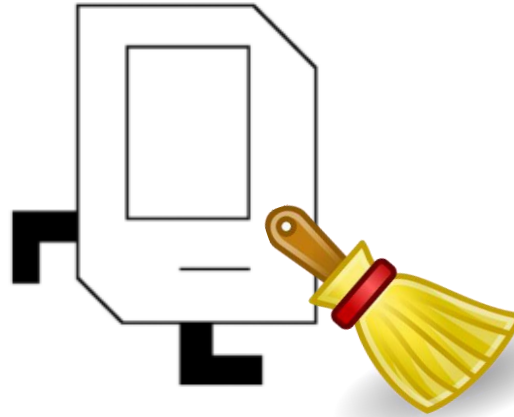
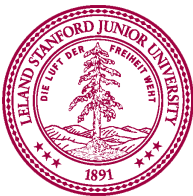


Housekeeping



- Assignment #1 due today
 - How long did you spend on it?
- Assignment #2 goes out today
 - It is due at 12:15pm on Friday, April 15th (one week)
- Invitation to participate in a study of learning introductory computing
 - Watch “Diary Study invitation” video on Canvas



Review

Statement Block

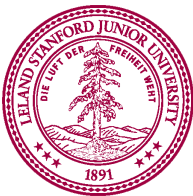
- A *block* (also called a *suite*) in Python is defined by a set of lines of code indented by (at least) the same amount
 - A block ends when the next line is indented less than the line before it
- Example:

```
main():
```

```
    move_to_wall()  
    for i in range(3):  
        turn_left()  
    move()
```

```
def move_to_wall():
```

```
    while front_is_clear():  
        move()
```



Statement Block

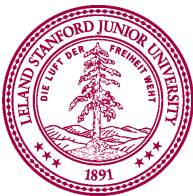
- A *block* (also called a *suite*) in Python is defined by a set of lines of code indented by (at least) the same amount
 - A block ends when the next line is indented less than the line before it
- Example:

```
main() :
```

```
    move_to_wall()  
    for i in range(3):  
        turn_left()  
    move()
```

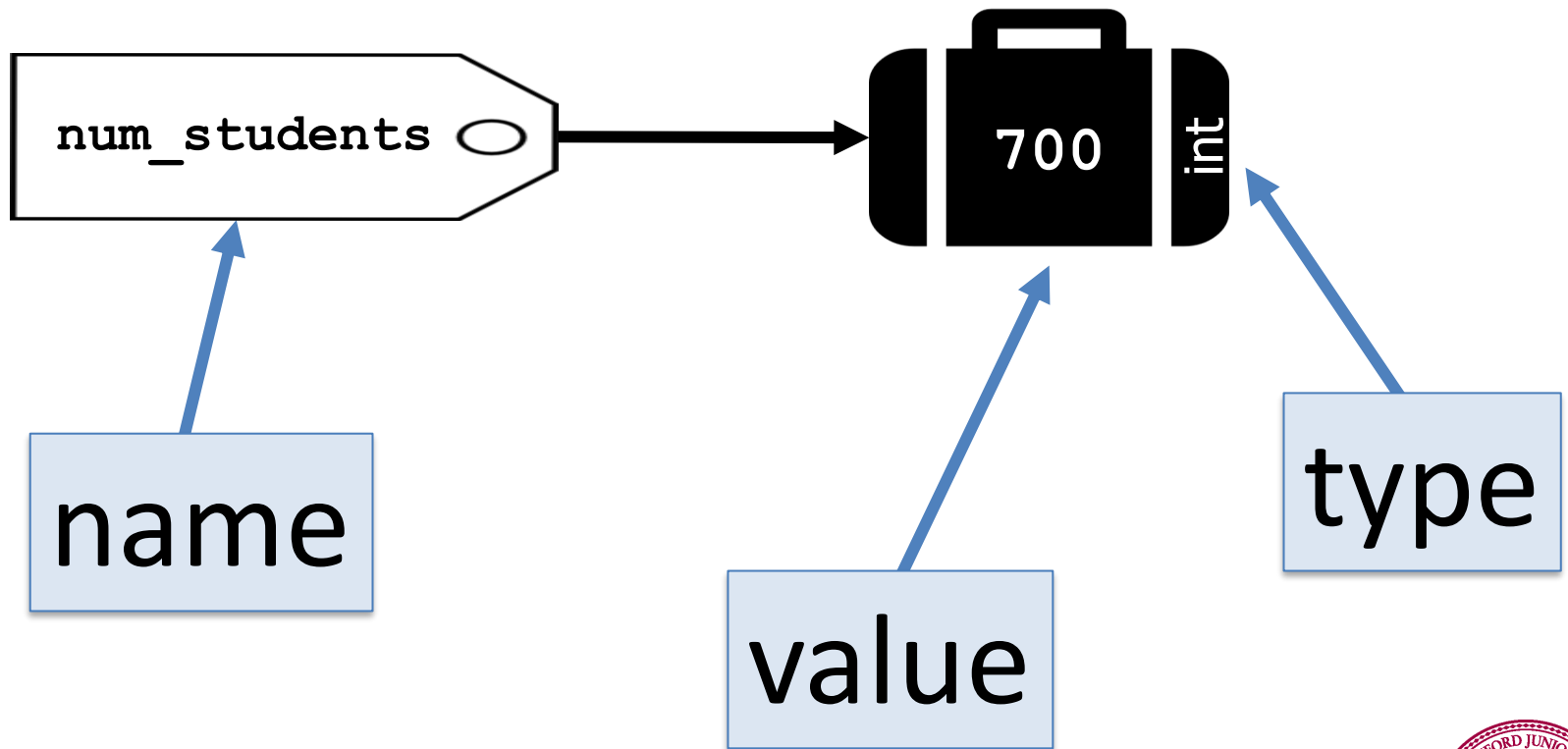
```
def move_to_wall():
```

```
    while front_is_clear():  
        move()
```

