CS111
Introduction to Computer Science

Fall 2015

- Arrays

Array

- Array is a fixed-size data structure that stores elements of the same type sequentially.
  - Arrays allows us to organize large amounts of data

- Declaring an array in Java

```
dataType[] arrayReferenceVariable;
```

- Declaring an integer array in Java

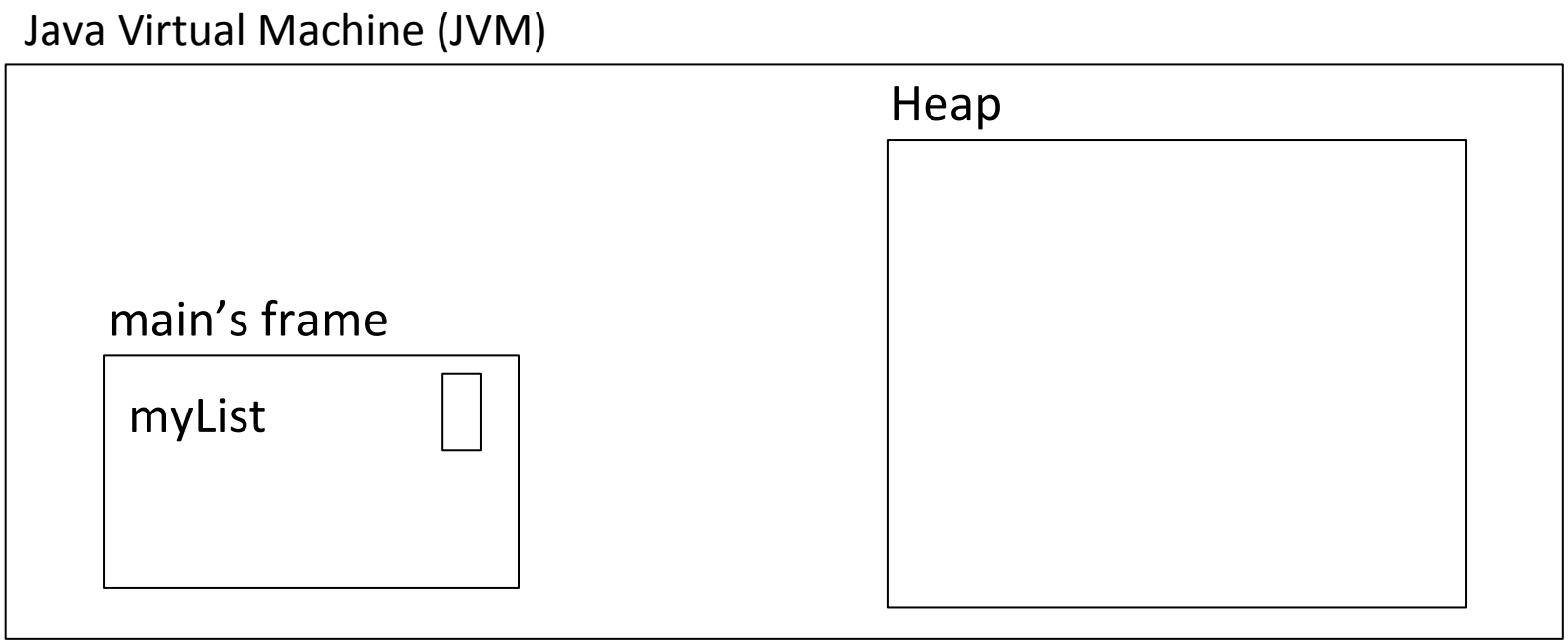
```
int[] myList;
