

EXAMPLE: drawing the stack

Consider the following code:

```
1     def main():
2         a = 10
3         b = 35
4         result = absval(a,b)
5         print "abs(%d-%d) = %d" % (a,b,result)
6
7     def absval(x,y):
8         if x > y:
9             aval = x-y
10        else:
11            aval = y-x
12
13        # draw stack to this point!
14        # as it would look just before return...
15
16        return aval
17
18    main()
```

Follow these steps to draw the stack:

- draw empty stack
- put first function called on stack (usually **main()**)
 - allocate parameters for this frame (assign to arguments)
 - set up any local variables (you can usually ignore loop variables)
 - execute function, step by step
 - send return value (if any) back to calling function
 - remove finished function from stack (erase or cross out)
- continue executing program/function that is now on top of stack

Below are the steps for the above example, with line numbers, so you know what each line is doing. Try it and see if you get the same stack drawing.

```

Line #    do this
-----    -----
14        put a frame for main() on the stack
2         put variable a in main frame, draw arrow to value 10
3         put variable b in main frame, draw arrow to value 35
4         put frame for absval() on the stack
6         put parameter/variable x in absval frame, draw arrow to value 10
6         put parameter/variable y in absval frame, draw arrow to value 35
10        put local variable aval in absval frame, draw arrow to value 25
11        stop at this point - stack drawing should look like below

```

