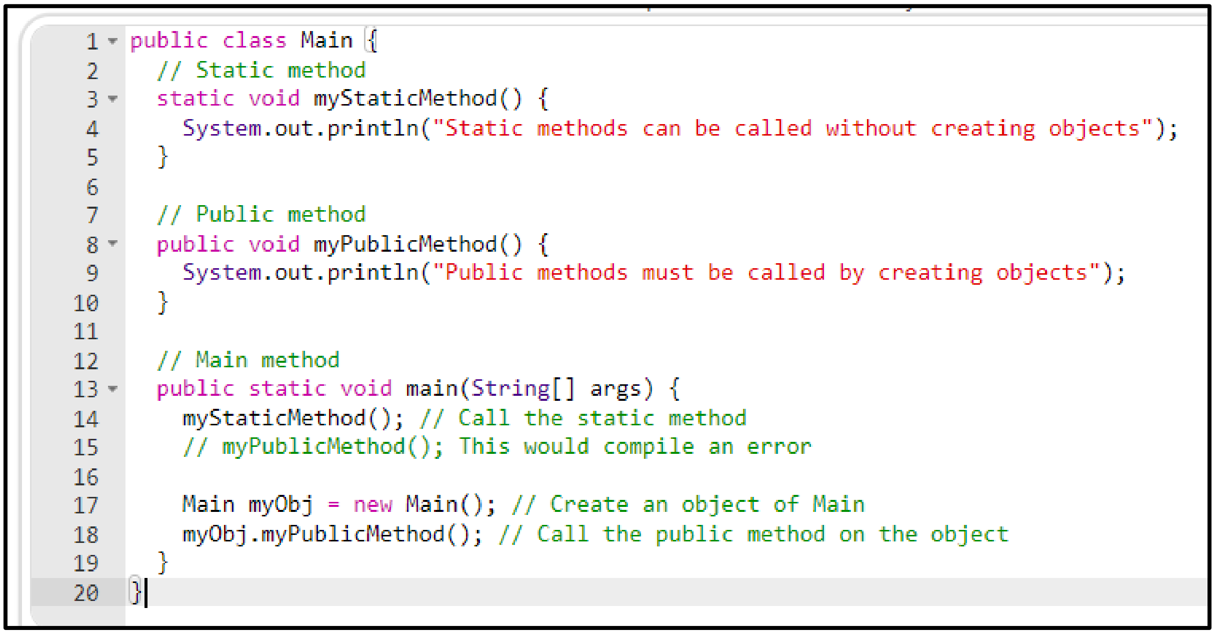
You learned from the Java Methods chapter that methods are declared within a class, and that they are used to perform certain actions:

myMethod() prints a text (the action), when it is **called**. To call a method, write the method's name followed by two parentheses **()** and a semicolon**;**

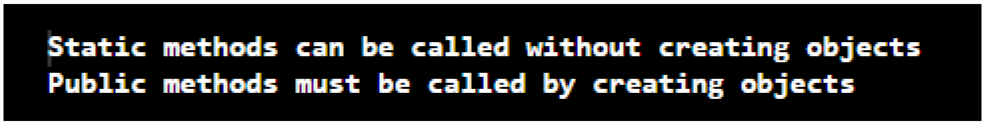
STATIC VS. PUBLIC

You will often see Java programs that have either static or public attributes and methods.

We created a static method, which means that it can be accessed without creating an object of the class, unlike public, which can only be accessed by objects:

