

Stepping through a 'for' loop

At the beginning - run module

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects

Start with the first line - the 'for' statement



```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects

Look at the list




```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects

Is it empty? No. Start with the first object



```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

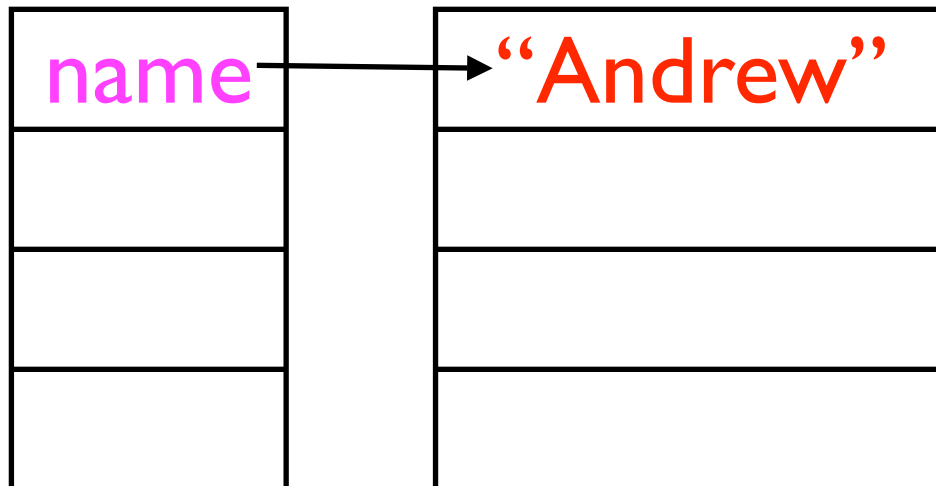
Objects

Assign the first object to the variable 'name'

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```


Variable
Names

Objects



Create the string
object "Andrew" and
assign it to the variable
named 'name'

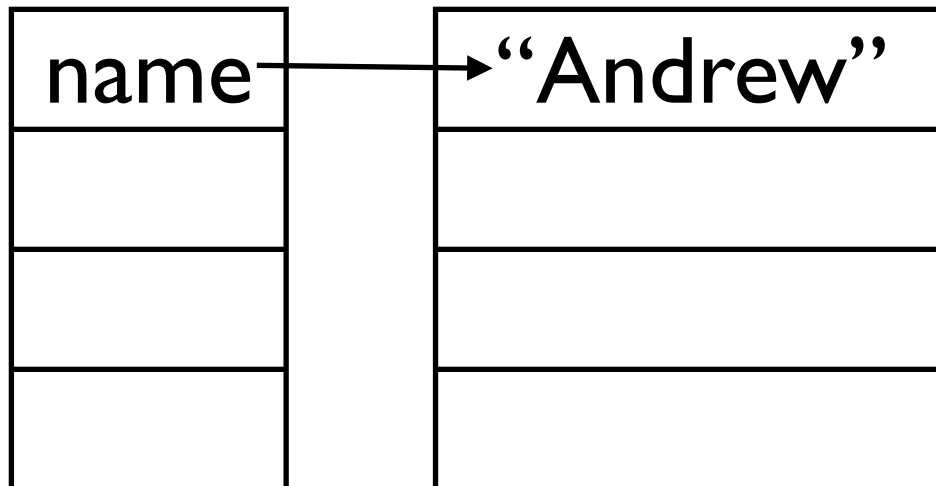
Then start the first line of the code block



