Arrays

The Key Problem

You need to store many related values.

- e.g. read in 10 numbers from the user, then print them in sorted order
- e.g. count how often each sum 2, 3, 4, …, 12 is rolled when rolling a pair of dice

Separate variables for each value would be tedious.

Two Additional Wrinkles

You may not know until runtime how many values there will be.

- e.g. read in numbers until the user enters a 0, then print them in sorted order

The number of values may change as the program runs or from one running of the program to the next.

- e.g. a game for 2-6 players
- e.g. a game where players are eliminated one by one
- e.g. the cards in your hand in a game like hearts or spades

Arrays

Arrays are like an ornament storage box.

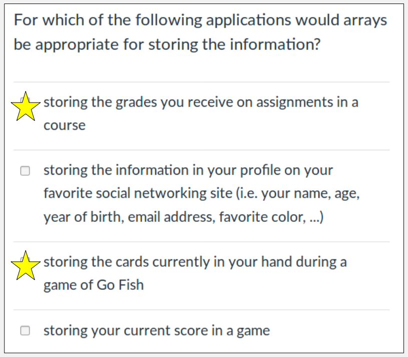
1D

2D

3D

The box – the variable – is subdivided into compartments, each of which can hold a single value at a time.

- each compartment must hold the same kind of value (the *base type*) of the array



For which of the following applications would arrays be appropriate for storing the information?

- ★ storing the grades you receive on assignments in a course
- □ storing the information in your profile on your favorite social networking site (i.e. your name, age, year of birth, email address, favorite color, ...)
- ★ storing the cards currently in your hand during a game of Go Fish
- □ storing your current score in a game

Two key tests for identifying when arrays are appropriate:

- *quantity test* – are there many things?
- *similarity test* – are the things the same kind of thing (both Java type and concept)

Must answer yes to both questions.

Working With Arrays

With (regular) variables, there are two steps before a variable can be used –

- declaration – make the box
  ```
  int x;
  ```
- initialization – put a value in the box
  ```
  x = 10;
  ```

```
int x = 10;
```

With array variables, there are three steps –

- declaration – make the box
  ```
  int[] a;
  ```
- make the compartments within the box
  ```
  a = new int[5];
  ```

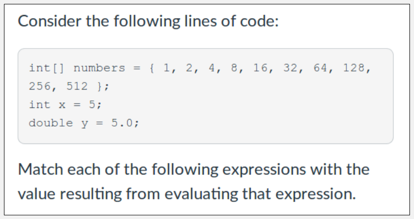
```
int[] a =
  new int[5];
```

- initialize each compartment
  ```
  for ( int i = 0 ; i < a.length ; i++ ) {
    a[i] = 10;
  }
  ```

Working With Arrays

Usage –

- the variable `name` refers to the whole box
  - `name.length` refer to the number of compartments
- `name[i]` refers to compartment *i*
  - indexing starts at 0



Consider the following lines of code:

```
int[] numbers = { 1, 2, 4, 8, 16, 32, 64, 128,
256, 512 };
int x = 5;
double y = 5.0;
```

Match each of the following expressions with the value resulting from evaluating that expression.

numbers[0]	1
numbers[4]	16
numbers[10]	runtime error
numbers[x]	32
numbers[y]	syntax error
numbers.length	10