

Java Static Methods

V102



Static Methods

- the main method is where a stand alone Java program normally begins

```
public class StaticMethodExample
{
    public static void main(String[] args)
    {
        //starting point of execution

        System.out.println("In main method");
        doThis();
    }

    public static void doThis()
    {
        System.out.println( "method 1");
    }

}
```

Parameter Passing

```
public class AddingNumbers
{
```

```
    public static void main(String[] args)
    { //main
        int num1 = 3;
        int num2 = 4;

        addThem(num1, num2);
    } //main
```

the call statement to the method

2 arguments

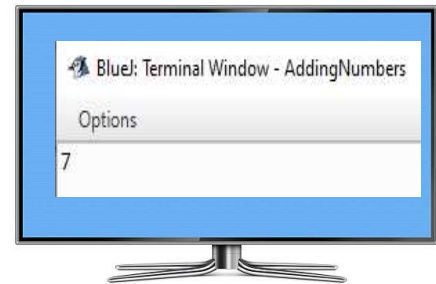
2 int Value Parameters

```
    public static void addThem(int x, int y)
    { //addThem

        System.out.println( x + y );

    } //addThem
```

```
}
```



Method Return Statement

```
/**
 * The method addThem adds the two parameters together and
 * returns the sum.
 *
 * @author James Bond
 * @version 100
 * @return The sum of the two parameters is returned
 */
```

```
public class AddingNumbers
{ //add
```

```
    public static void main(String[] args)
    { //main
        int num1 = 3;
        int num2 = 4;
        int sum = 0;

        sum = addThem(num1, num2);
        System.out.println( sum );

    } //main
```

```
    public static int addThem(int x, int y)
    { //addThem
```

```
        return x + y;
```

```
    } //addThem
```

```
} //add
```



```

/**
 * Here is an example of method overloading in Java
 *
 * @author Dave Slemon
 * @version v100
 */
public class Overloading_Example
{
    static int addThem(int x, int y)
    {
        return x + y;
    }

    static double addThem(double x, double y) {
        return x + y;
    }
}

public static void main(String[] args) {

    int myNum1 = addThem(8, 5);
    double myNum2 = addThem(4.1, 6.9);

    System.out.println("int: " + myNum1);
    System.out.println("double: " + myNum2);
}

```

```

/**
 * Here is an example of method overloading in Java
 *
 * @author Dave Slemon
 * @version v100
 */
public class Overloading_Example
{
    static int addThem(int x, int y)
    {
        return x + y;
    }

    static double addThem(double x, double y) {
        return x + y;
    }
}

static int addThem(double x, int y)
{
    return x + y;
}

static double addThem(int x, double y) {
    return x + y;
}

public static void main(String[] args) {

    int myNum1 = addThem(8, 5);
    double myNum2 = addThem(4.1, 6.9);

    System.out.println("int: " + myNum1);
    System.out.println("double: " + myNum2);
}
}

```

Method Overloading and Return

1. a java program may have multiple methods with the same name as long as the parameter signature is unique
2. if a method has a return value other than void it must have a return statement with a variable or expression of the proper type
3. multiple return statements allowed, the first one encountered is executed and method ends

```
public class Overloading_Example
{
    static int addThem(int x, int y)
    {
        return x + y;
    }

    static double addThem(double x, double y) {
        return x + y;
    }
}
```