```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects

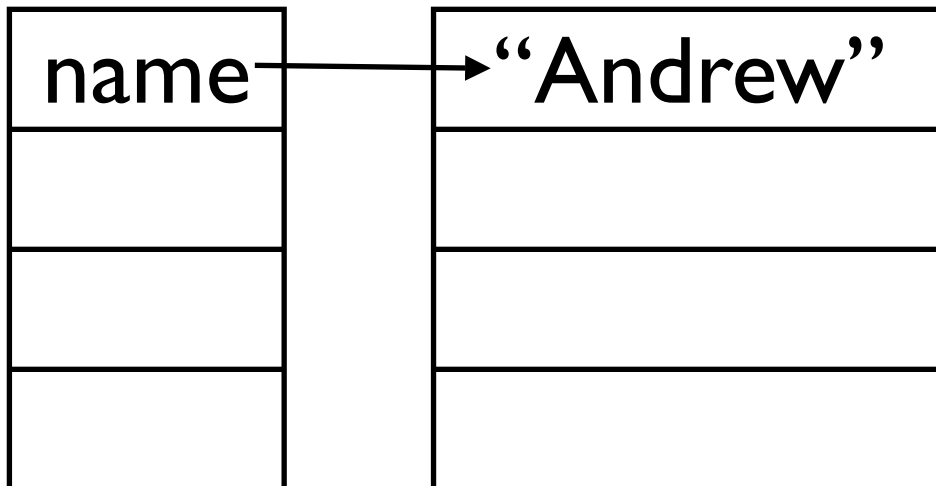


This is the 'print' statement

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects

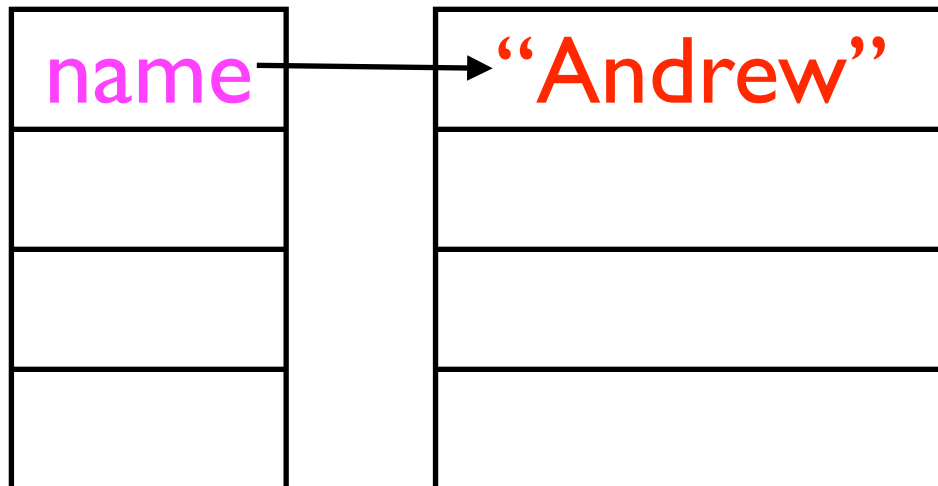


print the string object “Hello,” and the value of the variable with name ‘name’

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```


Variable
Names

Objects




Hello, Andrew

The print statement is finished.
Python is at the end of the code block...

