CSE 503

Introduction to Computer Science for Non-Majors

Dr. Eric Mikida
epmikida@buffalo.edu
208 Capen Hall

Day 10
Ranges, Sequences and Loops

Announcements

- Starting next week, my Tues office hours will be on Thurs instead
  - For this class, it may help to talk to me ahead of time (but not necessary)

Recap

- **Arrays:** Objects that can hold multiple values
  - We can store their address in a variable, ie: `let x = [1, 2, 3, 4];`
  - Element access uses `[ ]`, ie `x[0]`, `x[3]`, etc.
    - Indexing starts at 0
  - Length can be access via `.length`, ie `x.length`
- **For Loops:** A control flow construct that let's us repeat a group of statements multiple times…more on this shortly

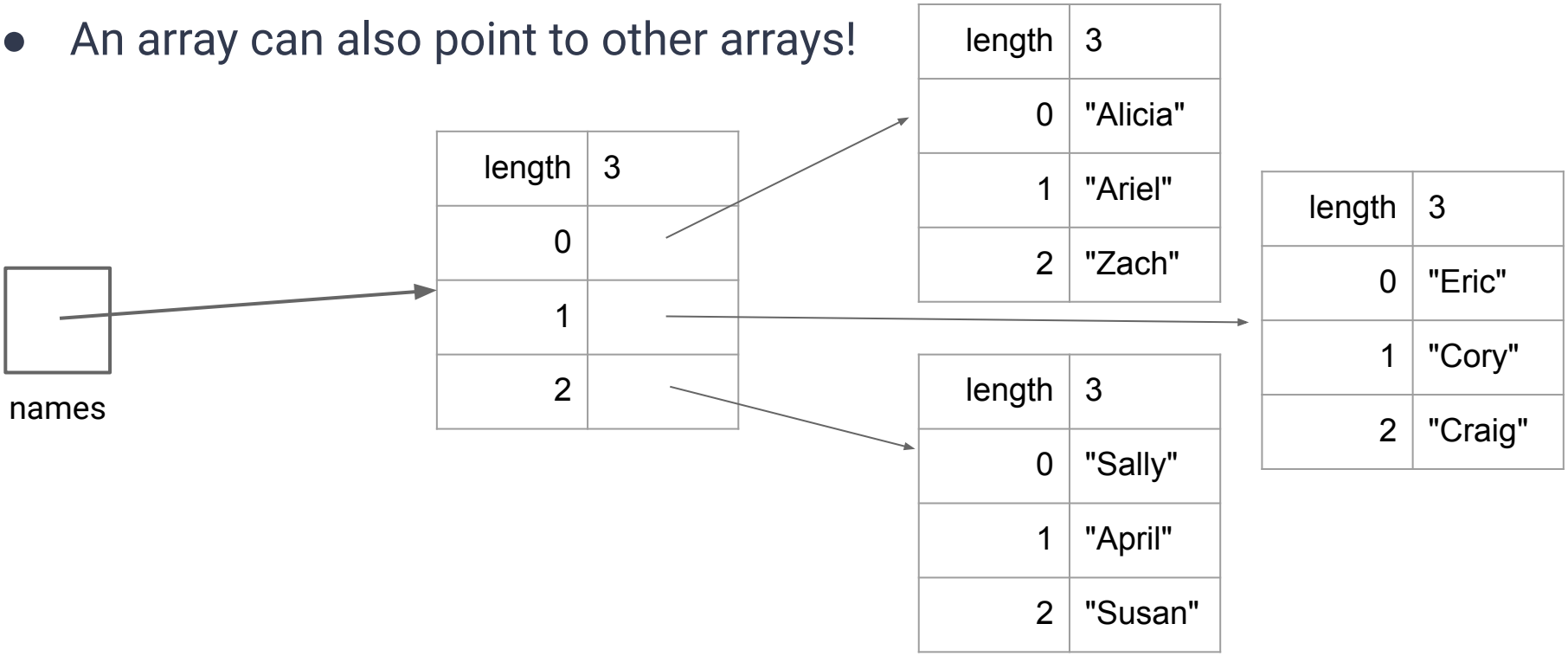
2D Arrays

- Last Time we saw that the address of an Array can be stored in a variable. ie, we can treat the address as a value.
- An array stores many values…
  - …so it stands to reason that…

length	5
0	"Alicia"
1	"Ariel"
2	"Zach"
3	"Cory"
4	"Kate"

