Intermediate Material

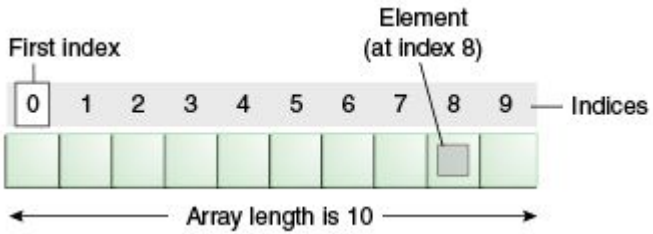
SD Girls Coding Camp 2016

Day 2

Arrays

- Arrays are a way of **storing data**
- Like the dressers that store your clothes
  - If you know which draw to look in, you have instant access
  - If you don’t know which draw to look in, you have to go through all
- Only difference: each “draw” of an array stores **1** element
- All elements must have the same type: bool, char, int, etc.
- Syntax for initializing with specific *size* and *type*:
  - *type*[] arrayName = new *type*[*size*]

- Can also initialize simultaneously with creation:
  - *type*[] arrayName = {value1, value2, value3, ...};
  - Array will have # elements = # items in curly braces
- Accessing/Modifying a specific index of array is instant
  - Access: arrayName[*index*]
  - Modify: arrayName[*index*] = newValue;
- Number of elements in the array: arrayName.length
- Extra: Also possible to make 2-dimensional arrays
  - *type*[][] arrayName = new *type*[*rows*][*columns*];



Methods

- Help prevent unnecessary repetition of code
- Basic idea: write code once, assign it a name, run code anytime by referring to name

Examples:

```
public static void main(String[] args)
{
        printHello();
        printHello();
        System.out.println(“Done”);
}

public void printHello()
{
        System.out.println(“Hello”);
}
```

Computer Console:

```
Hello
Hello
Done
```

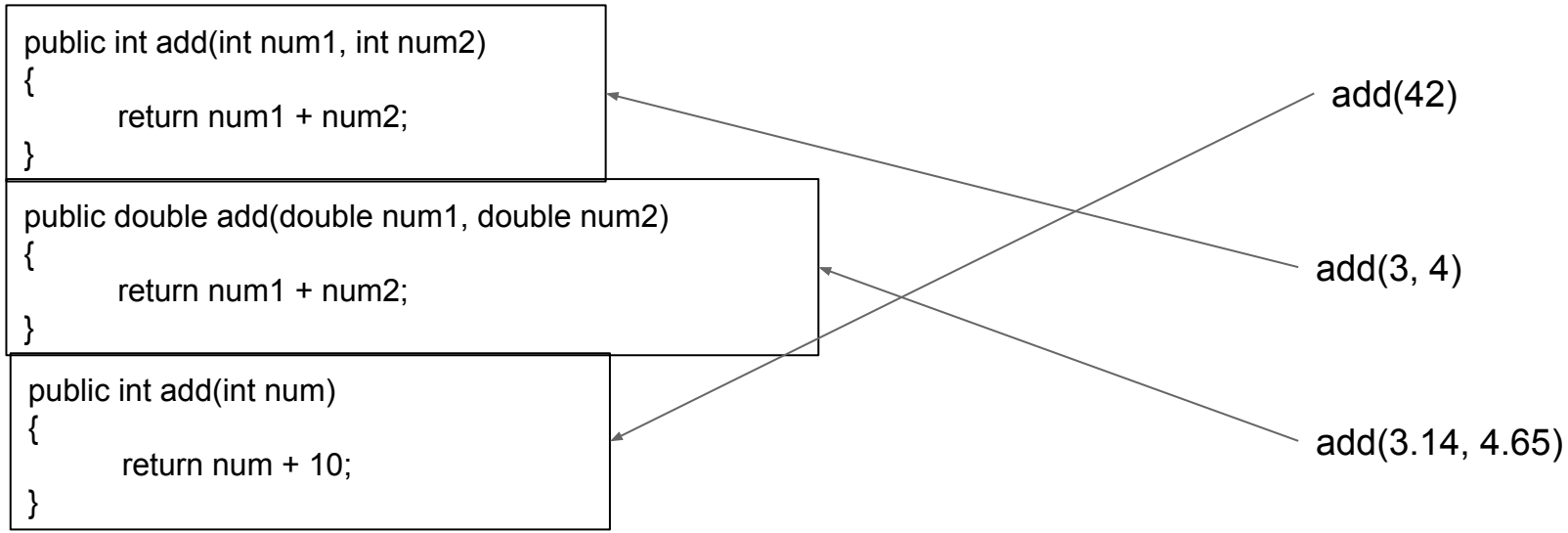
- Can take in input arguments to use during execution
- Good coding practice to have each method perform ONE specific task
  - Example: Compute sum of numbers in a file
  - Method/Task 1: Read in numbers from file
  - Method/Task 2: Compute sum from list of numbers

Example (Methods with Arguments):

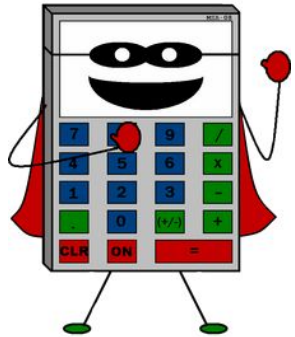
```
public int sum(int num1, int num2)
{
        return num1 + num2;
}

public int diff(int num1, int num2)
{
        return num1 - num2;
}
```

sum(3, 4) = 7
sum(5, 8) = 13
diff(42, 22) = 20
sum(sum(3, 4), diff(42, 22)) = 27
etc...

- Can have multiple methods with same name - **Overloading**
- HOWEVER, the methods must differ in their arguments (types of arguments OR number of arguments)
- Think of it ask: if overloaded methods exist, computer must be able to figure out which one to use (aka some difference exists)



Stats Calculator



- Given a text file that has 1000 student scores
  - Calculate mean, max and min score
  - Challenge yourself: calculate mode too
- Your program must utilize proper design: each method should accomplish ONE task
- The file should only be read in once
  - Hint: consider storing values in an array

* The text file can be found on the camp website :)