```

Declaring an Array in Java

- Declare

```
int[] myList;
```

Creates a variable that will refer to an array of integers

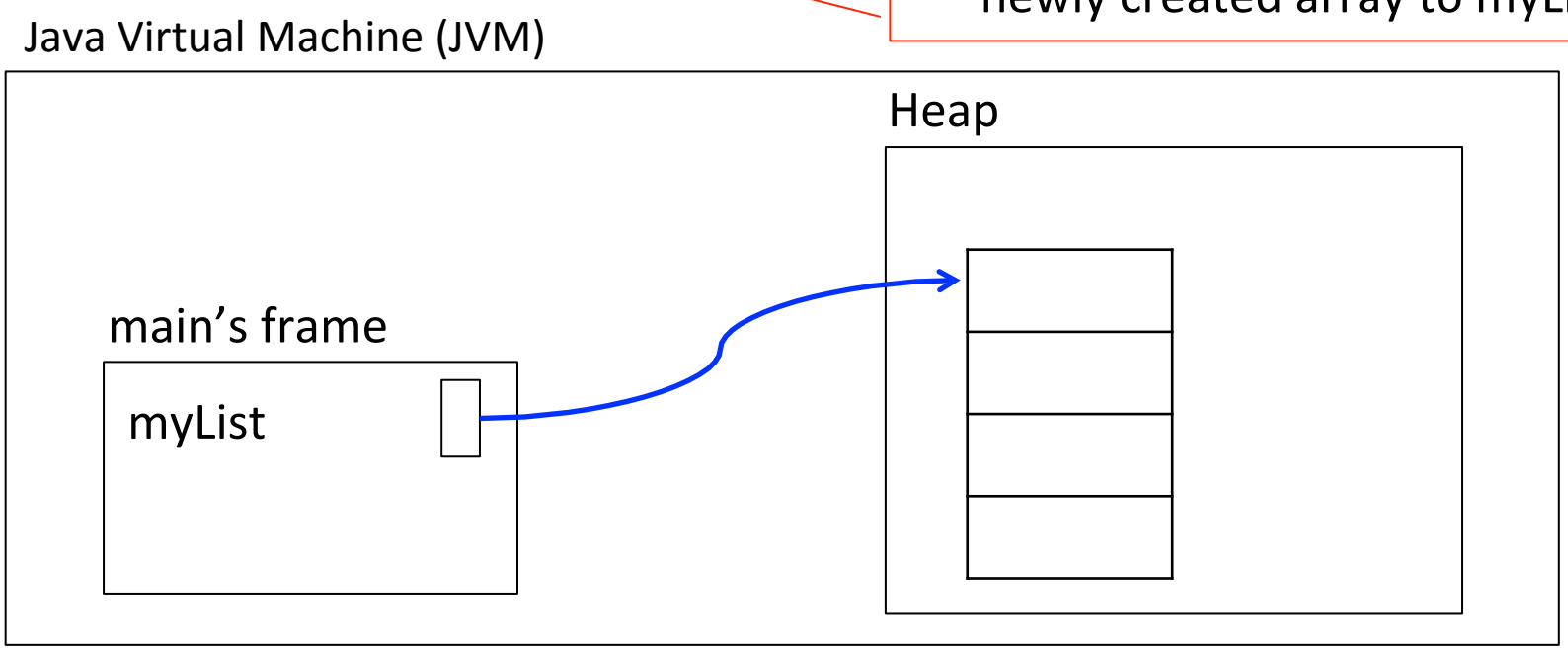


Creating an Array in Java

- Declare and create an array

```
int[] myList = new int[4];
```

1. Physically creates an array
2. Assigns the reference of the newly created array to myList

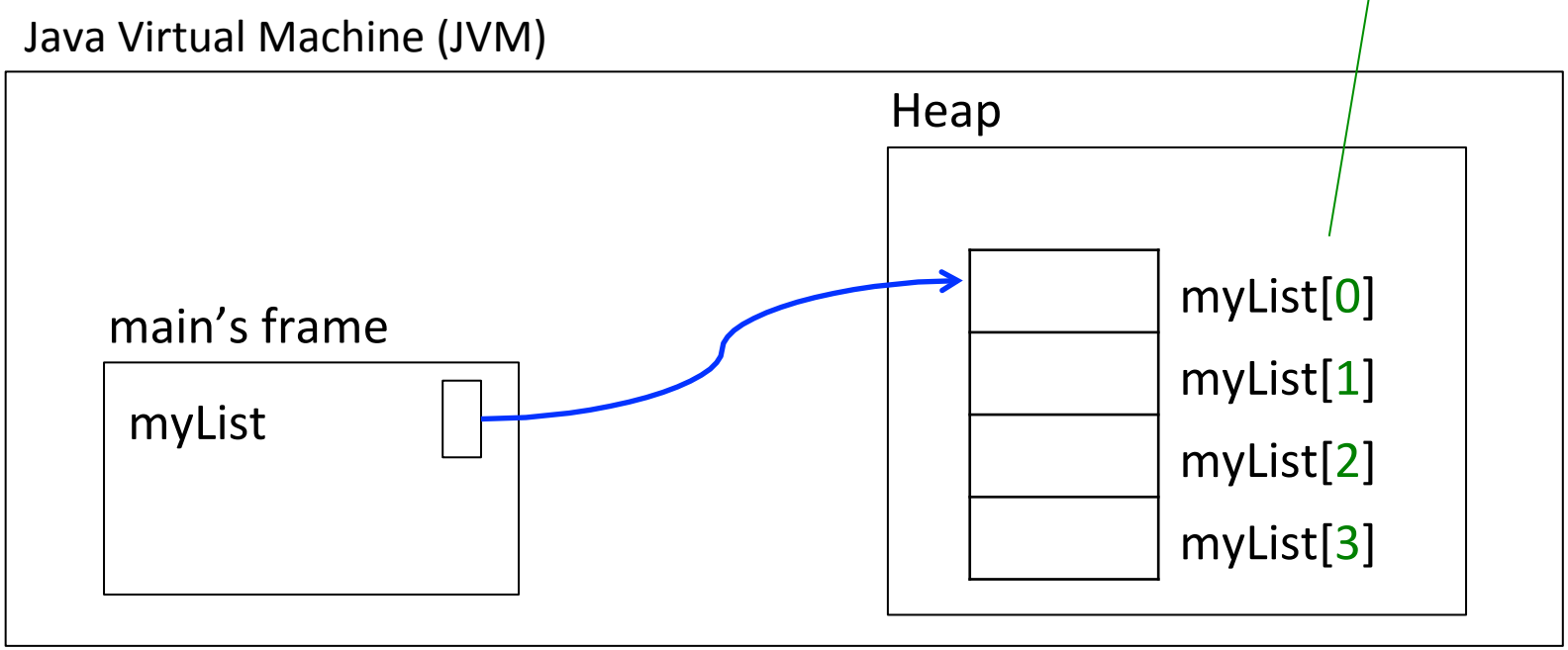


