String.split() method
The **split** Method

- The **split** method creates an array of smaller strings from a larger string based on one or more *separators*.
- **split** is called on the longer string (the one to “to be cut”)
- Takes a single argument: a string with the separator(s).
- Returns the created array

```java
String text = "Today is a hot summer day."
String[] words = text.split(" ");
for (int i = 0; i < words.length; i++) {
    System.out.printf("word[%d] = %s\n", i, words[i]);
}
```

Output:
- `words[0] = Today`
- `words[1] = is`
- `words[2] = a`
- `words[3] = hot`
- `words[4] = summer`

Here it separates at an empty space

| "Today" | "is" | "a" | "hot" | "summer" | "day." |
The split Method

- You can specify multiple characters that separate words.
- Below, 4 separators are given: comma, space, and dash.
- To specify multiple characters, you must enclose them in square brackets: [] (see the example argument: "[, -]").

```java
String text = "Let's count: One,two,three.";
String[] words = text.split("[,, -]");
for (int i = 0; i < words.length; i++) {
    System.out.printf("word[%d] = %s
", i, words[i]);
}
```

Output:
```
word[0] = Let's count: One,
word[1] = ,three.
```

- If no [], then the entire argument string is the separator. See "two" :

```java
String text = "Let's count: One,two,three.";
String[] words = text.split("two");
for (int i = 0; i < words.length; i++) {
    System.out.printf("word[%d] = %s
", i, words[i]);
}
```

Output:
```
word[0] = Let's count: One,
word[1] = ,three.
```
Split method

This method is useful for

- reading easily and array of strings from the user
- Extracting individual pieces of information from a string in a specific format. E.g. :
  - date (09/15/2019):
    "09/15/2019".split("/")  ->  ["09", "15", "2019"]
  - phone number: 817-888-9999
    "817-888-9999".split("-")  ->  ["817", "888", "9999"]

See also how to convert a string to a number (int or double):

```java
int x = Integer.parseInt("78");  // throws error if bad
int bad = Integer.parseInt("78.3");  // throws NumberFormatException
double y = Double.parseDouble("78.3");
Integer n = Integer.valueOf("78");  // valueOf returns Integer object
```