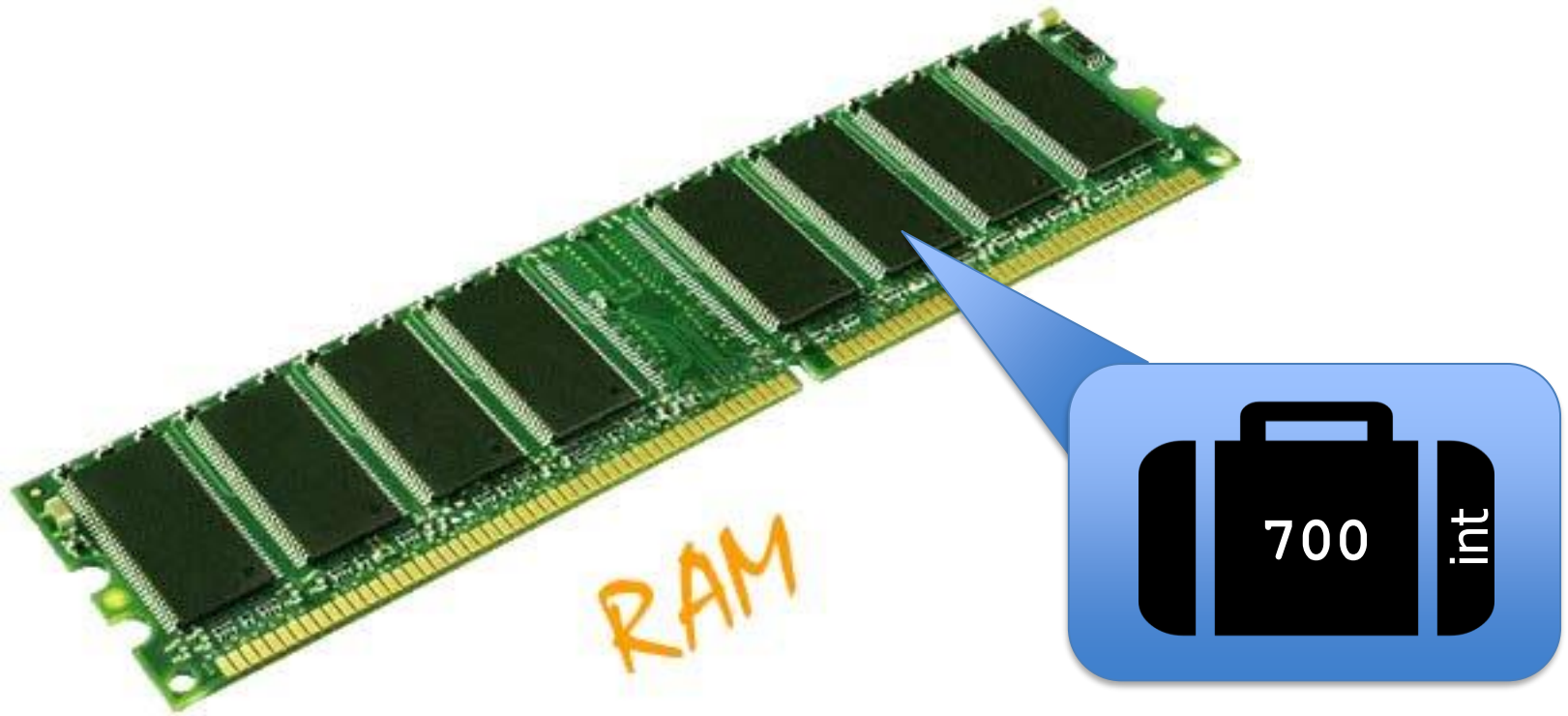


Suitcase Analogy

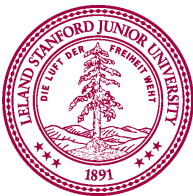
```
num_students = 700
```



Teeny Tiny Suitcases



Your computer has space for millions of suitcases!

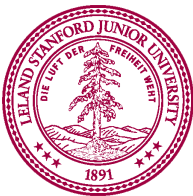


Create, Modify, Use

```
# Create a variable, of type int  
# called age with the value 31.  
age = 31
```

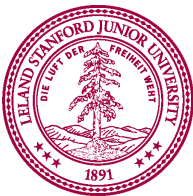
```
# Use the value in age (output it)  
print("age is: " + str(age))
```

```
# Modify age to be one greater.  
age = age + 1
```