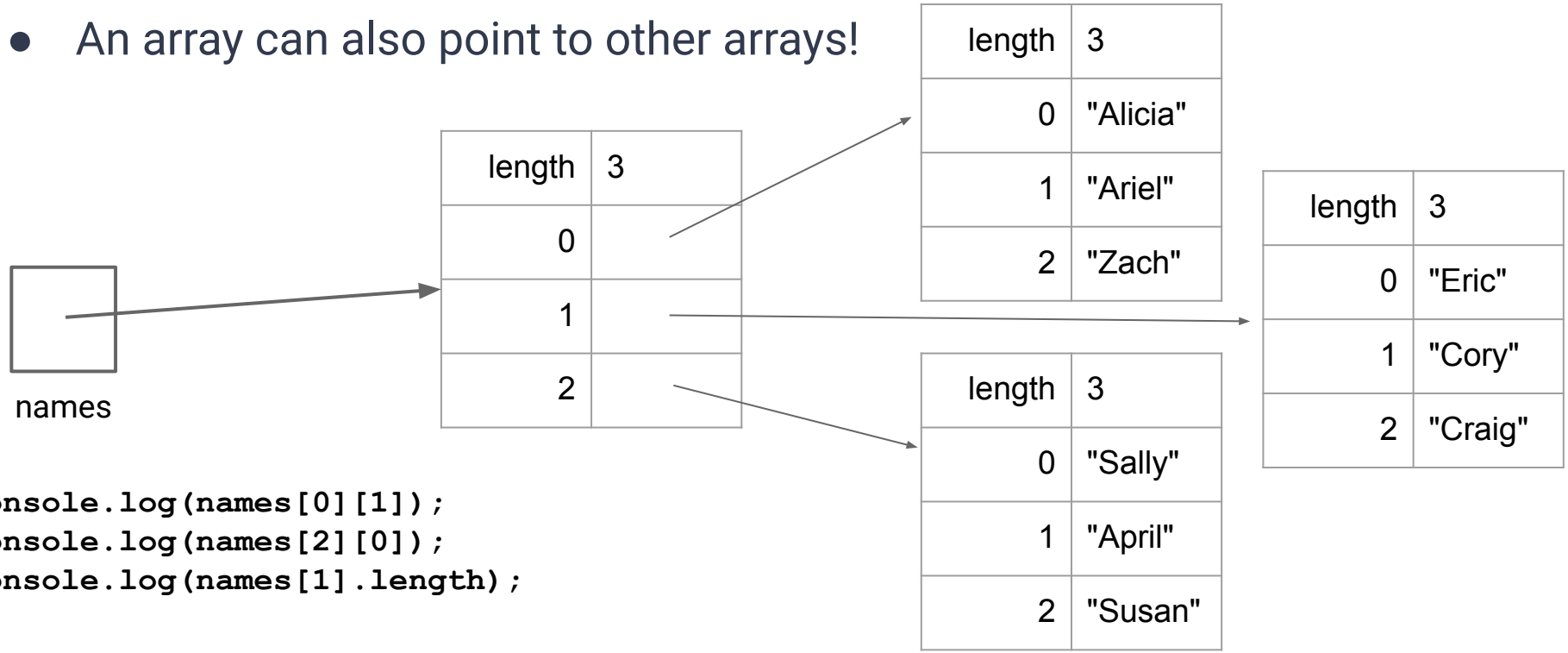
2D Arrays

- An array can also point to other arrays!



