

# Reference variables and Strings

Course: CPSC 1150  
Instructor: Dr. Bitá Shadgar

Lecture 19

# Learning Outcomes

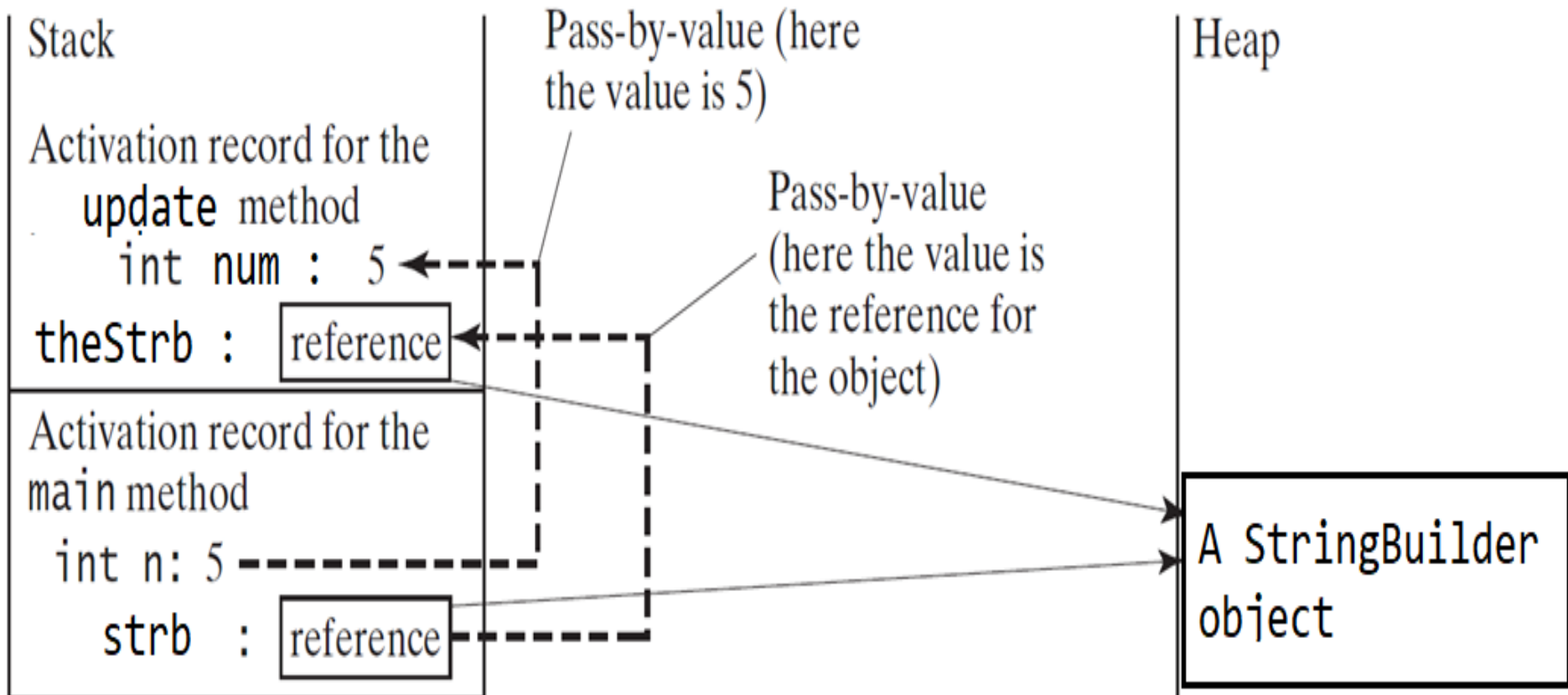
- Pass parameters by reference to methods
- Pass parameters by value to methods
- Pass mutable and immutable objects to methods

# Reference variables

- Objects are accessed via object's **reference variable**, which contains references to the object.
- If a data field of a reference type **does not reference any object**, the data field holds a special literal value, **null**.
- **Arrays** are treated as objects in Java. An array variable is a variable that contains a reference to an array.

# Passing objects to methods

- Objects can be passed to methods, like passing arrays.
- Passing an object is actually passing the reference of the object.
- The object is shared between two reference variables; the argument and the parameter
- Check out `PassParameters.java` to find out more.
- `StringBuilder` is an alternative java class for `Strings`. It is more flexible to manipulate strings.



# Mutable and immutable objects

- Immutable objects cannot be changed.
- String objects are immutable objects.
- StringBuilder objects are mutable objects
- Arrays are mutable objects

```
str1 = "Hello";
```



```
str1 += " World";
```



# Passing immutable objects to Methods

- Immutable objects behave like primitive types when they are passed to a method.
- Check out `PassImmutable.java` to find out more.

# Some methods in the String class

- These are all instance methods
  - **split(String regex):** Splits this string around matches of the given regular expression into a String array.
  - **toCharArray():** Converts this string to a new character array.
- Check out `ParsingString.java` for better understanding.



# More Practice

- Read a list of words from a file (such as `unsortedFile.txt`) and store them in a file (such as `sortedList.txt`) in an ascending order.
- To keep the problem simple, only consider words that include only alphabetical character or hyphen.

# MorePractice

Develop body of the method `count(String[ ] arr)` that takes an array of strings as argument, and prints how many times each digits '1' – '9' occurred in the strings.

For example, if the method is called with following argument

```
String [ ] arr = {"This is 2 of 2", "July 14", "cpsc1150"}
```

Then the program should print:

**digit 1 : 3**

**digit 2 : 2**

**digit 4 : 1**

**digit 5 : 1**

Since digit '1' occurs three times, '2' two times, '4' once, and '5' once.