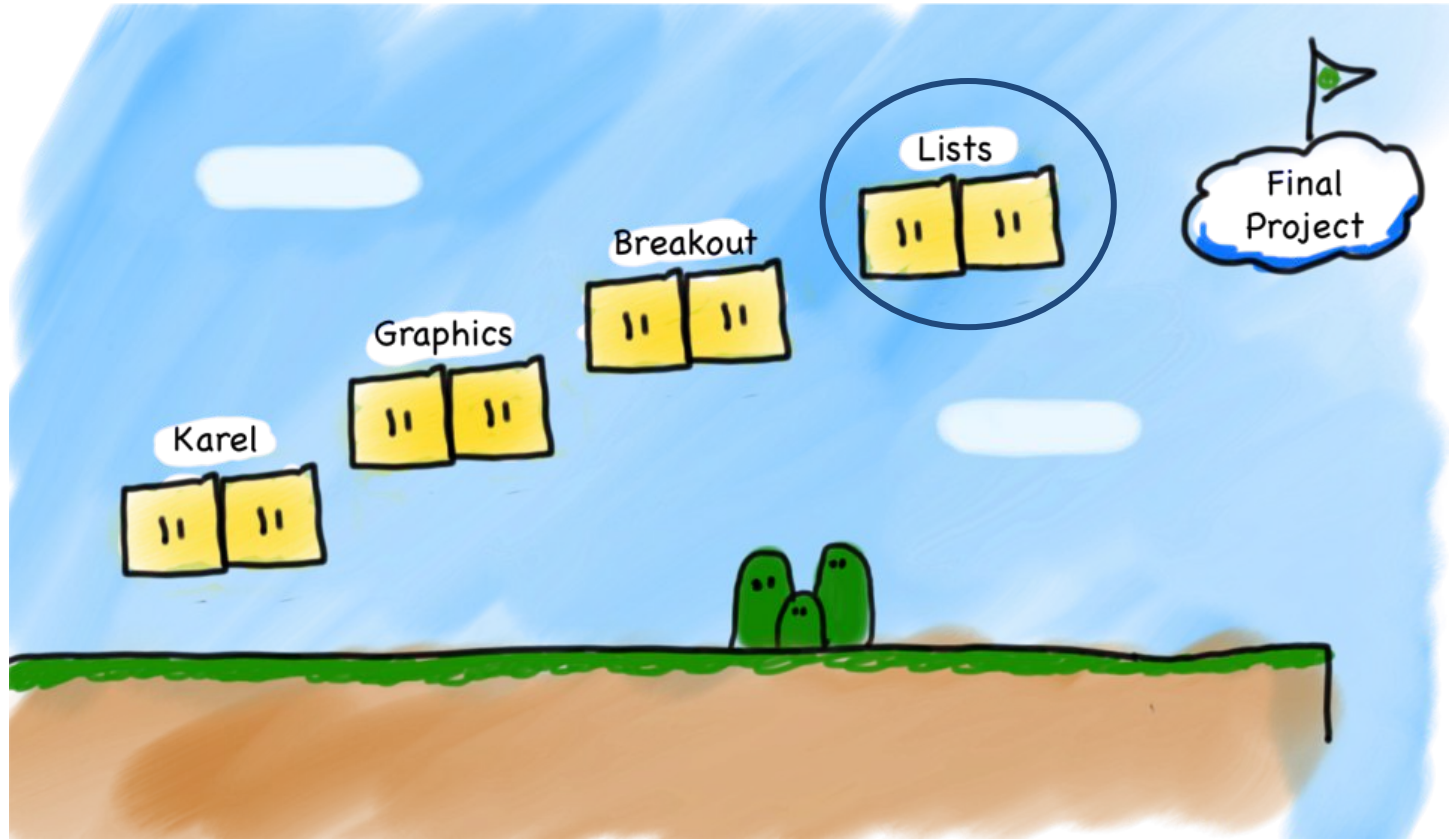




Arrays

Where are we?