2D Arrays

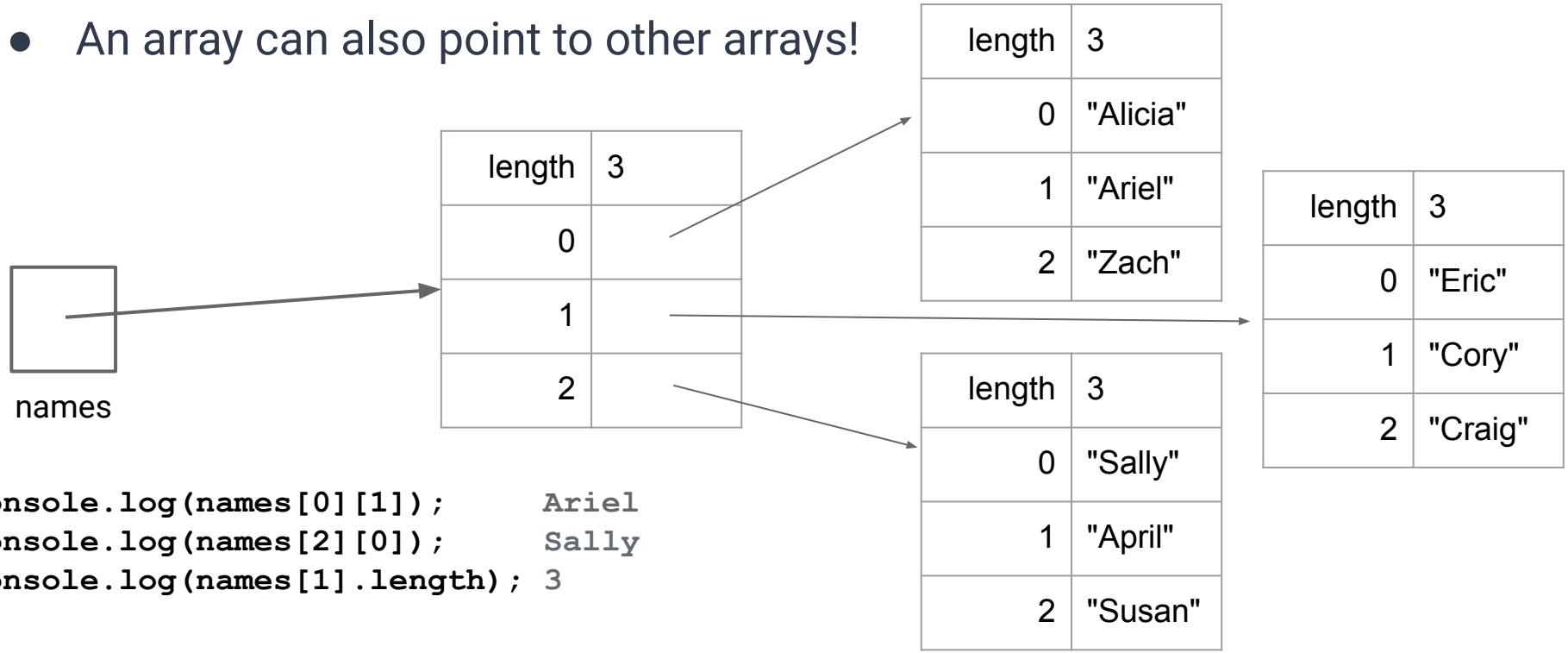
- An array can also point to other arrays!




```
console.log(names[0][1]);
console.log(names[2][0]);
console.log(names[1].length);
```

2D Arrays

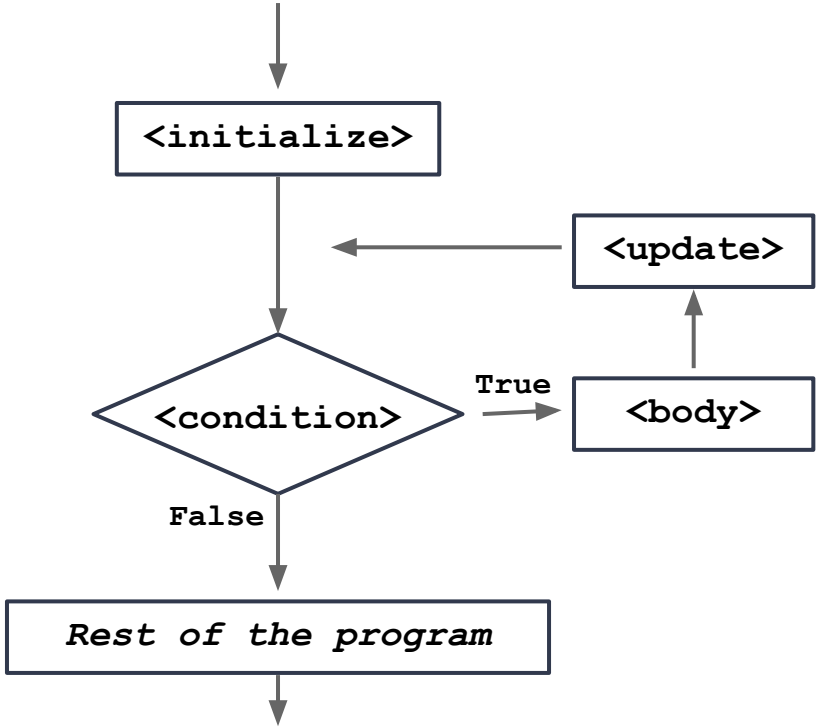
- An array can also point to other arrays!




```
console.log(names[0][1]);      Ariel
console.log(names[2][0]);      Sally
console.log(names[1].length);  3
```