Creating an Array in Java

- Declare and create an array

```
int[] myList = new int[4];
```

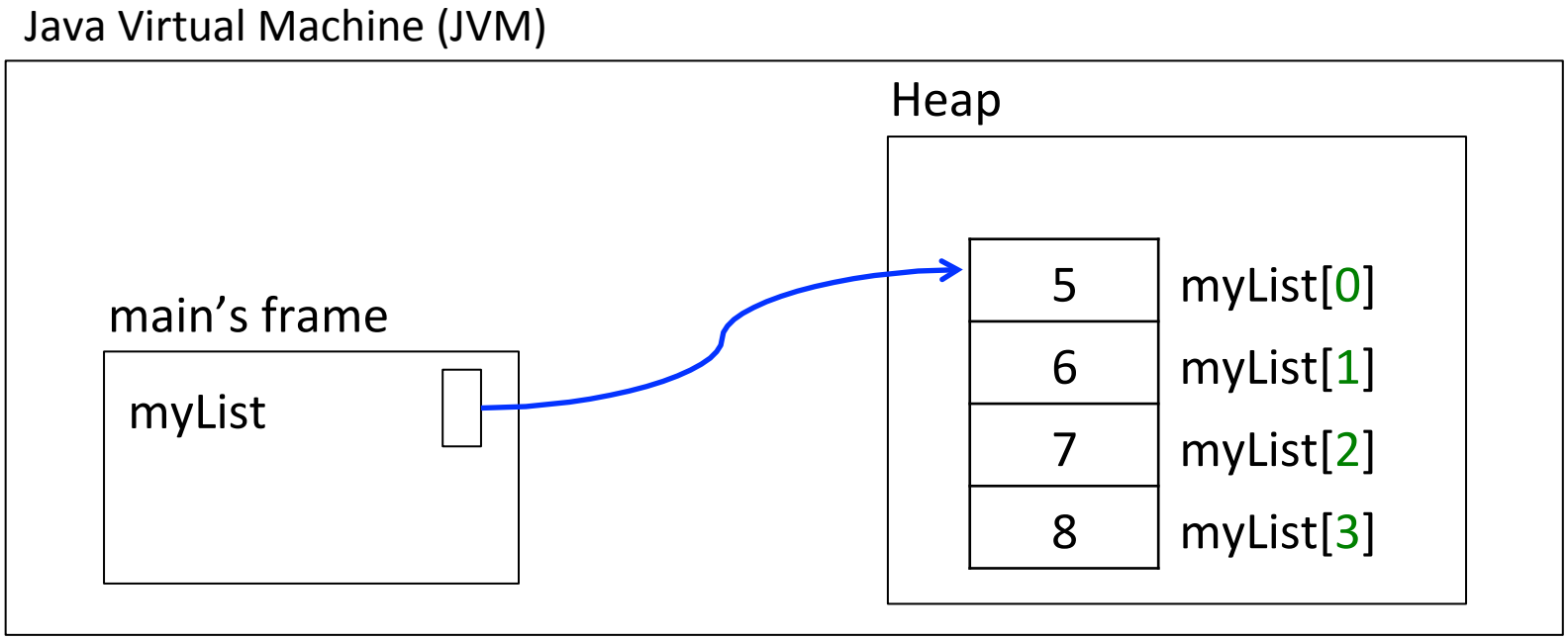
Array indices range from 0 to length-1



Initializing an Array in Java

- Use a loop to go over the elements of an array

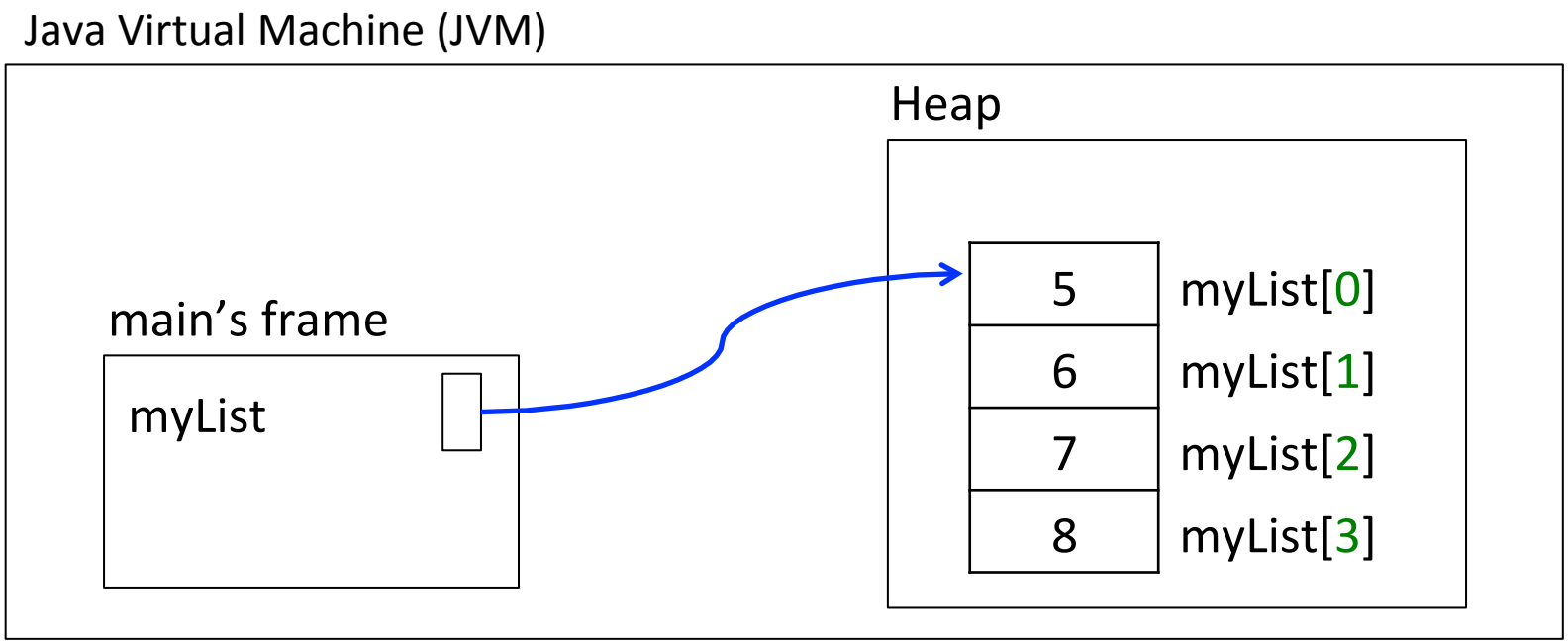
```
for (int i = 0; i < 4; i++) {
    myList[i] = i+5;
}
```



