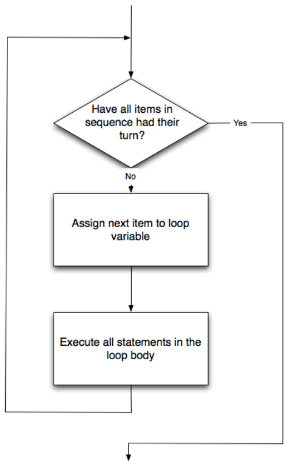
7.3. Flow of Execution of the for Loop

As a program executes, the interpreter always keeps track of which statement is about to be executed. We call this the **control flow**, or the **flow of execution** of the program. When humans execute programs, they often use their finger to point to each statement in turn. So you could think of control flow as “Python’s moving finger”.

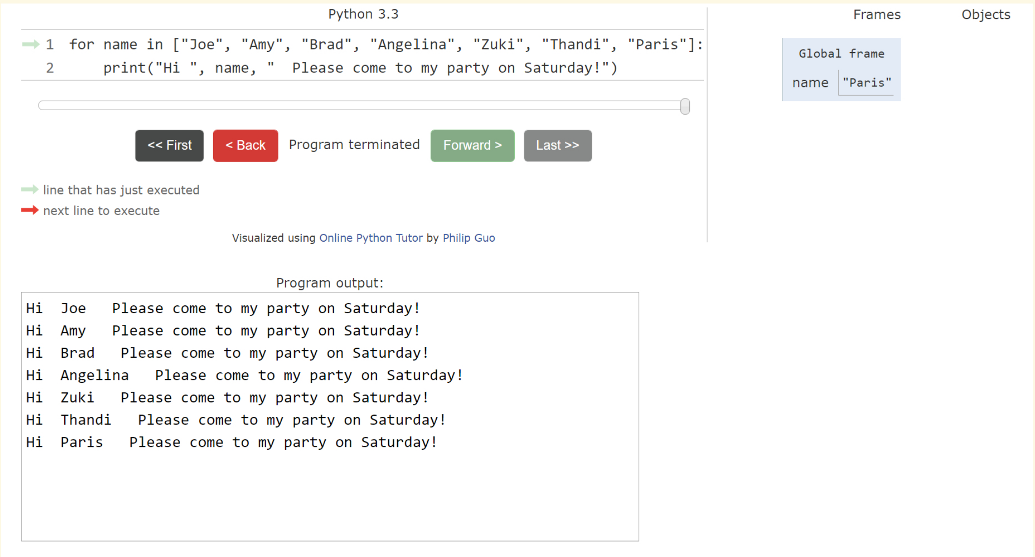
Control flow until now has been strictly top to bottom, one statement at a time. We call this type of control **sequential**. Sequential flow of control is always assumed to be the default behavior for a computer program. The `for` statement changes this.

Flow of control is often easy to visualize and understand if we draw a flowchart. This flowchart shows the exact steps and logic of how the `for` statement executes.


Have all items in sequence had their turn?
Yes
No
Assign next item to loop variable
Execute all statements in the loop body

> **Note**
>
> Not sure what a flowchart is? Check out this funny take on it, in XKCD. And this one.

A codelens demonstration is a good way to help you visualize exactly how the flow of control works with the for loop. Try stepping forward and backward through the program by pressing the buttons. You can see the value of `name` change as the loop iterates through the list of friends.

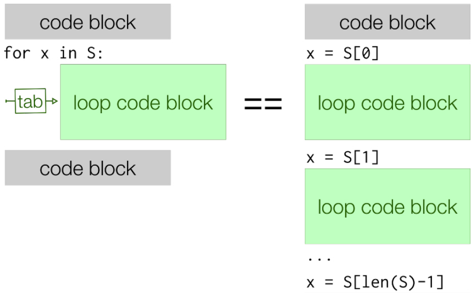


```
for name in ["Joe", "Amy", "Brad", "Angelina", "Zuki", "Thandi", "Paris"]:
    print("Hi ", name, "  Please come to my party on Saturday!")
```

```
Hi  Joe   Please come to my party on Saturday!
Hi  Amy   Please come to my party on Saturday!
Hi  Brad   Please come to my party on Saturday!
Hi  Angelina   Please come to my party on Saturday!
Hi  Zuki   Please come to my party on Saturday!
Hi  Thandi   Please come to my party on Saturday!
Hi  Paris   Please come to my party on Saturday!
```

Activity: 1 -- CodeLens: (vtest)

While loops may not seem to be necessary when you’re iterating over a few items, it is extremely helpful when iterating over lots of items. Imagine if you needed to change what happened in the code block. On the left, when you use iteration, this is easy. On the right, when you have hard coded the process, this is more difficult.



loop code block

code block

You have attempted 2 of 1 activities on this page

✔ Completed. Well Done!

7.2. The for Loop">

e for Loop">

7.4. Strings and for loops">

| Back to top