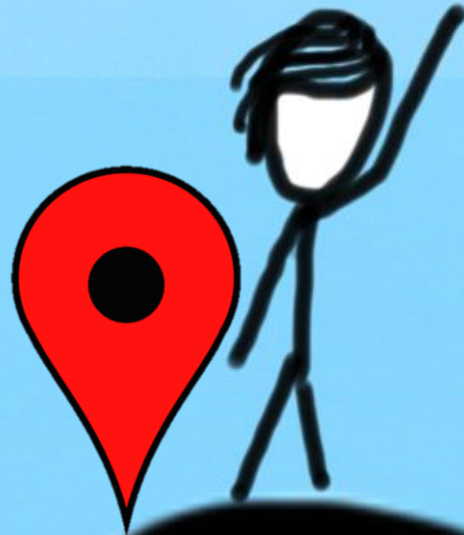
Recall, Arithmetic Operators

+	Addition
−	Subtraction
*	Multiplication
/	Division

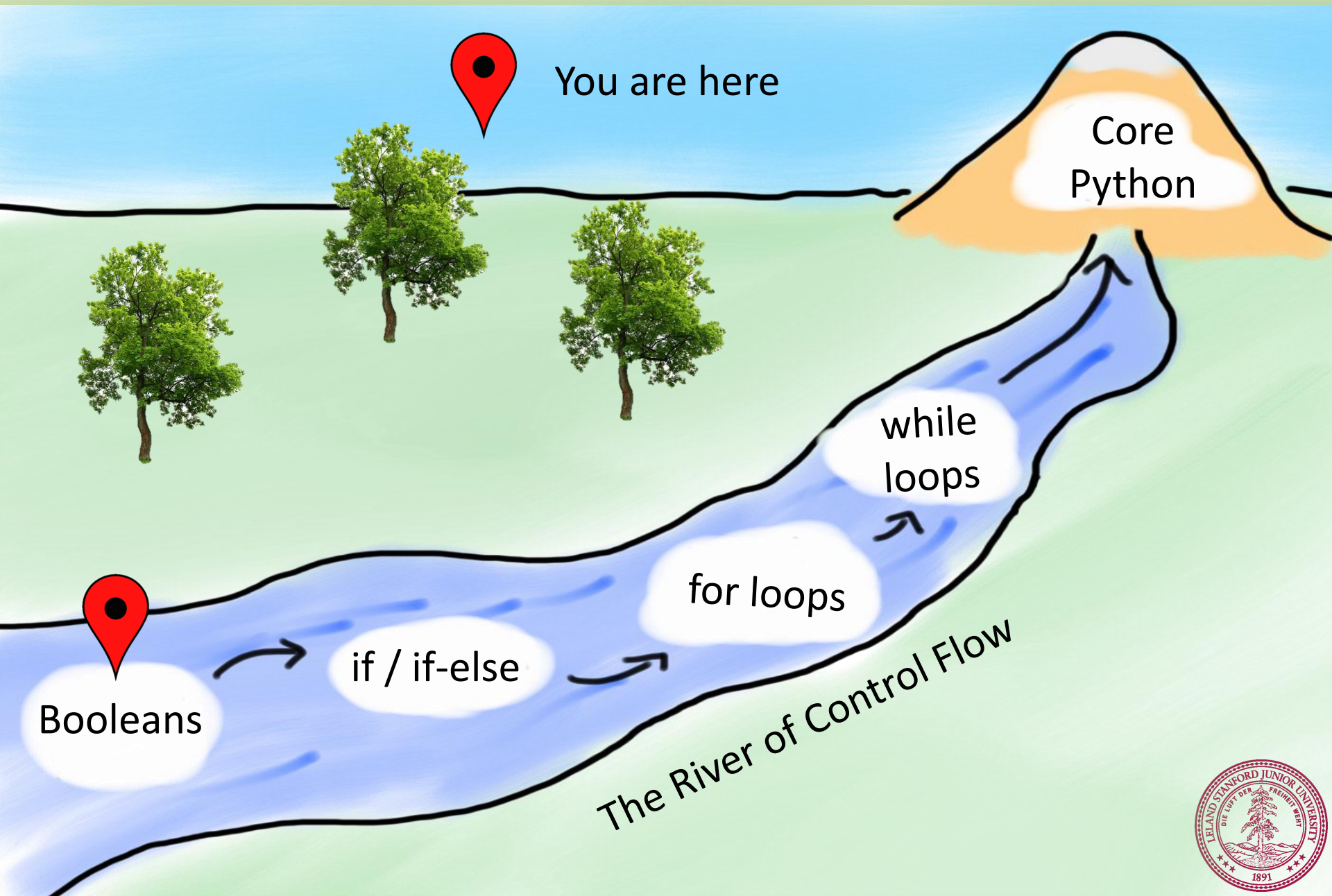


Today's Goal

1. Learn about Boolean variables