For Loop Review

```
for (<init>; <cond>; <update>) {
    <body>
}
```

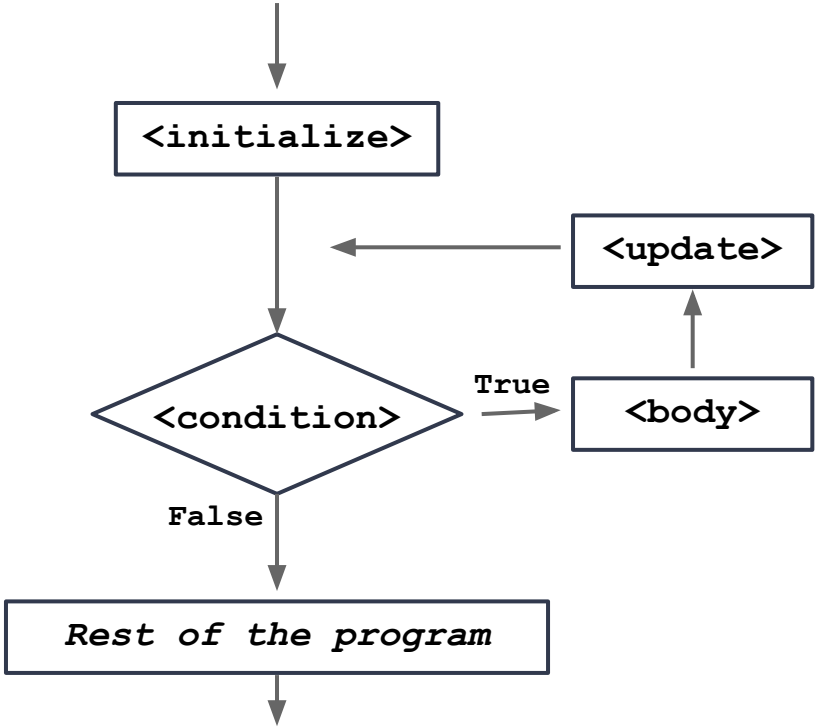


For Loop Review

```
for (<init>; <cond>; <update>) {
    <body>
}
```

statements

boolean expression



For Loop Examples

```
1.  for (let i = 0; i < 10; i = i + 1) {
2.      console.log(i + 1);
3.  }
```

How many iterations in each loop?
What gets printed?

```
1.  let s = 0;
2.  for (let i = 5; i > 0; i = i - 1) {
3.      s = s + i;
4.  }
5.  console.log(s);
```

```
1.  let a = ["Hi", "Hello", "Hey"];
2.  for (let i = 0; i < a.length; i = i + 1) {
3.      console.log(a[i]);
4.  }
```

Ranges in Python

The **`range`** type in Python represents an immutable sequence of numbers

It is commonly used in conjunction with for loops

```
class range(stop)
```

```
class range(start, stop [, step])
```

The arguments must be integers. If the step is omitted, it defaults to 1. If the start is omitted it defaults to 0.