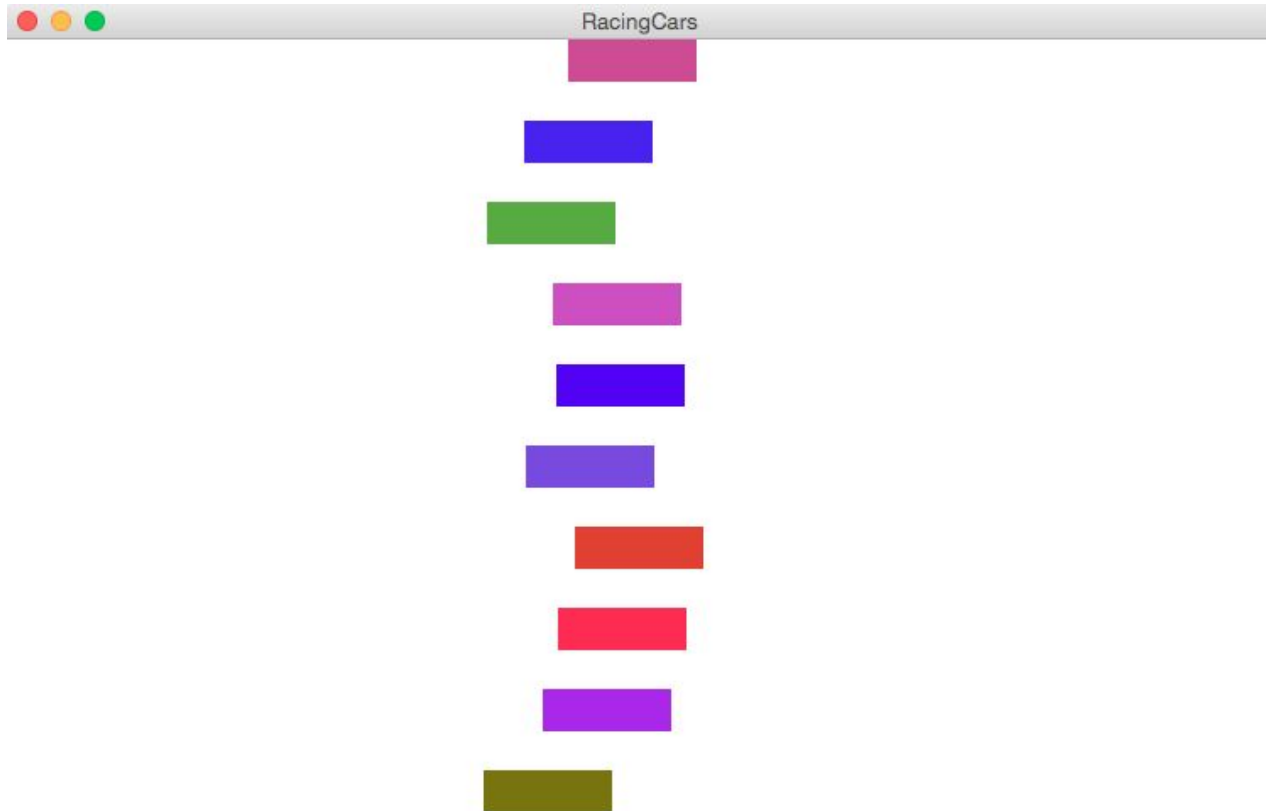


Learning Goals

1. Understand array data structures
2. Use arrays in a program



Racing Cars!

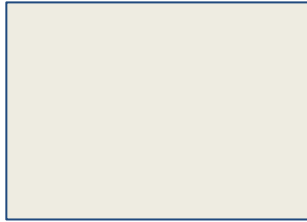


What are Data Structures?

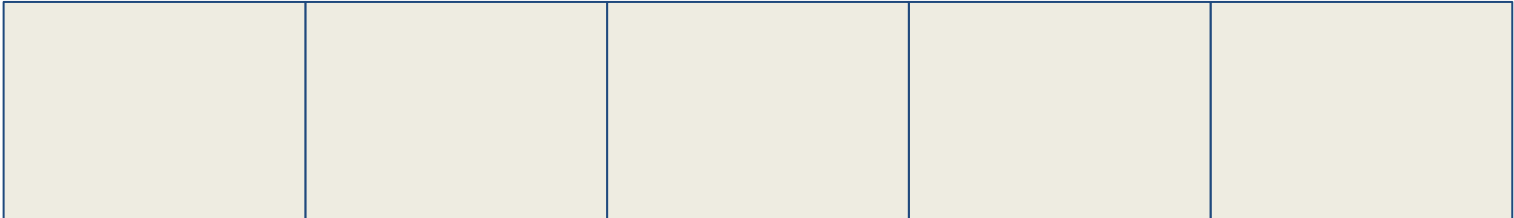
Data structures are variable types that can store data in interesting ways.

What are Arrays?