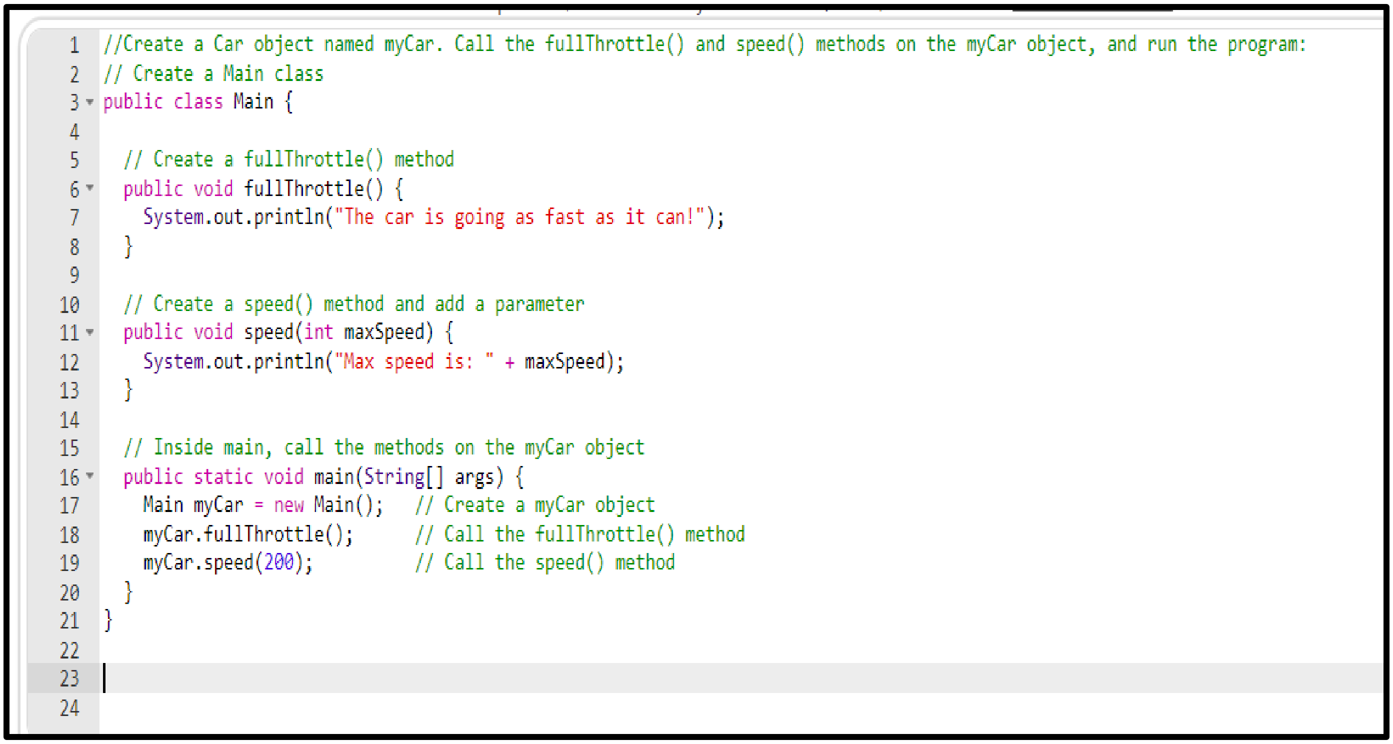


```java
public class Main {
  // Static method
  static void myStaticMethod() {
    System.out.println("Static methods can be called without creating objects");
  }

  // Public method
  public void myPublicMethod() {
    System.out.println("Public methods must be called by creating objects");
  }

  // Main method
  public static void main(String[] args) {
    myStaticMethod(); // Call the static method
    // myPublicMethod(); This would compile an error

    Main myObj = new Main(); // Create an object of Main
    myObj.myPublicMethod(); // Call the public method on the object
  }
}
```



```
Static methods can be called without creating objects
Public methods must be called by creating objects
```

ACCESS METHODS WITH AN OBJECT



```java
//Create a Car object named myCar. Call the fullThrottle() and speed() methods on the myCar object, and run the program:
// Create a Main class
public class Main {

  // Create a fullThrottle() method
  public void fullThrottle() {
    System.out.println("The car is going as fast as it can!");
  }

  // Create a speed() method and add a parameter
  public void speed(int maxSpeed) {
    System.out.println("Max speed is: " + maxSpeed);
  }

  // Inside main, call the methods on the myCar object
  public static void main(String[] args) {
    Main myCar = new Main();   // Create a myCar object
    myCar.fullThrottle();      // Call the fullThrottle() method
    myCar.speed(200);          // Call the speed() method
  }
}
```

Remember that.....

The dot (.) is used to access the object's attributes and methods.

To call a method in Java, write the method name followed by a set of parentheses **()**, followed by a semicolon (;).