

## Unit 10

- String concatenation
- Example with loops



# Methods — Concatenation

`String result = s1 + s2;`  
concatenates s1 and s2

`String result = s1.concat(s2);`  
the same as s1 + s2

`result += s3;`  
concatenates s3 to result

`result += num;`  
converts **num** to **String** and  
concatenates it to result



# Numbers to Strings

- Three ways to convert a number into a string:

1.

```
String s = "" + num;
```

2.

```
String s = Integer.toString (i);
```

```
String s = Double.toString (d);
```

3.

```
String s = String.valueOf (num);
```

**Integer** and **Double** are “wrapper” classes from **java.lang** that represent numbers as objects. They also provide useful static methods.



# Numbers from Strings

```
String s1 = "-123", s2 = "123.45";  
int n = Integer.parseInt(s1);  
double x = Double.parseDouble(s2);
```

- These methods throw a `NumberFormatException` if `s` does not represent a valid number (integer, real number, respectively).



# String Example

Ex. Write a method that determines whether String s is a palindrome.

```
public boolean isPalindrome(String s)
{
```



# String Example

Ex. Write a method that counts the number of occurrences of the String target in the given String s.

```
public int countOccurrences(String s, String target)
{
```



# String JavaBat Examples