1 box = a variable

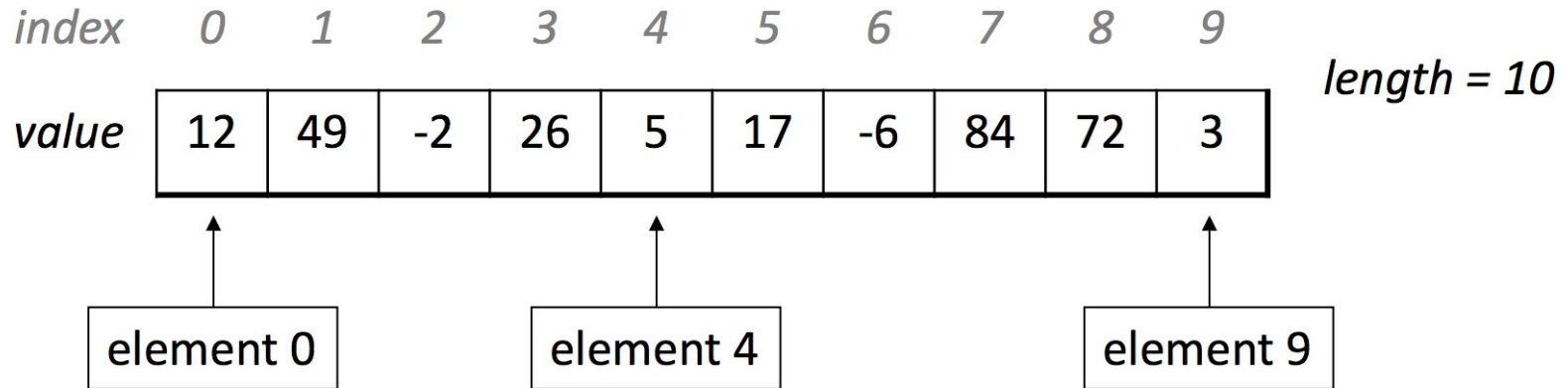


many connected boxes = an array



What are Arrays?

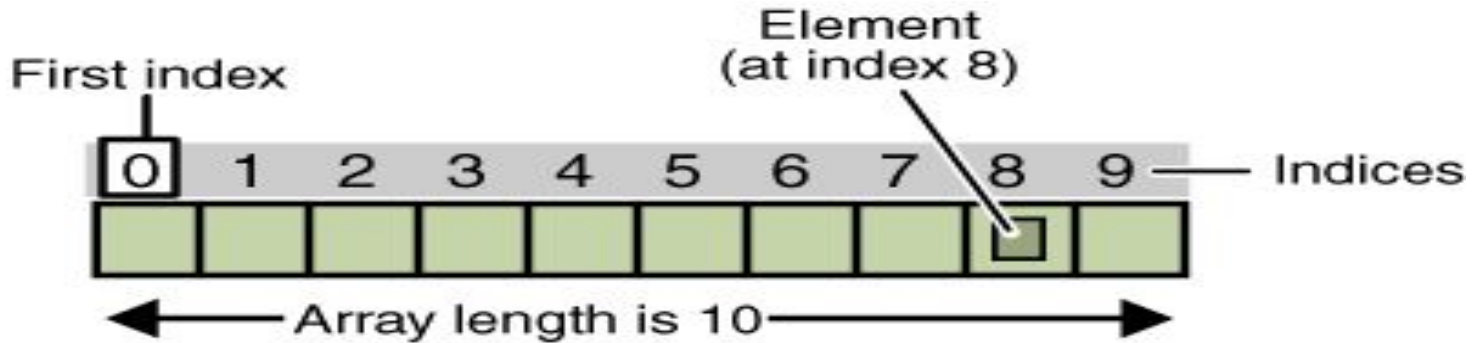
Arrays have *elements* that you can access using *indices*.





Arrays: Basic properties

1. *An array is ordered.(considering indices not the contents)*
2. *An array is homogeneous.*



Declaring an Array variable

```
type[] name = new type[n];
```

```
int[] intArray = new int[10];
```

intArray

0	0	0	0	0	0	0	0	0	0
0	1	2	3	4	5	6	7	8	9

Array selection

- You can, for example, select the initial element by writing

```
intArray[0]
```

- Assigning a value to an element

```
intArray[9] = 42;
```

intArray

0	0	0	0	0	0	0	0	0	42
0	1	2	3	4	5	6	7	8	9

Cycling through array elements

- Cycling through each of the array elements

```
for (int i = 0; i < array.length; i++) {  
    Operations involving the  $i^{\text{th}}$  element of the array  
}
```

- As an example, you can reset every element in `intArray` to -1 using the following `for` loop:

```
for (int i = 0; i < intArray.length; i++) {  
    intArray[i] = -1;  
}
```

Initializing arrays

- Java makes it easy to initialize the elements of an array as part of a declaration. The syntax is

```
type[] name = { elements } ;
```

- For example, the following declaration initializes the variable **powersOfTen** to the values 10^0 , 10^1 , 10^2 , 10^3 , and 10^4 :

```
int[] powersOfTen = { 1, 10, 100, 1000, 10000 } ;
```

This declaration creates an integer array of length 5 and initializes the elements as specified.