Range Examples

`range(5)` consists of the values 0, 1, 2, 3, 4

`range(3, 7)` consists of the values 3, 4, 5, 6

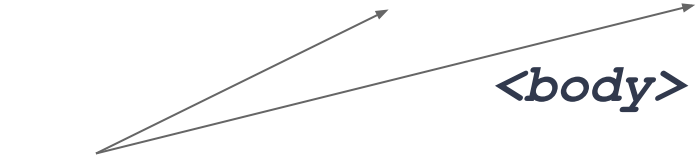
`range(3, 10, 2)` consists of the values 3, 5, 7, 9

`for` Loops in Python

```
for <var> in <sequence>:
    <body>
```

`for` Loops in Python

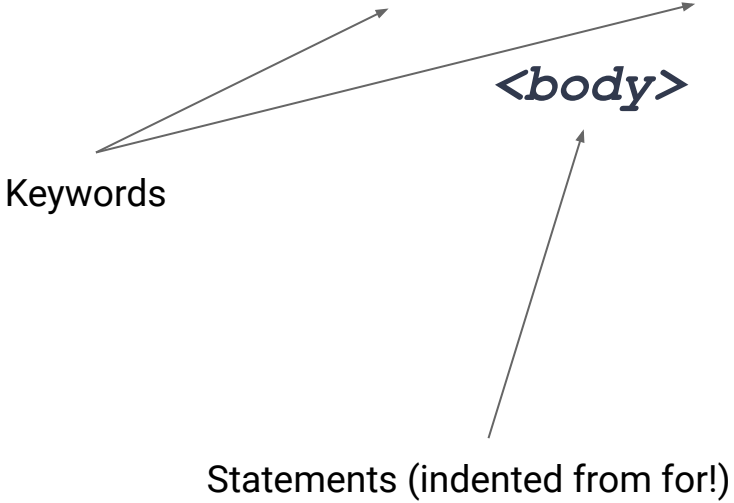
```
for <var> in <sequence>:
    <body>
```