2. Use For / While / If statements in Python



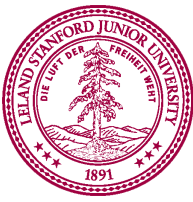
Today's Route



George Boole

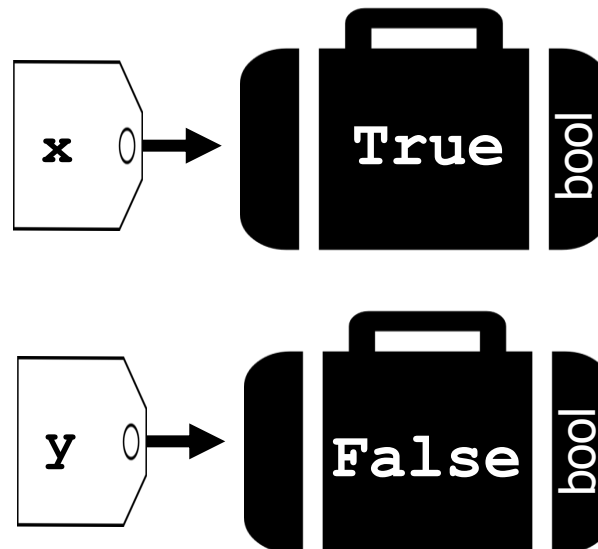


English Mathematician teaching in Ireland 1815 – 1864
Boole died of being too cool (literally)!



