Arrays of Numbers

Programs often need to store multiple values of the same type, such as a list of phone numbers, or the names of your top 20 favorite songs. Rather than create a separate variable for each one, we can store them together using an array.

Manager:		Recorder:	
Presenter:		Reflector:	

Content Learning Objectives

After completing this activity, students should be able to:

- Explain course/school policies about academic honesty.
- Declare and initialize array variables of primitive types.
- Draw a memory diagram of an array of reference types.

Process Skill Goals

During the activity, students should make progress toward:

- Justifying answers based on evidence provided in the model. (Problem Solving)

Facilitation Notes

The case studies can be helpful to clarify policies and discuss what is appropriate collaboration vs cheating. Each team will need a copy of your school's Honor Code (or similar document). Only 15 minutes are allocated for this discussion, but it could last longer if desired.

Model 1 introduces arrays for the first time, with a focus on array syntax. Students learn about declaring, initializing, and using array variables. Be sure to discuss #12 and #13 when reporting out. Explain how Java generally requires the `new` operator when creating arrays, except in the case of #12.

In Model 2, explain that the `new` operator automatically zeros-out memory for the array. Therefore, the default value will be 0 for integers, 0.0 for doubles, `null` for strings, etc. These values should be present in the team's diagrams. You may need to remind students that references (i.e., to string objects) are drawn with an arrow.

Key questions: #13, #17, #18

Case Study: Panic Attack

Frank was behind in his programming assignment. He approached Martin to see if he could get some help. But he was so far behind and so confused that Martin just gave him his code with the intent that he would "just look at it to get some ideas." Frank copied much of the code and turned it in as his own.

In the paraphrased words of Frank: "I started the assignment three days after you put it up. But then other assignments came in and I started on them too. I felt like I was chasing rabbits and began to panic. It was already past the due date and I got really scared. That's when I went to Martin to see if he could help."

Questions (7.5 min) Start time:

1. Which, if any, of the students were at fault? Why?

Both students are at fault, assuming the assignment was supposed to be done individually. It's not okay to look at someone else's code and/or give your code to someone else.

2. Which specific Honor Code violations occurred?

[Frank] Collaborating in an unauthorized manner with one or more students on any work submitted for academic credit.

[Martin] Rendering unauthorized assistance to another student by knowingly permitting him to see a portion of work to be submitted for academic credit.

3. What should Martin have done in this situation?

This type of situation often occurs when students feel pressure before a deadline. Martin should have refused to show his code to Frank and reminded him to get help from the instructor and/or a TA.

4. What options did Frank have besides cheating?

Frank could have asked the instructor for help during office hours, met with a TA during lab hours, posted a question on Piazza, or asked a question during class. It's much easier to get help when it's not the night before.

Case Study: Oops!

Emily was working in the lab on her programming assignment. She finished the program, submitted it, and went on to do some other work. Shortly thereafter, she left the lab.

Another student, Kyle, was working nearby. He knew that Emily had successfully submitted the assignment, and he had not been able to get his to work properly. When Emily left, he noticed that she had not logged out of her computer. He moved to her workstation, found the work under her Documents directory, and copied it onto his flash drive. He then logged out, logged in as himself, and copied the code onto his Desktop where he modified the program a bit, then successfully submitted it.

Questions (7.5 min) **Start time:**

5. Which, if any, of the students were at fault? Why?

Kyle is certainly at fault, and depending on the lab policy, Emily may also be penalized. But in general, students are not held accountable when files are stolen without their knowledge.

6. Which specific Honor Code violations occurred?

[Kyle] Committing the act of plagiarism: copying information, ideas, or phrasing of another person without proper acknowledgment of the true source.

[Kyle] Using computing facilities in an academically dishonest manner.

7. What should Emily have done in this situation?

To avoid this situation, Emily should have logged out or at least locked her screen when leaving the room.

8. What options did Kyle have besides cheating?

Kyle could have asked the instructor for help during class or office hours, met with a TA during lab hours, or posted a question on Piazza.