Creating an Array in Java

- Declare, create and initialize an array

```
int[] myList = {5, 6, 7, 8};
```



Passing Arrays to Methods

- Just as you can pass any primitive type values to methods
- Method to print all elements of an array:

```
public static void printArray (int[] array) {

    for (int i = 0; i < array.length; i++) {
        System.out.printf("%d ", array[i]);
    }
}
```

Return Arrays from Methods

- A method may also return an array
- Method to reverse all elements of an array

```
public static int[] reverseArray (int[] array) {

   int[] result = new int[array.length];

   for (int i = 0; i < array.length; i++) {
      int j = result.length – 1 – i;
      result[j] = array[i];
   }
   return result;
}
```

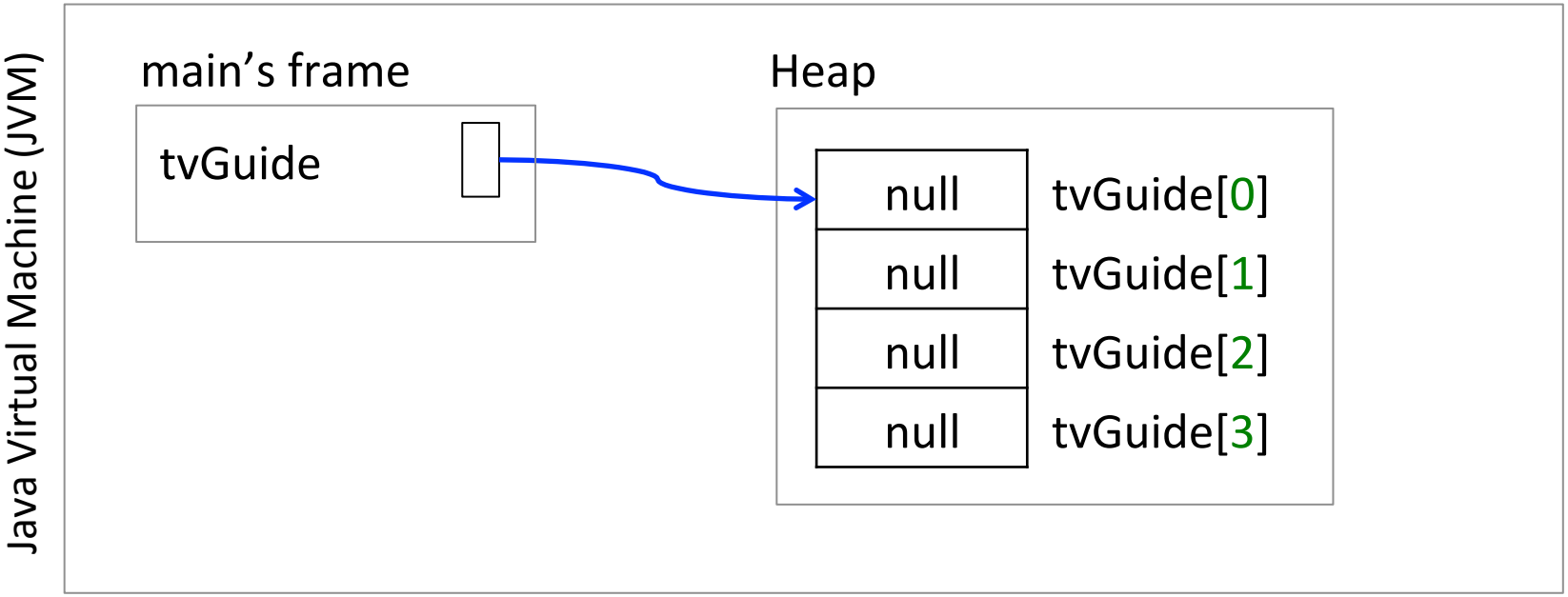
TV Guide

- Program keeps track of channels available
  - Read maximum number of channels available from user
  - Offer menu options:
    - Add channel
    - Look up channel by number
    - Print all channels
    - Look up channel by name