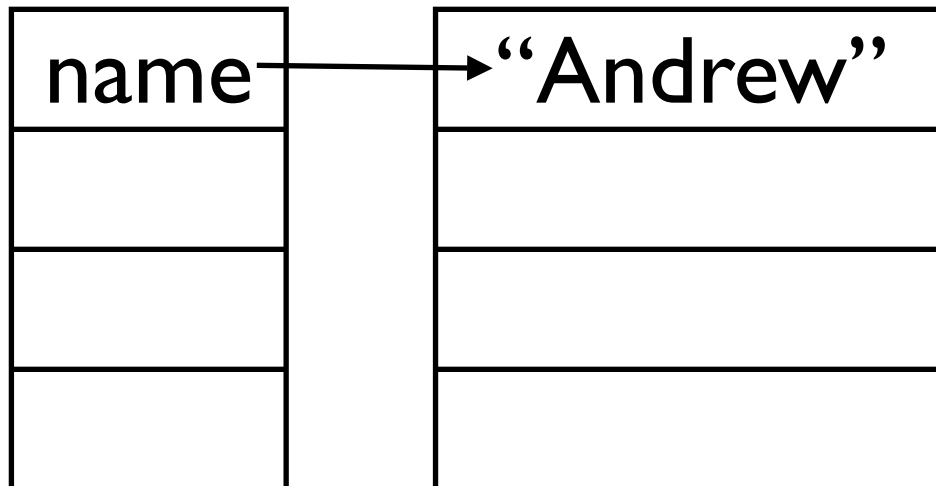


```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```



Variable
Names

Objects

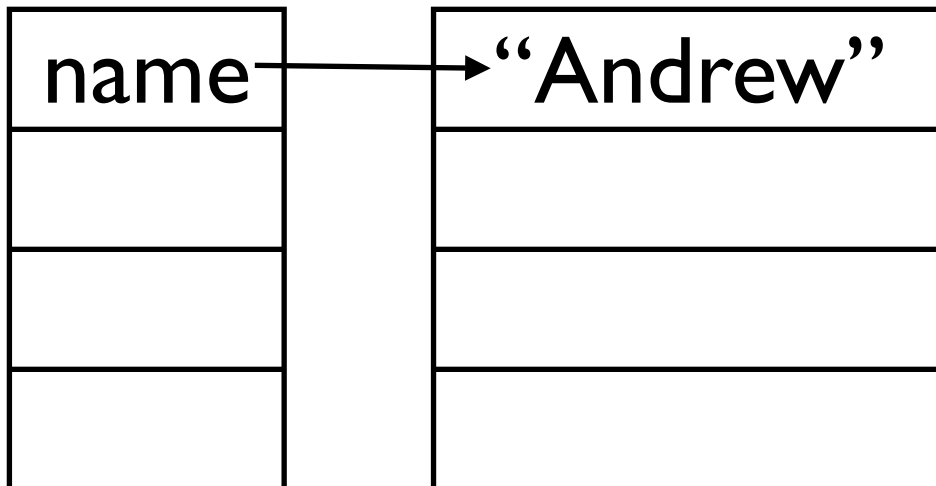


... so go to the 'for' statement again

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects

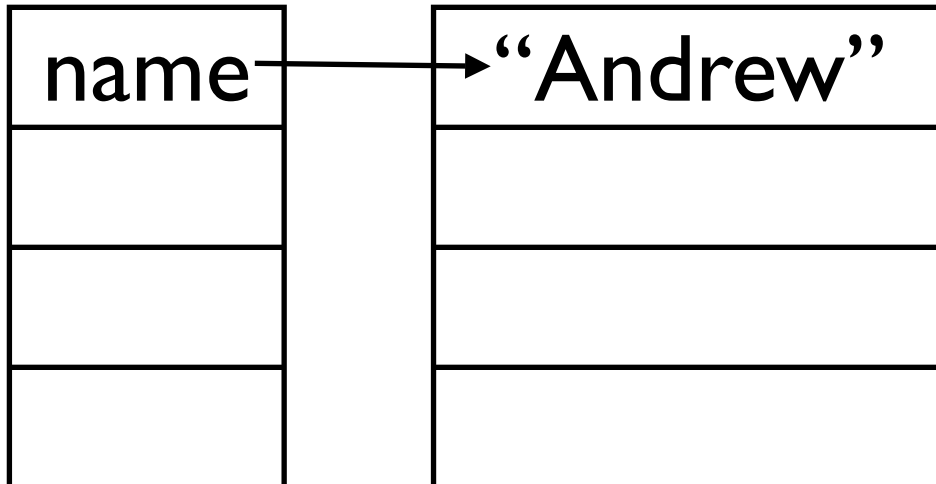


Move the list pointer forward by one

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects

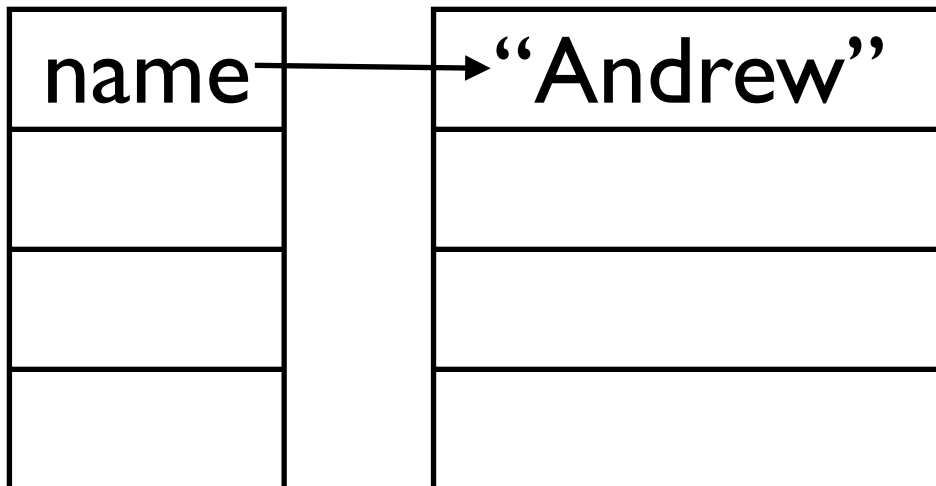


Is it past the end? No, it's on the second item.

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects



Assign the second object to the variable 'name'

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names


name

Objects

"Andrew"
"Tsanwani"

Create the string
object "Tsanwani" and
assign it to the variable
named 'name'

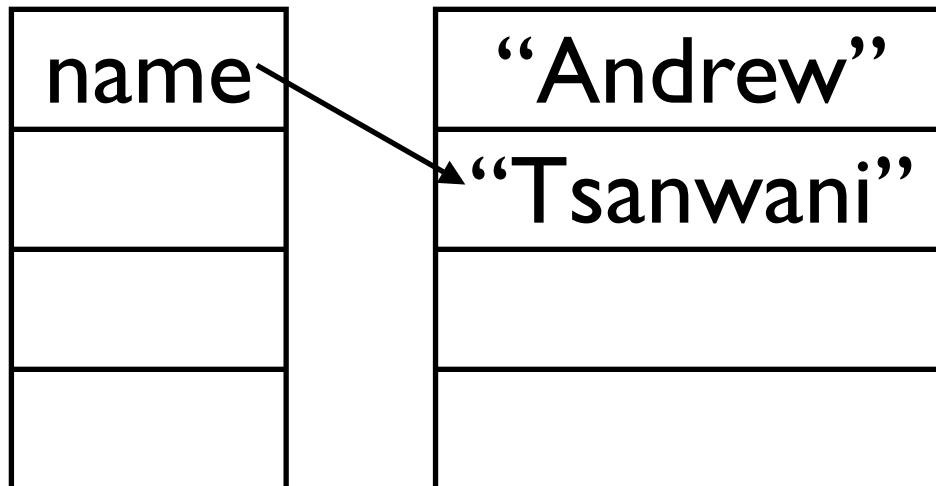
Then start the first line of the code block



