What could go wrong here?

```java
int table[][] = new int[2][3];
// some code
int sum = 0;
for (int i = 0; i < 2; i++)
    for (int j = 0; j < 3; j++)
    {
        int value = table[i][j];
        sum += value;
    }
```

Array Definition

- A data structure for storing a fixed number of elements.
- Elements are of the same type and share the same name.
- Each element is accessed using its relative position in the array.

Array of four elements:

```
0 1 2 3
```

Motivation

Write a program that computes the average of three numbers.

```java
import java.util.Scanner;
// ...
Scanner in = new Scanner(System.in);
int num1 = in.nextInt(); // handle InputMismatchException
int num2 = in.nextInt();
int num3 = in.nextInt();
int total = num1 + num2 + num3;
double average = total / 3;
```
Set up an array

You have to tell Java what kind of data is going into your array, and how many elements the array has.

dataType name[] = new dataType[size];

Type of an array and its elements

Common Mistakes

boolean status[] = new boolean[3];
status[3] = true;  Array Index Out of Bounds Exception

Arrays of Primitives

boolean status[];
status = new boolean[3];
status[2] = true;

null
false false false
false false true

int nums[] = new int[4];
nums[0] = 1;

null
1 2 3 0
2 3 0 1

Common Mistakes

int nums[] = new int[4];
nums[2] = true;  Type mismatch error
Common Mistakes

Number nums[] = new Float[2];
nums[0] = new Integer(1);  // Array Store Exception

Array Initialization

int nums[] = new int[4];
nums[0] = 4;
nums[1] = 1;
nums[2] = 3;
nums[3] = 2;

Array Initialization

int nums[] = new int[4];
nums[0] = 4;
nums[1] = 1;
nums[2] = 3;
nums[3] = 2;

Common Mistakes

int nums[];
nums = {4, 1, 3, 2};  // Array constants can only be used in initializers
### Common Mistakes

```java
int nums[];
nums = {4, 1, 3, 2}; // Array constants can only be used in initializers
```

```java
void f(float[] input) { ... }
void f(double[] input) { ... }
void g()
  f([1, 2, 3, 4]);
Which function to call?
```

### Arrays of Objects

```java
Car garage[] = new Car[4];
garage[0] = new Car();
garage[0].color = white;
Car car4 = new Car();
car4.color = black;
cars[3] = car4;
```

### Multi-dimensional Arrays

Multidimensional arrays are arrays of arrays with each element of the array holding the reference of another array.

```java
dataType name[][] = new dataType[]...[];
```

Examples:
- Board games, Spreadsheets, ...

### Multi-dimensional Arrays (2D)

Multidimensional arrays (2D)

```java
int matrix[][] = new int[3][3];
```

1-D

2-D

```java
[1][2]
```
Array Traversal

String row[] = new String[2];

for (int i = 0; i < row.length; i++)
{
    // assign or read "row[i]"
}

for (String s : row)
{
    // read "s"
}

Array Traversal (2D)

int table[][] = new int[2][3];

// some code
int sum = 0;
for (int i = 0; i < table.length; i++)
    for (int j = 0; j < table[i].length; j++)
        int value = table[i][j];
    sum += value;

Ragged Arrays

int part[][] = {{1, 2}, {3}, {4, 5, 6}};

int part[][] = new int[3][];
part[0] = new int[2];
part[1] = new int[1];
part[2] = new int[3];

Array Copy

int[] list_A = {1, 3, 5, 7};
int[] list_B = new int[2];

// list_B = list_A;
// System.out.println(list_B[3]);
list_B[1] = -1;
// System.out.println(list_A[1]);
Array Copy

```java
int[] list_A = {1, 3, 5, 7};
int[] list_B = new int[2];

list_B = list_A;
System.out.println(list_B[3]); // 7
list_B[1] = -1;
System.out.println(list_A[1]); // -1
```

Array Comparison

```java
int[] list_A = {1, 3, 5, 7};
int[] list_B = {1, 3, 5, 7};

if(list_A == list_B){...}
or
if(list_A.equals(list_B)){...}

Are they referring to the same array?

To determine whether two arrays contain the same elements, compare them element by element.
```

Passing and returning arrays

```java
int numOfInputs = scanner.nextInt();
int[] input = new int[numOfInputs];
int sum = sum(input);
double average = sum / numOfInputs;
```

```java
public int[] getInput(int num) {
    int[] input = new int[num];
    for(int i=0; i < num; i++){
        input[i] = scanner.nextInt();
    } return input;
}
```

```java
public int sum(int[] nums) {
    int result = 0;
    for(int i : nums){
        result += i;
    } return result;
}
```

```java
public int[] copyOf;
equals
depthEquals
```

java.util.Arrays

- fill
- sort
- binarySearch
- copyOf
- equals
- deepEquals
What you should know

• What an array is
• Array declaration, instantiation, and initialization
• Arrays of primitives and arrays of objects
• Multi-dimensional arrays
• Traversing arrays
• Passing arrays to methods and returning arrays from methods

Exercise 1

Given a text corpus, compute bi-grams and record their frequencies.

For example: “this is some text”
	his is
is some
some text

Exercise 2

A 3x3 Sudoku puzzle: The user should place the numbers 1 to 9 such that each row, column, and diagonal adds up to the same number.

\[
\begin{array}{ccc}
8 & 1 & 6 \\
3 & 5 & 7 \\
4 & 9 & 2 \\
\end{array}
\]

N.B. each number should be used exactly once.