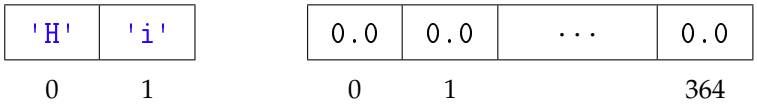
Model 1 Array Syntax

An ***array*** variable allows you to store multiple variables (of the same type). Each value in an array is known as an ***element***. The programmer must specify the ***length*** of the array (the number of array elements). Once the array is created, its length cannot be changed.

```
char[] letterArray = {'H', 'i'};
System.out.println(letterArray[0]);          // outputs H
System.out.println(letterArray.length);      // outputs 2

double[] numberArray = new double[365];
System.out.println(numberArray[0]);          // outputs 0.0
System.out.println(numberArray.length);      // outputs 365
```

Array elements are accessed by ***index*** number, starting at zero:



Questions (15 min) Start time:

9. Examine the results of the code.

a) What is the length of `letterArray`? 2

b) What is the length of `numberArray`? 365

c) What is the index of the element `'i'` in `letterArray`? 1

d) What is the index of the last element of `numberArray`? 364

10. Now examine the syntax of the code.

a) What are three ways that square brackets [] are used?

1) To declare the type: `double[]`
2) To specify the length: `double[365]`
3) To access an element: `numberArray[0]`

b) In contrast, how are curly braces {} used for an array?

To create an array with an initial set of values.

11. What are the resulting type and value of the following expressions? Show your work by writing the value of each array element in the space provided.

```
int[] a = {3, 6, 15, 22, 100, 0};
double[] b = {3.5, 4.5, 2.0, 2.0, 2.0};
String[] c = {"alpha", "beta", "gamma"};
```

a) `a[3] + a[2]` Type: `int` Value: `37`

`22` `15`

b) `b[2] - b[0] + a[4]` Type: `double` Value: `98.5`

`2.0` `3.5` `100`

c) `c[1].charAt(a[0])` Type: `char` Value: `'a'`

`beta` `3`

d) `a[4] * b[1] <= a[5] * a[0]` Type: `boolean` Value: `false`

`100` `4.5` `0` `3`

As shown in #11, an array variable can be declared and initialized without using `new`. However, to assign an array variable that was previously declared, `new` is required:

```
a = new int[] {3, 6, 15, 22, 100, 0};
c = new String[] {"alpha", "beta", "gamma"};
```

12. Write statements that declare and initialize variables for the following arrays.

a)

0	14	1024	127	3	5521

```
int[] a = {0, 14, 1024, 127, 3, 5521};
```

b)

3.23	1.52	4.23	32.5	2.45	5.23	3.33

```
double[] b = {3.23, 1.52, 4.23, 32.5, 2.45, 5.23, 3.33};
```

13. Write statements that assign the following arrays to variables you declared in #12.

a)

0	14	1024	127	3	5521

```
a = new int[] {0, 14, 1024, 127, 3, 5521}
```

b)

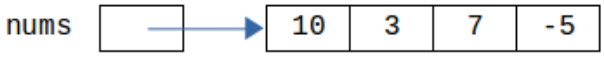
3.23	1.52	4.23	32.5	2.45	5.23	3.33

```
b = new double[] {3.23, 1.52, 4.23, 32.5, 2.45, 5.23, 3.33}
```

Model 2 Array Diagrams

Array elements are stored together in one contiguous block of memory. To show arrays in memory diagrams, we simply draw adjacent boxes.

```
int[] nums = {10, 3, 7, -5};
```




```
double[] stats = new double[3];
```

Questions (15 min) Start time:

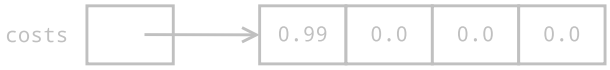
14. What is the default value for array elements?

Zero or equivalent value, depending on the data type. For numeric types like `int` and `double`, the default is 0; for `boolean`, it's `false`; for `char`, it's `'\u0000'`; for reference types, it's `null`.

15. Draw a memory diagram for the following arrays. (*Hint:* You should have no empty boxes.)

a)
```
int[] sizes = new int[5];
sizes[2] = 7;
```

b)
```
double[] costs = new double[4];
costs[0] = 0.99;
```



c)
```
String[] names = {"Anita"};
```

16. Like strings, arrays are reference types. What is the *value* of an array variable?

An integer representing the memory location of the array.

17. Does the statement `int[] copy = nums;` create a new array? Justify your answer.

No. If you assign one array variable to another, you're only copying the reference, not the array itself.

18. Draw a memory diagram of the following array. (*Hint:* You should have four arrows.)

```
String[] greek = {"alpha", "beta", "gamma"};
```