```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects



print the string object “Hello,” and the value of the variable with name ‘name’

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects

name

“Andrew”
“Tsanwani”

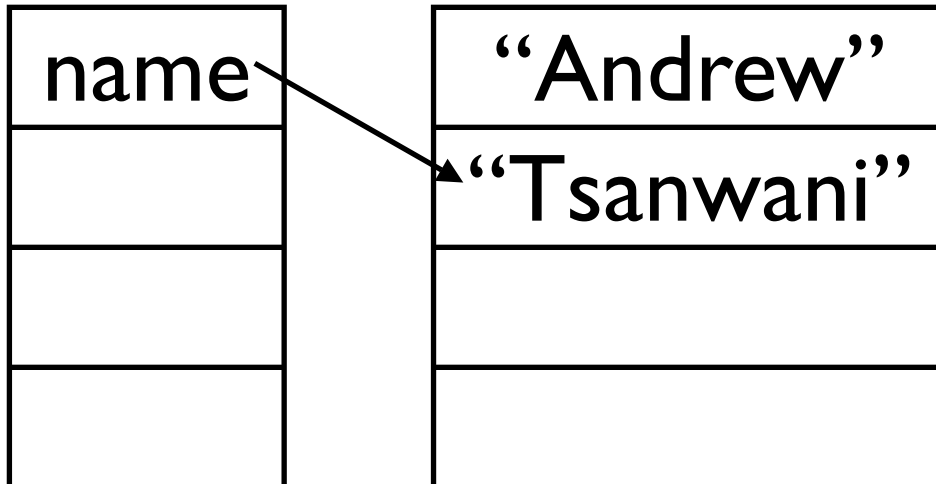
Hello, Andrew
Hello, Tsanwani

The print statement is finished.
Python is at the end of the code block...