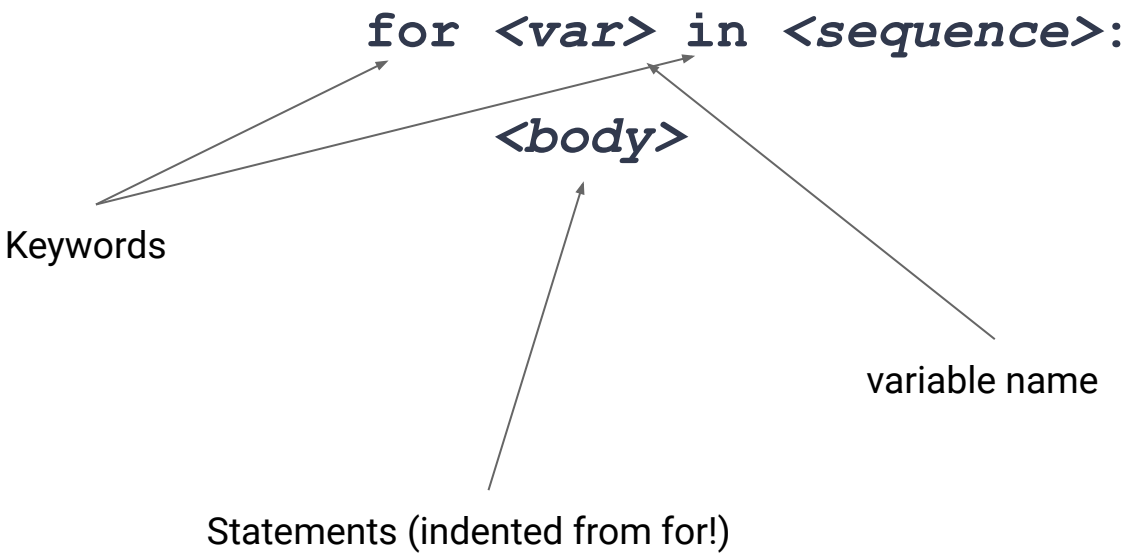
Keywords

`for` Loops in Python

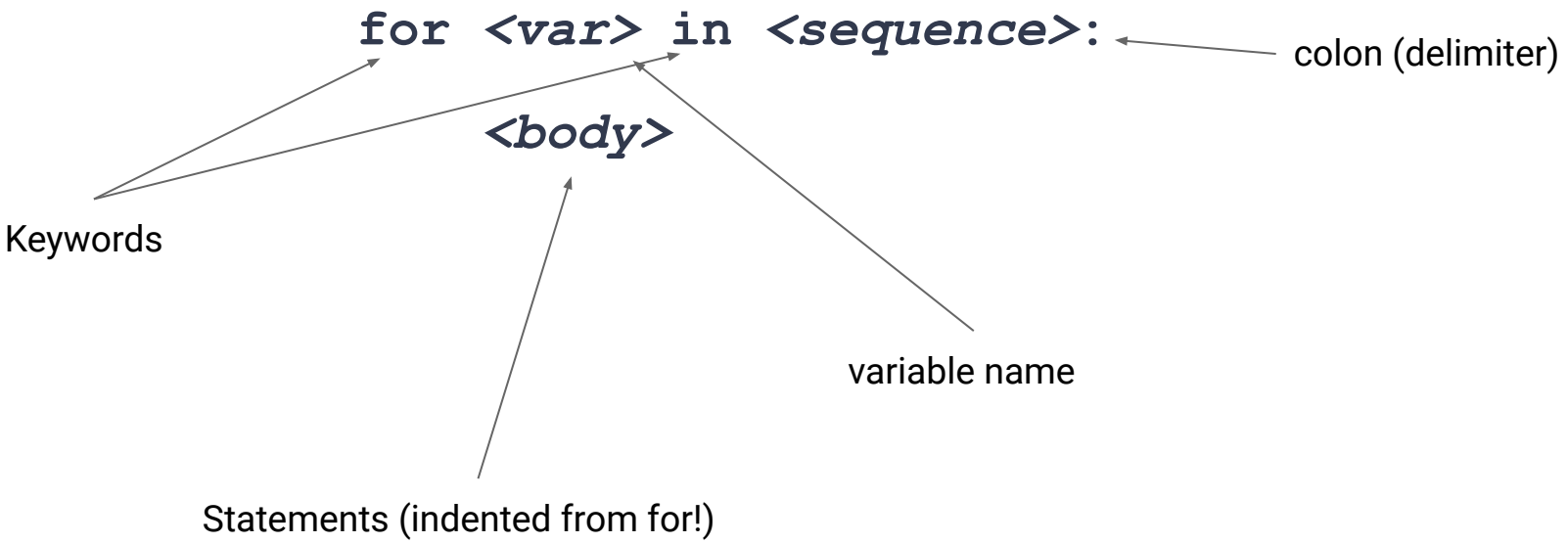
```
for <var> in <sequence>:
    <body>
```