See the TVGuide.java

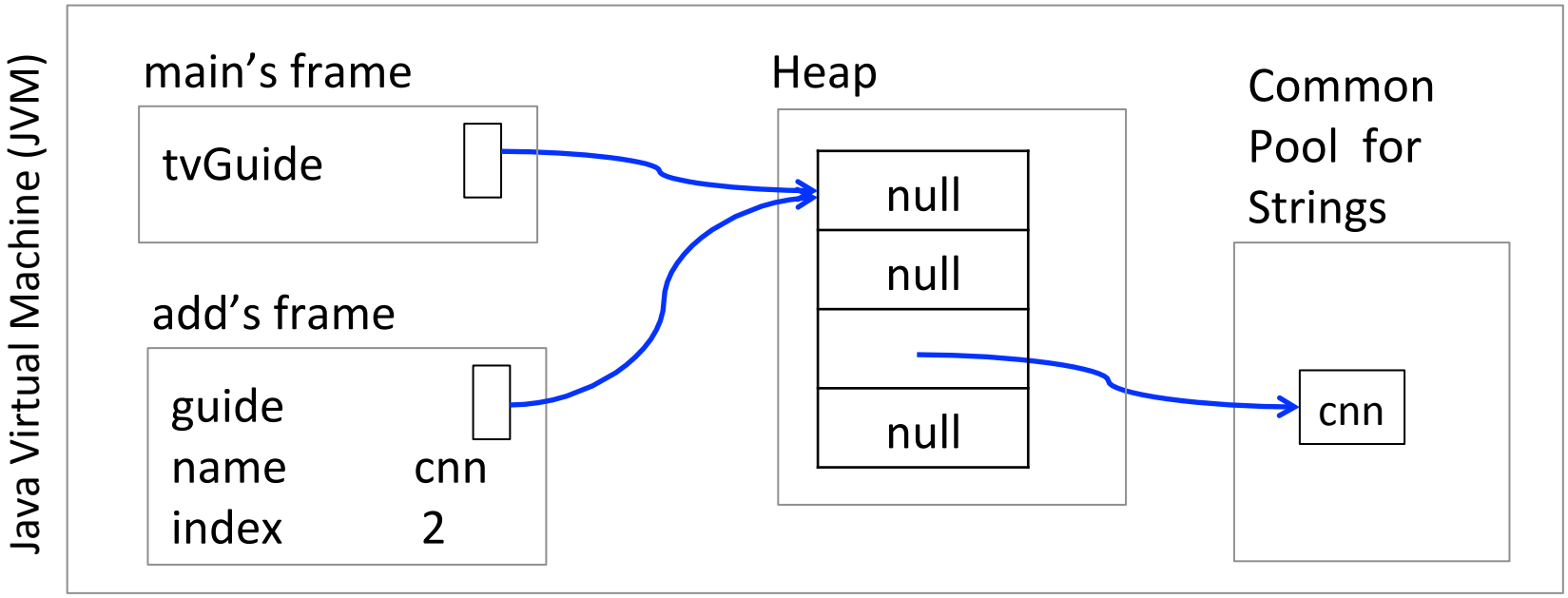
TV Guide: updating the array

```
public static void main(String[] args) {
    String[] tvGuide = new String[4];
    add(tvGuide, “cnn”, 2);
}
```



TV Guide: updating the array

```
public static void add(String[] guide, String name, int
index) {
        if (index >= 0 && index < guide.length) {
            guide[index] = name;
        }
}
```



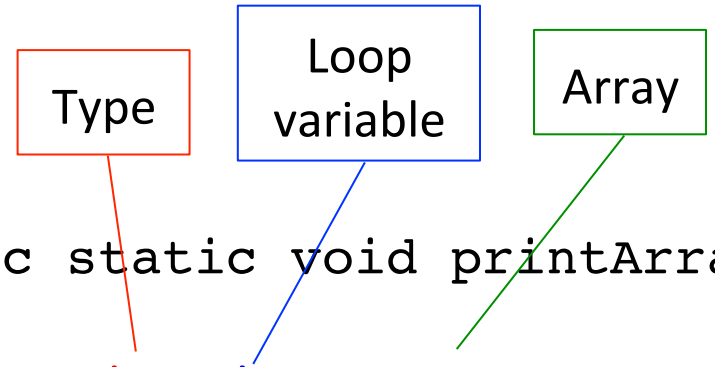
CS111
Introduction to Computer Science

Fall 2015

- For-each loop
- 2D Arrays

Loop: For-each

- A new way to loop through an array




```
public static void printArray (int[] array) {

    for (int i : array) {
        System.out.printf("%d ", i);
    }
}
```

2-Dimensional Arrays

- Suppose we want to store information based on two indexes
  - Scores of 3 students on 4 exams