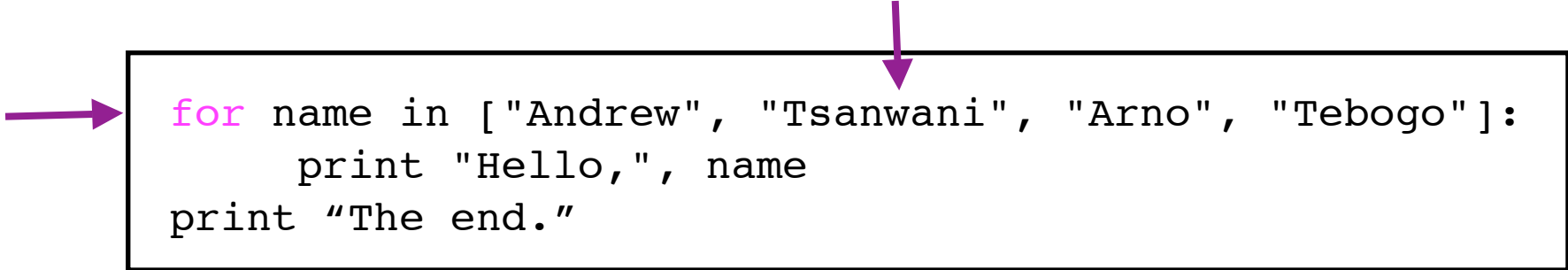
```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects



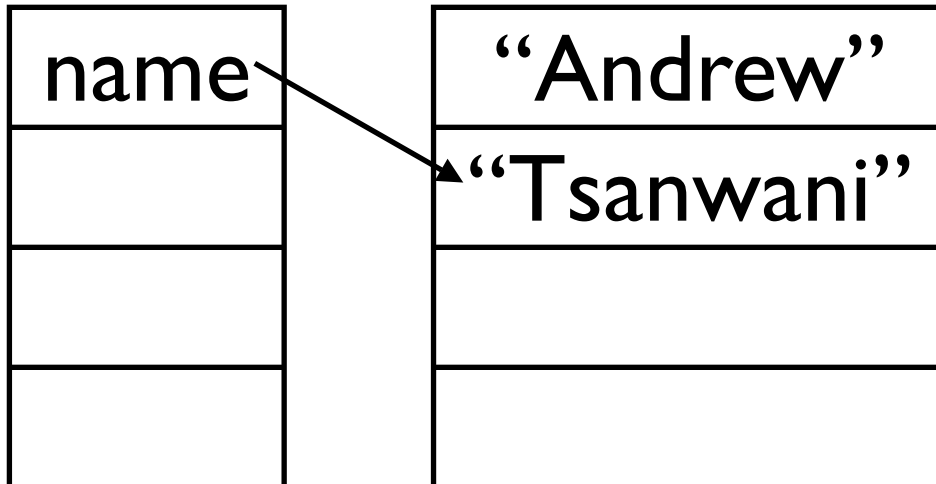
... so go to the 'for' statement again



