

Arrays as method parameters

```
public static void main(String[] args) {  
  
}
```

Notice here that the parameter to the main method is an array of String.

This means we can pass an array of Strings to this method when it's called.

Or, if we use this method as the entry point to our application, we can pass data on the command line to this method.

Up until now, I've only shown you this particular method signature.

Variable arguments (varargs)

But this signature can be written in a slightly different way.

We can replace the brackets after the String type, which we know tells us this method will take an array of String.

And we can instead replace that with three periods.

This is a special designation for Java that means Java will take zero, one, or many Strings as arguments to this method and create an array with which to process them in the method.

```
public static void main(String... args) {  
  
}
```

Variable arguments (varargs)

The array will be called args and be of type String.

So, what's the difference then?

The difference is minor within the method body but significant to the code that calls the method.

```
public static void main(String... args) {  
  
}
```

When can you use variable arguments (varargs) ?

There can be only one variable argument in a method.

The variable argument must be the last argument.