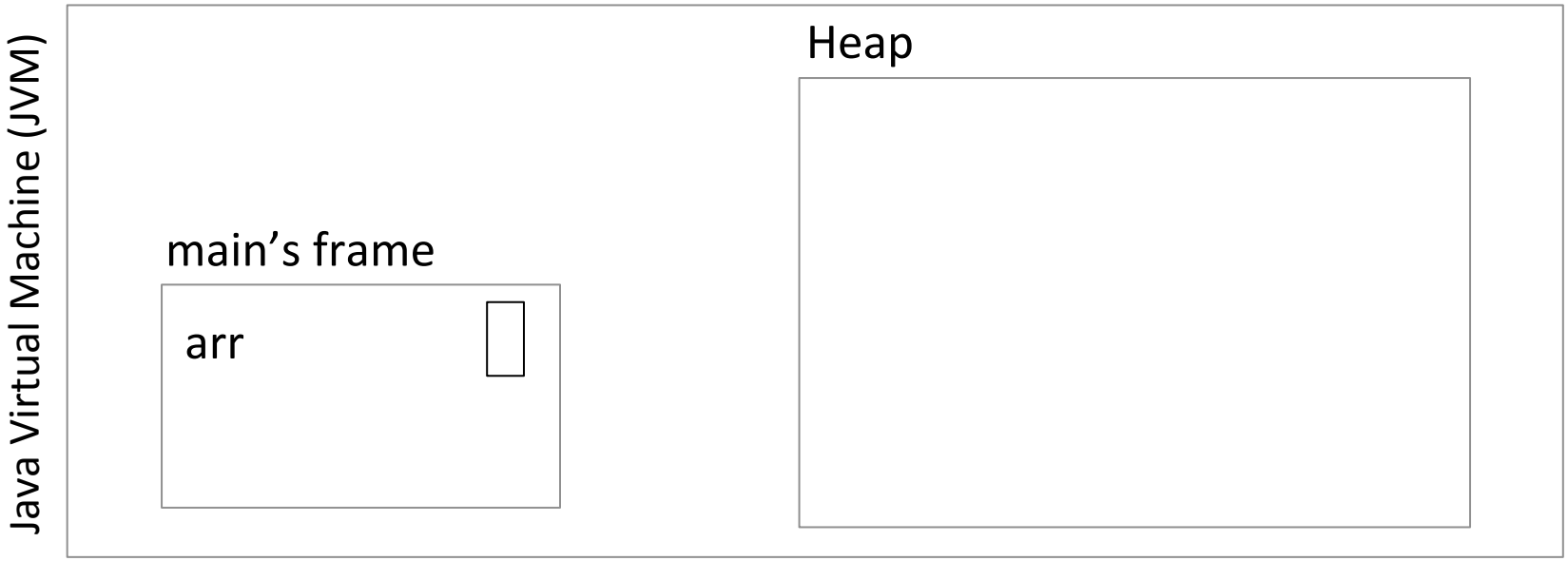
	Student 1	**Student 2**	**Student 3**
Exam 1	5.6	7.8	6.7
Exam 2	7.9	9.2	8.3
Exam 3	9.0	8.9	7.9
Exam 4	5.1	6.7	8.4

Declaring a 2-D Array in Java

- Declare

```
double[][] arr;
```

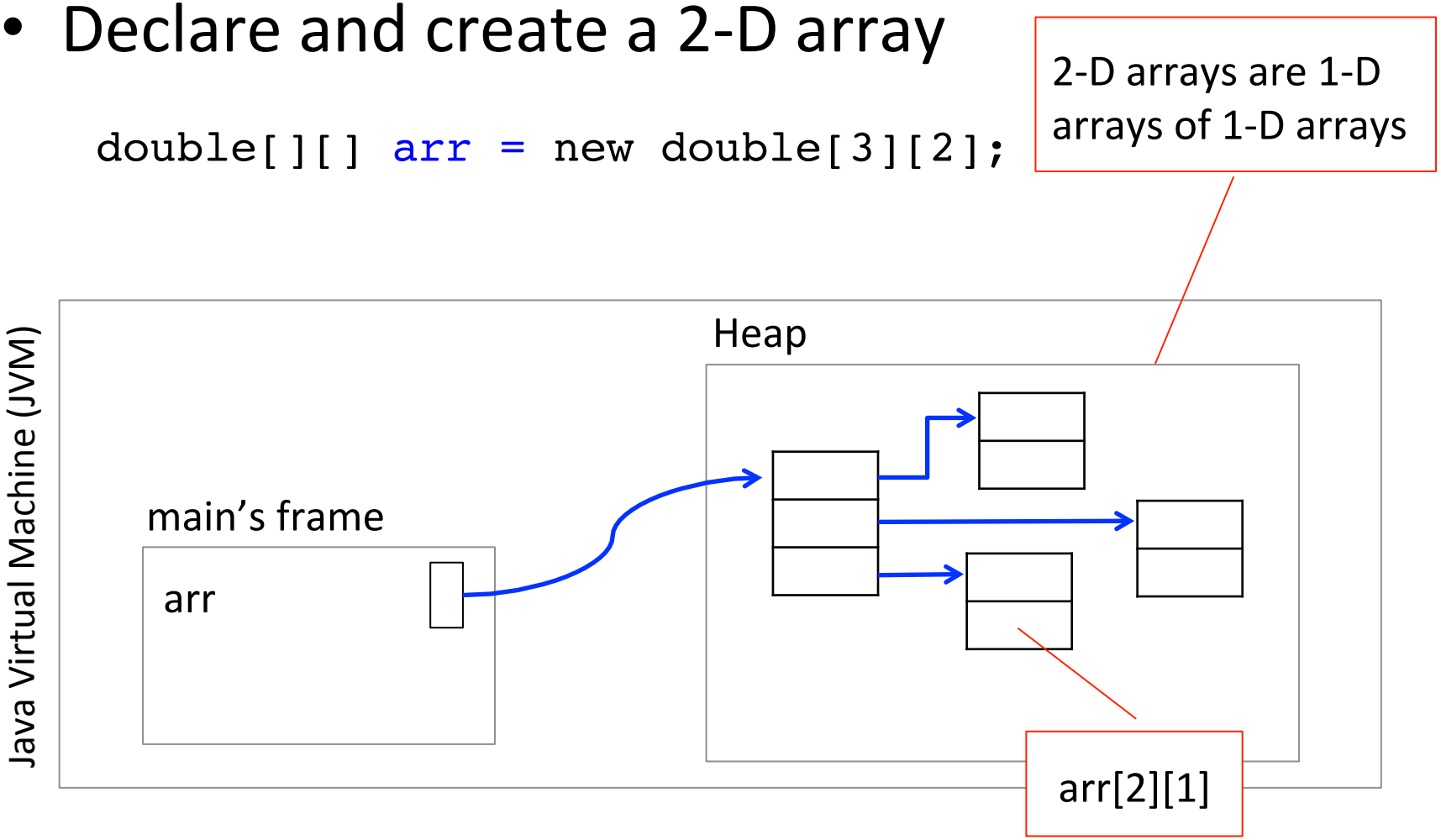
Creates a variable that will refer to a 2-D array of doubles