```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects

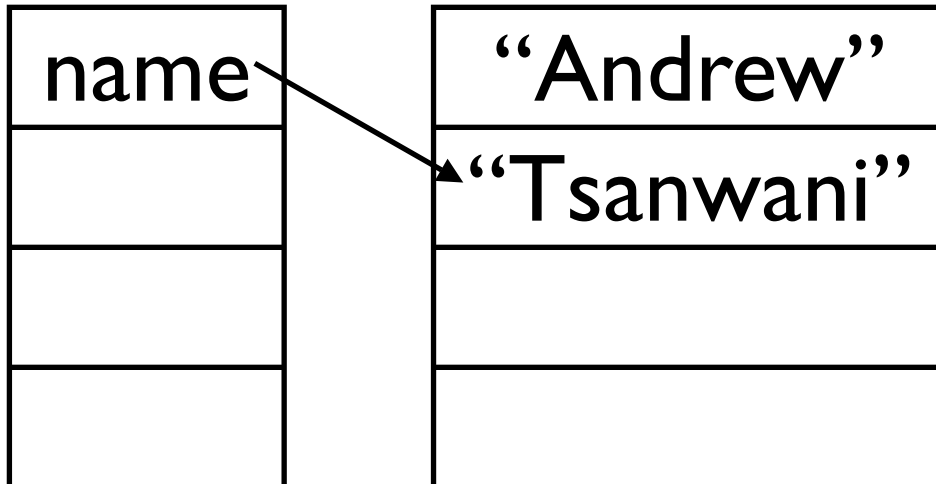


Move the list pointer forward by one

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects

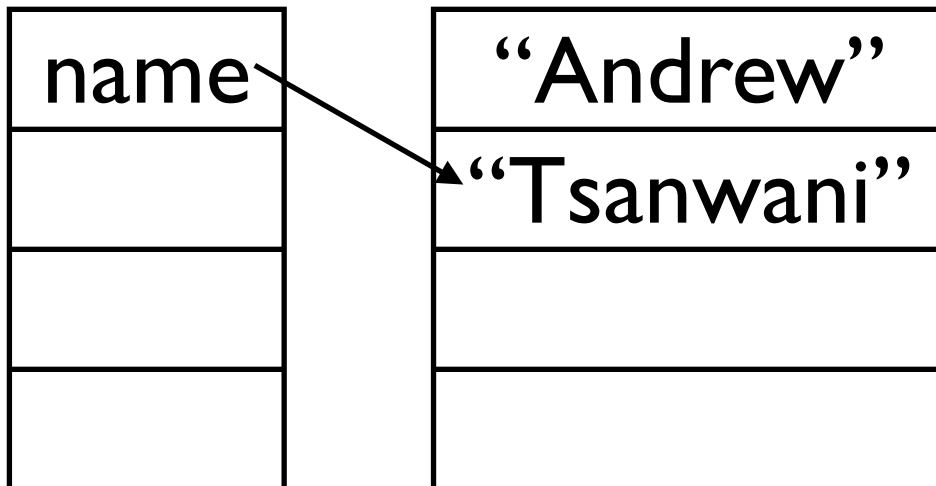


Is it past the end? No, it's on the third item.

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects



Assign the third object to the variable 'name'

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

name

Objects

"Andrew"
"Tsanwani"
"Arno"

Create the string
object "Arno" and
assign it to the variable
named 'name'

Then start the first line of the code block

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

name

Objects

"Andrew"
"Tsanwani"
"Arno"

print the string object **“Hello,”** and the value of the variable with name **‘name’**

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

Objects

name

“Andrew”
“Tsanwani”
“Arno”

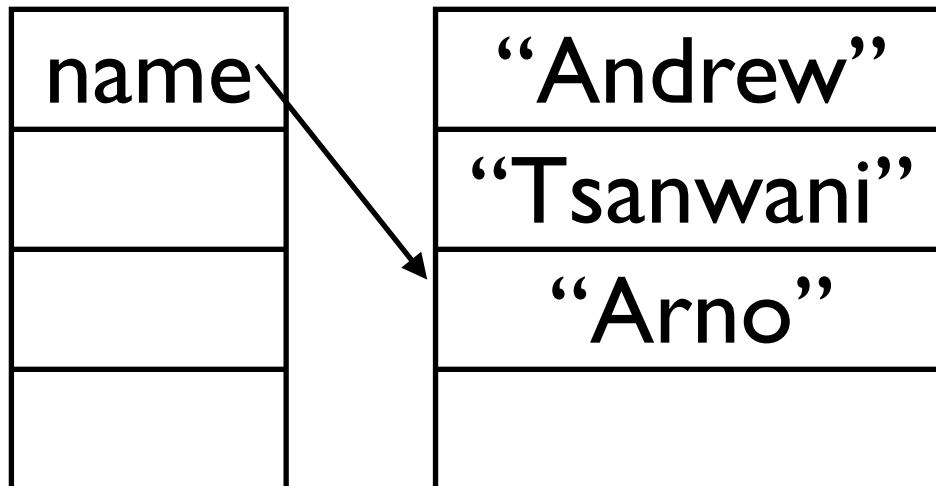
Hello, Andrew
Hello, Tsanwani
Hello, Arno

The print statement is finished.
Python is at the end of the code block...


```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names


Objects



... so go to the 'for' statement again

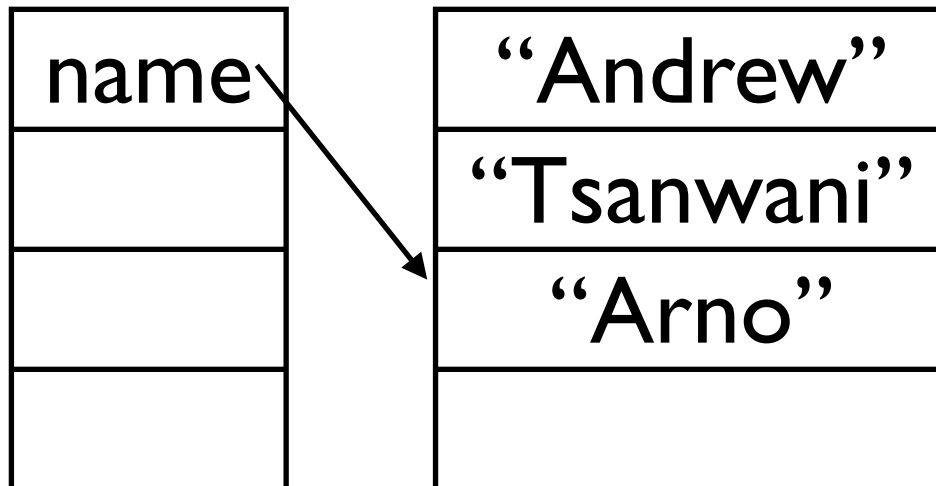