Initializing arrays

- In Java, an array's length must be specified on creation and cannot be modified afterwards.



```
int[] powersOfTen = new int[];
```



Variable must provide either dimension expressions or an array initializer



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



```
int[] powersOfTen = { 1, 10, 100, 1000, 10000 };
```

Exercise: Statistics!

How many values would you like to input? 5

-5

3.1415926535

0

88

12.3

input:

-5.0

3.1415926535

0.0

88.0

12.3

Sum: 98.4415926535

Mean: 19.6883185307

Max: 88.0

|

Array Types

- You can make an array of any other type!

```
int[] intArray = new int[10];
```

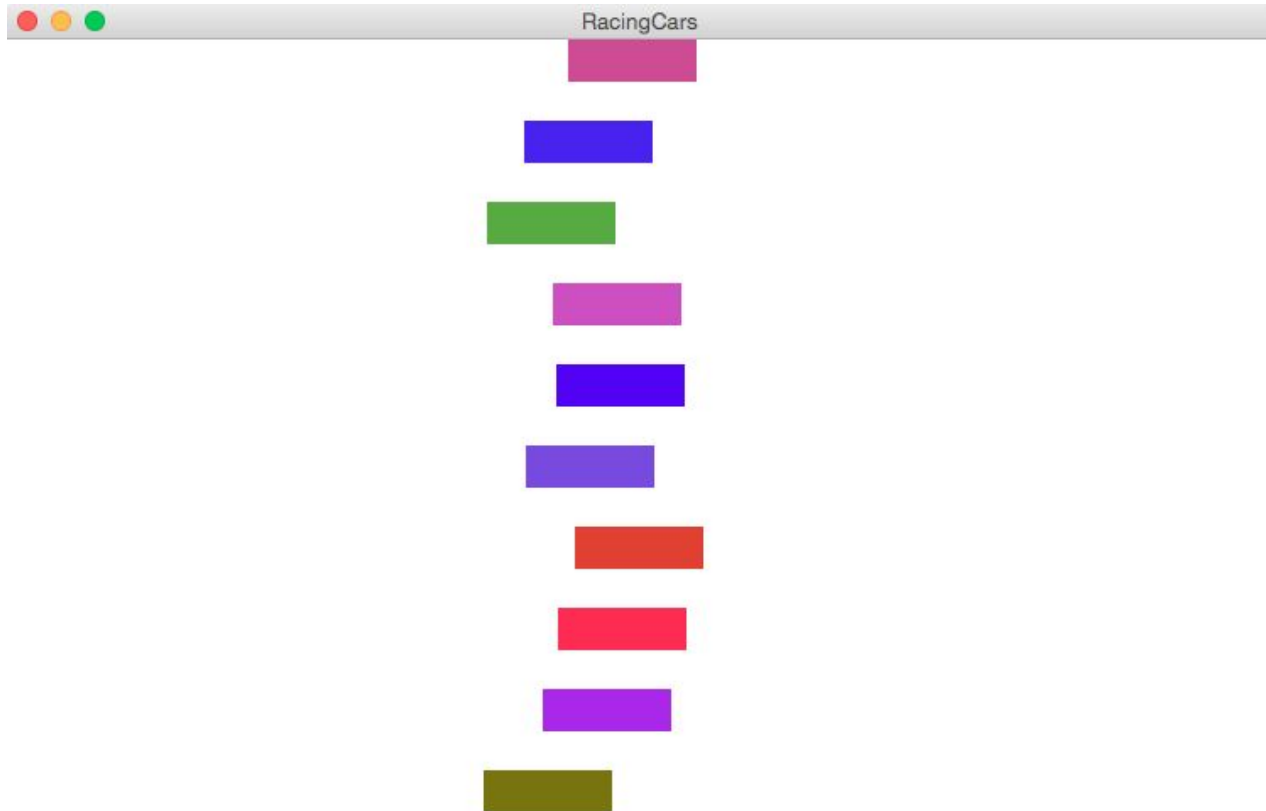
```
char[] charArray = new char[10];
```

```
Color[] colorArray = new Color[10];
```


```
GRect[] grectArray = new GRect[10];
```

```
int[][] arrayArray = new int[10][10];
```