Creating a 2-D Array in Java

- Declare and create a 2-D array

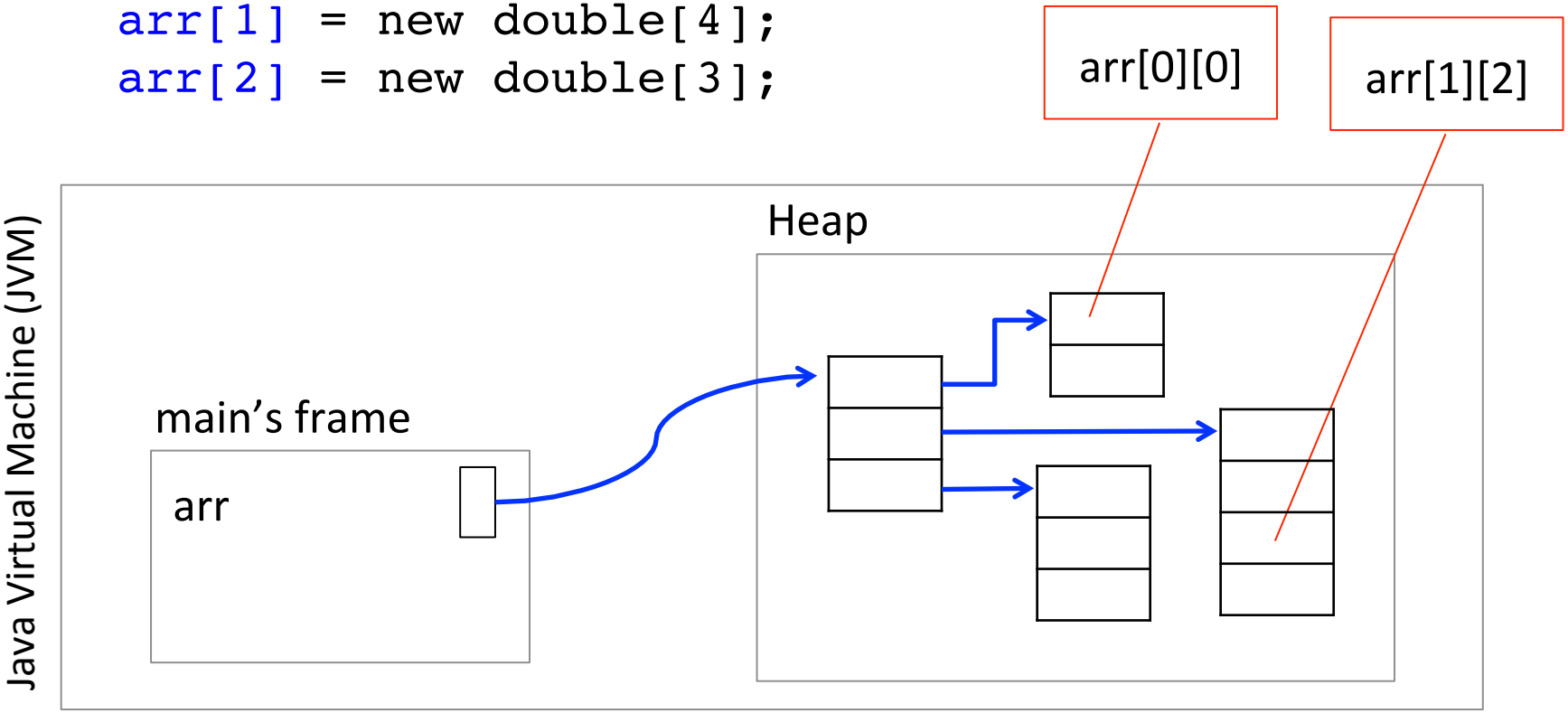
```
double[][] arr = new double[3][2];
```