```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```



Variable
Names

Objects

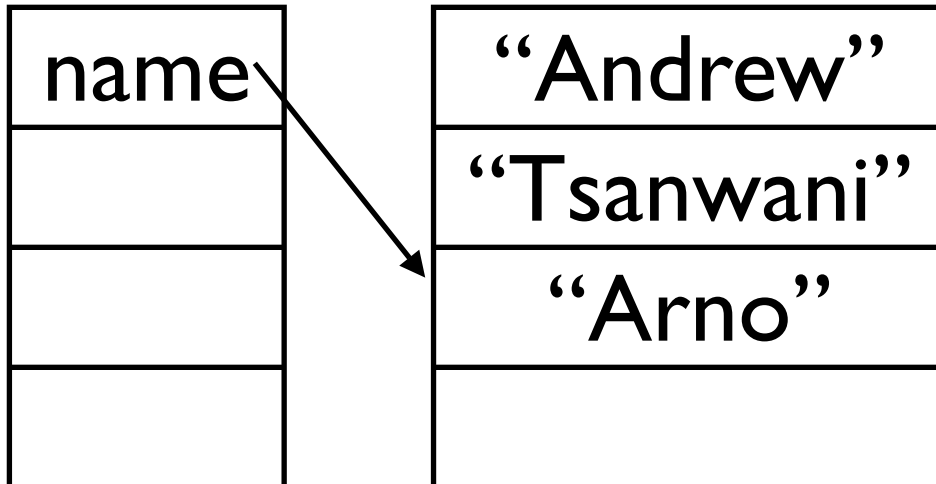


Move the list pointer forward by one


```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names


Objects



Is it past the end? No, it's on the fourth item.

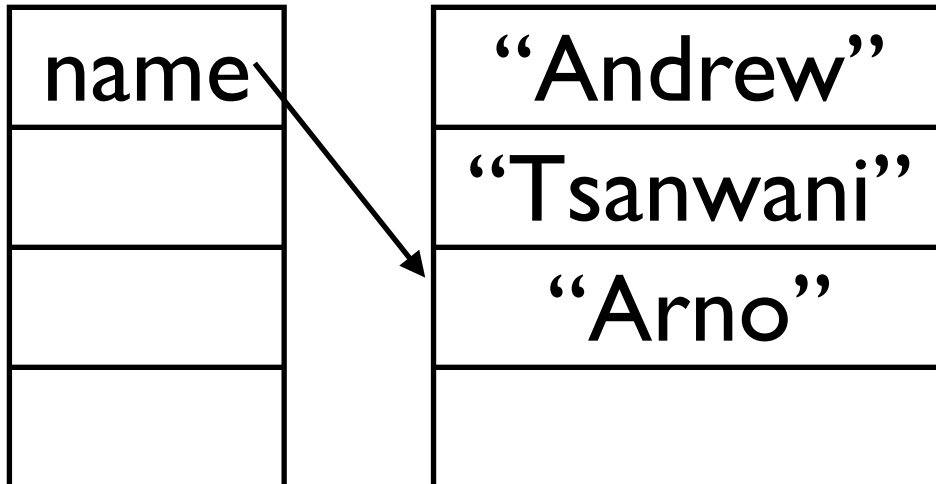


```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```



Variable
Names

Objects



Assign the fourth object to the variable 'name'

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names


name

Objects


"Andrew"
"Tsanwani"
"Arno"
"Tebogo"

Create the string
object "Tebogo" and
assign it to the variable
named 'name'

Then start the first line of the code block



```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

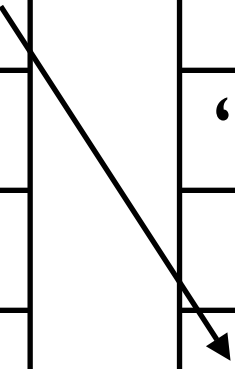


Variable
Names

name

Objects

"Andrew"
"Tsanwani"
"Arno"
"Tebogo"



print the string object “Hello,” and the value of the variable with name ‘name’

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

name

Objects

“Andrew”
“Tsanwani”
“Arno”
“Tebogo”

Hello, Andrew
Hello, Tsanwani
Hello, Arno
Hello, Tebogo

The print statement is finished.
Python is at the end of the code block...

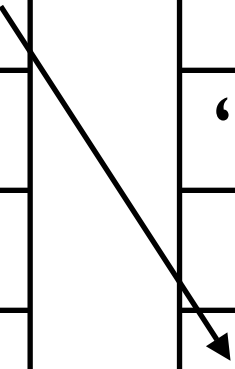
```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names


name

Objects


"Andrew"
"Tsanwani"
"Arno"
"Tebogo"