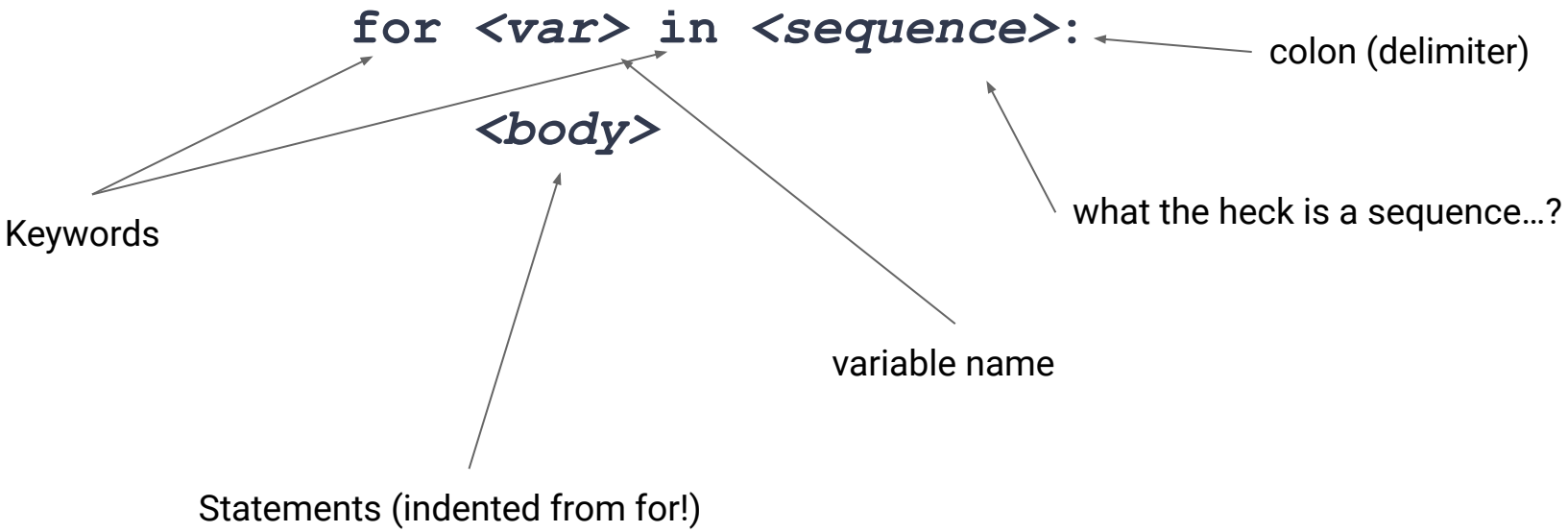
`for` Loops in Python



`for` Loops in Python



`for` Loops in Python



For Comparison

Python

```
sum = 0

for i in range(1, 10):

    sum = sum + i
```

For Comparison

Python

A range is an example of a sequence

```
sum = 0

for i in range(1, 10):

    sum = sum + i
```

For Comparison

Python

```
sum = 0
for i in range(1, 10):
    sum = sum + i
```

JavaScript

```
let sum = 0;
for (let i=1; i<10; i=i+1) {
    sum = sum + i;
}
```

What is a "Sequence" in Python?

```
def printSequence(seq):
    for x in seq:
        print(x)
```

What is a "Sequence" in Python?

```
def printSequence(seq):
    for x in seq:
        print(x)
printSequence(range(1,16,3))
```

A range is an example of a sequence

What is a "Sequence" in Python?

```
def printSequence(seq):
    for x in seq:
        print(x)
printSequence(range(1,16,3))
printSequence("Hello World!")
```

A string is an example of a sequence

What is a "Sequence" in Python?

```
def printSequence(seq):
    for x in seq:
        print(x)
printSequence(range(1,16,3))
printSequence("Hello World!")
printSequence(['a', 'b', 'c'])
```

A list is an example of a sequence

A list in Python works the same as an Array in JavaScript (mostly)

Another for Comparison

Python

```
def printSequence(seq):
    for x in seq:
        print(x)


printSequence(range(1,16,3))

printSequence("Hello World!")

printSequence(['a', 'b', 'c'])
```

JavaScript

Another for Comparison

Python

```
def printSequence(seq):
    for x in seq:
        print(x)


printSequence(range(1,16,3))

printSequence("Hello World!")

printSequence(['a', 'b', 'c'])
```

JavaScript

```
function printSequence(seq) {
  for(let i=0;i<seq.length;i=i+1) {
    console.log(seq[i]);
  }
}
// No range in JavaScript

printSequence("Hello World!");

printSequence(["a", "b", "c"]);
```

`for … of` loops in JavaScript

- JavaScript also has a for loop that works on sequences

```
for (<var> of <sequence>) {
    <body>
}
```

Another for Comparison

Python

```
def printSequence(seq):
    for x in seq:
        print(x)


printSequence(range(1,16,3))

printSequence("Hello World!")

printSequence(['a', 'b', 'c'])
```

JavaScript

```
function printSequence(seq) {
  for(let x of seq) {
    console.log(x);
  }
}
// No range in JavaScript

printSequence("Hello World!");

printSequence(["a", "b", "c"]);
```

Exercises

1. Write a function, **`sumTo`**, that sums all numbers up to (and including) a given number. ie: **`sumTo(3)`** would sum 1, 2, and 3.
2. Write a function, **`sumSquaresTo`**, that sums all perfect squares up to a given number. ie: **`sumSquaresTo(19)`** would sum 1, 4, 9, and 16.
3. Write a function, **`countChars`**, that takes a string and a character, and counts how many times the character appears in that string. ie: **`countChars("Hello World!", "l")`** would return 3.