Creating a 2-D Array in Java

- Declare and create a 2-D array

```
double[][] arr = new double[3][];
arr[0] = new double[2];
arr[1] = new double[4];
arr[2] = new double[3];
```



2-D Array Length

```
double[][] arr = new double[3][];
arr[0] = new double[2];
arr[1] = new double[4];
arr[2] = new double[3];
```

- arr.length
  - number of rows
  - number of 1-D arrays
- arr[1].length
  - second row number of columns

Iterating over a 2-D Array

```
public static void printArray (double[][] array) {

    for (int i = 0; i < array.length; i++) {
        for (int k = 0; k < array[i].length; k++) {
            System.out.println(array[i][k]);
        }
    }
}
```

Inner loop iterates over the columns

- use array[i].length to find the number of columns of each row

Outer loop iterates over the rows

- use array.length to find the number of rows

2-Dimensional Arrays

- Class of 3 students, each students takes 4 exams
  - average per student
  - average of each exam

	Student 1	Student 2	Student 3
Exam 1	5.6	7.8	6.7
Exam 2	7.9	9.2	8.3
Exam 3	9.0	8.9	7.9
Exam 4	5.1	6.7	8.4

See TwoDArray.java

CS111
Introduction to Computer Science

Fall 2015

- ArrayList

ArrayList Class

- The ArrayList class provides a dynamic array
  - stores elements of the same type sequentially
  - grows as needed (double its size)
  - useful if the size of the array is unknown until runtime
  - implements the List interface
- Declaring

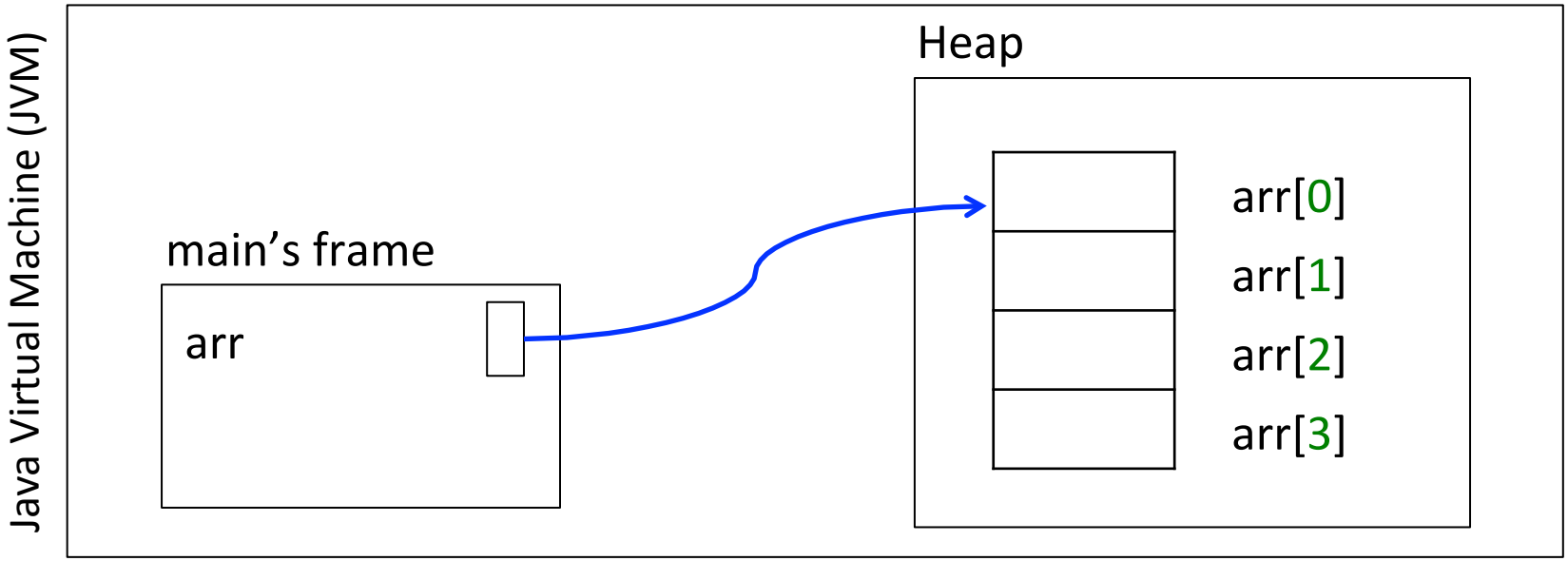
```
ArrayList <E> arrayReferenceVariable;
```

Element data type

Creating an ArrayList

- Declare and create an array list
  - creates an ArrayList object

```
ArrayList<String> arr = new ArrayList<String>(4);
```



ArrayList Methods

- Adding elements
  - arr.add(“apple”);
  - arr.add(0, “orange”);
- Retrieving elements
  - arr.get(0);
- Removing elements
  - arr.remove(0);
  - arr.remove(“orange”);

ArrayList Methods

- Find the index of an element
  - arr.indexOf(“orange”);
- Find the size of the array
  - arr.size();

See ArrayListUse.java