... so go to the 'for' statement again



```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

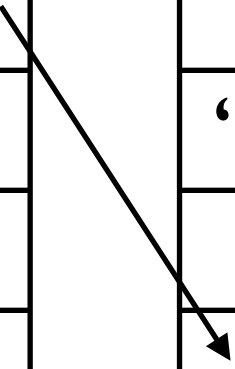


Variable
Names

name

Objects

"Andrew"
"Tsanwani"
"Arno"
"Tebogo"



Move the list pointer forward by one

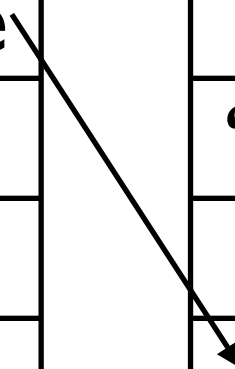
```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

name

Objects

"Andrew"
"Tsanwani"
"Arno"
"Tebogo"



Is it past the end? Yes!

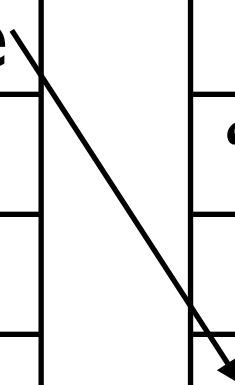
```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

name

Objects

"Andrew"
"Tsanwani"
"Arno"
"Tebogo"



So skip past the code block to the next statement.

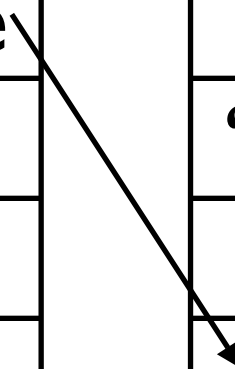
```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
    print "The end."
```

Variable
Names

name

Objects

"Andrew"
"Tsanwani"
"Arno"
"Tebogo"



This is another print statement

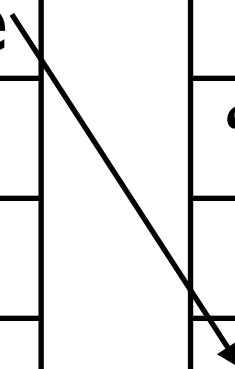
```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

Variable
Names

name

Objects

"Andrew"
"Tsanwani"
"Arno"
"Tebogo"



It prints the string “The end.”

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
    print "The end."
```

Variable
Names

name

Objects

“Andrew”
“Tsanwani”
“Arno”
“Tebogo”

Hello, Andrew
Hello, Tsanwani
Hello, Arno
Hello, Tebogo
The end.

Python looks for the next statement...

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

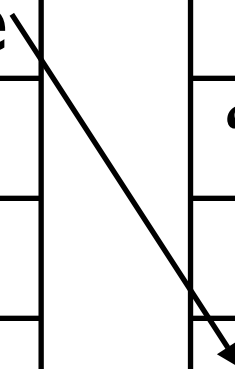


Variable
Names

name

Objects

"Andrew"
"Tsanwani"
"Arno"
"Tebogo"



... but there isn't any, so Python stops.

```
for name in ["Andrew", "Tsanwani", "Arno", "Tebogo"]:  
    print "Hello,", name  
print "The end."
```

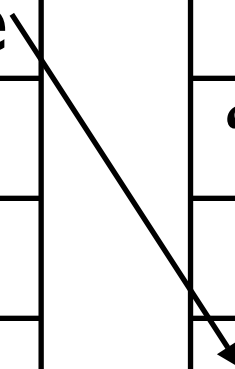


Variable
Names

name

Objects

"Andrew"
"Tsanwani"
"Arno"
"Tebogo"



Final Program Output

```
Hello, Andrew  
Hello, Tsanwani  
Hello, Arno  
Hello, Tebogo  
The end.
```