An array of graphical objects



Creating an array of objects

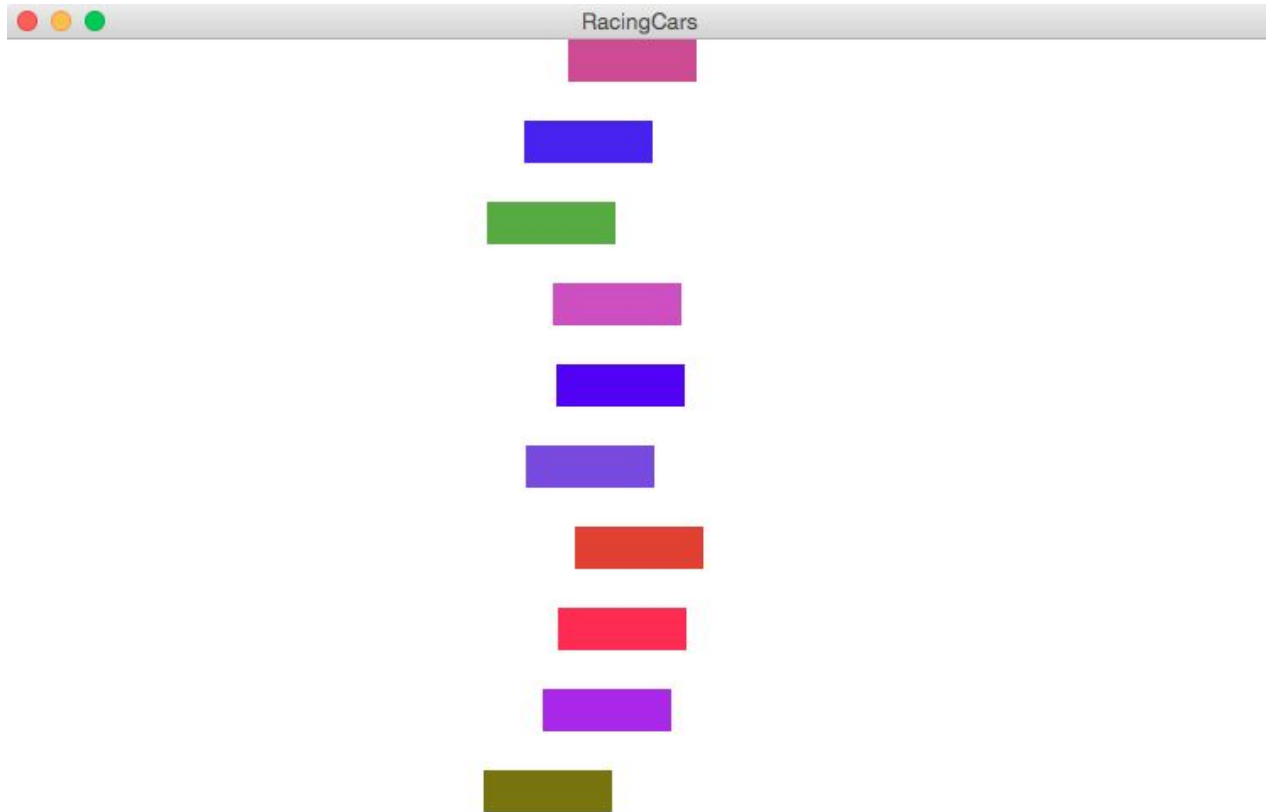


```
RacingCars

private final int CAR_WIDTH=75;
private final int X_OFFSET=50;
private final int NUM_CARS=50;

public void run() {
    double carHeight=getHeight() / (2*NUM_CARS);
    GRect[] cars=new GRect[NUM_CARS];
    for(int i=0;i<NUM_CARS;i++){
        cars[i]=new GRect(CAR_WIDTH,carHeight);
        cars[i].setColor(rgen.nextColor());
        cars[i].setFilled(true);
        add(cars[i],X_OFFSET,(2*i)*carHeight);
    }
}
```

Animating an array of objects



Summary

- An array is an *ordered, homogeneous* list of data.
- Arrays can store both primitives (int, double, char, etc) and objects (CGRect, GLine, etc)
- Array indices start at 0.
- **An array's length cannot be changed once it is created.**

Extra: Array Challenges

- Reversing
- Sorting
- 2-D Arrays