Boolean Variables

- Boolean variables only have values **True** or **False**
 x = True
 y = False
- Represent logical values
- Type is called **bool** in python

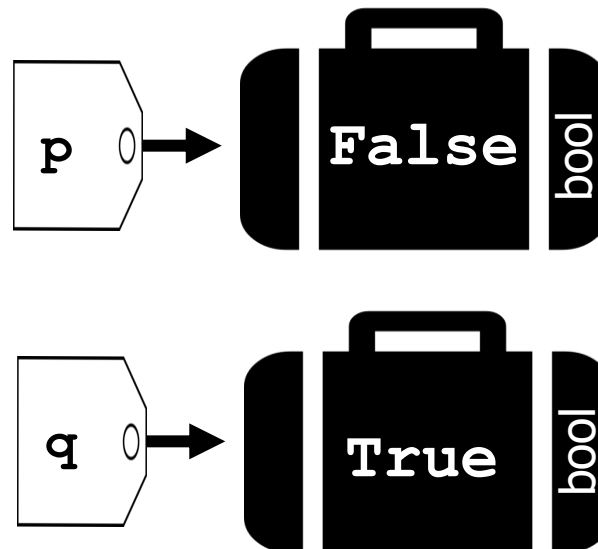


Boolean Variables

- Can also set value of Boolean variable by comparison
 - This is called a Boolean expression

p = 5.0 < 4.0

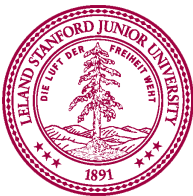
q = 2 > 1



Comparison Operators

Operator	Meaning	Example	Value
==	equals	$(1 + 1) == 2$	True
!=	not equals	$3.2 != 2.5$	True
<	less than	$10 < 5$	False
>	greater than	$10 > 5$	True
<=	less than or equal to	$126 <= 100$	False
>=	greater than or equal to	$5.0 >= 5.0$	True

All have comparisons have equal precedence

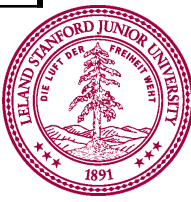


Logical Operators

In order of precedence (assuming **p** and **q** are Boolean values):

Operator	Meaning
not	if p is True then not p is False , and vice versa
and	p and q is only True if p and q are <u>both</u> True . So, it is False when either p or q are False .
or	p or q is True if <u>either</u> p or q (or both) are True . So, it is only False when both p and q are False

Operator	Example	Result
not	not (2 > 3)	True
and	(2 == 3) and (-1 < 5)	False
or	(4 == 5) or (7 >= 3)	True



Logical Operators

- What if we want to determine that a variable **x** does not have a value of either 1 or 2?

- Might write something like how it sounds in English:

p = (x != 1) or (x != 2)

– That's a bug! **p** will always be **True**

- Really, want to say:

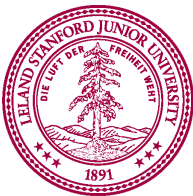
p = (x != 1) and (x != 2)

- Bonus: you can chain tests for a range, like in algebra!

p = (2 <= x <= 10)

– Same thing as:

p = (2 <= x) and (x <= 10)



Short Circuit Evaluation

- Python stops evaluating a Boolean expression as soon as it knows the answer

- Consider:

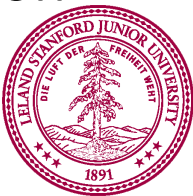
`p = (5 > 3) or (4 <= 2)`

- The test `(4 <= 2)` is not performed!

- Example of useful case:

`p = (x != 0) and ((y % x) == 0)`

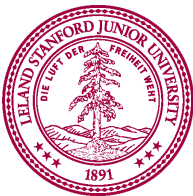
- Avoid division by 0 error, since `((y % x) == 0)` is not performed when `x` is 0
 - To compute remainder (%), Python needs to do division



Please ...

**NO FOOD OR
DRINKS**

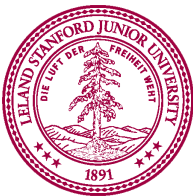
FreeSignaPrinter.com





FreeSignPrinter.com

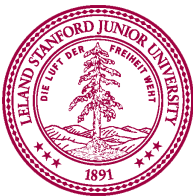
`is_allowed = not food or drinks`



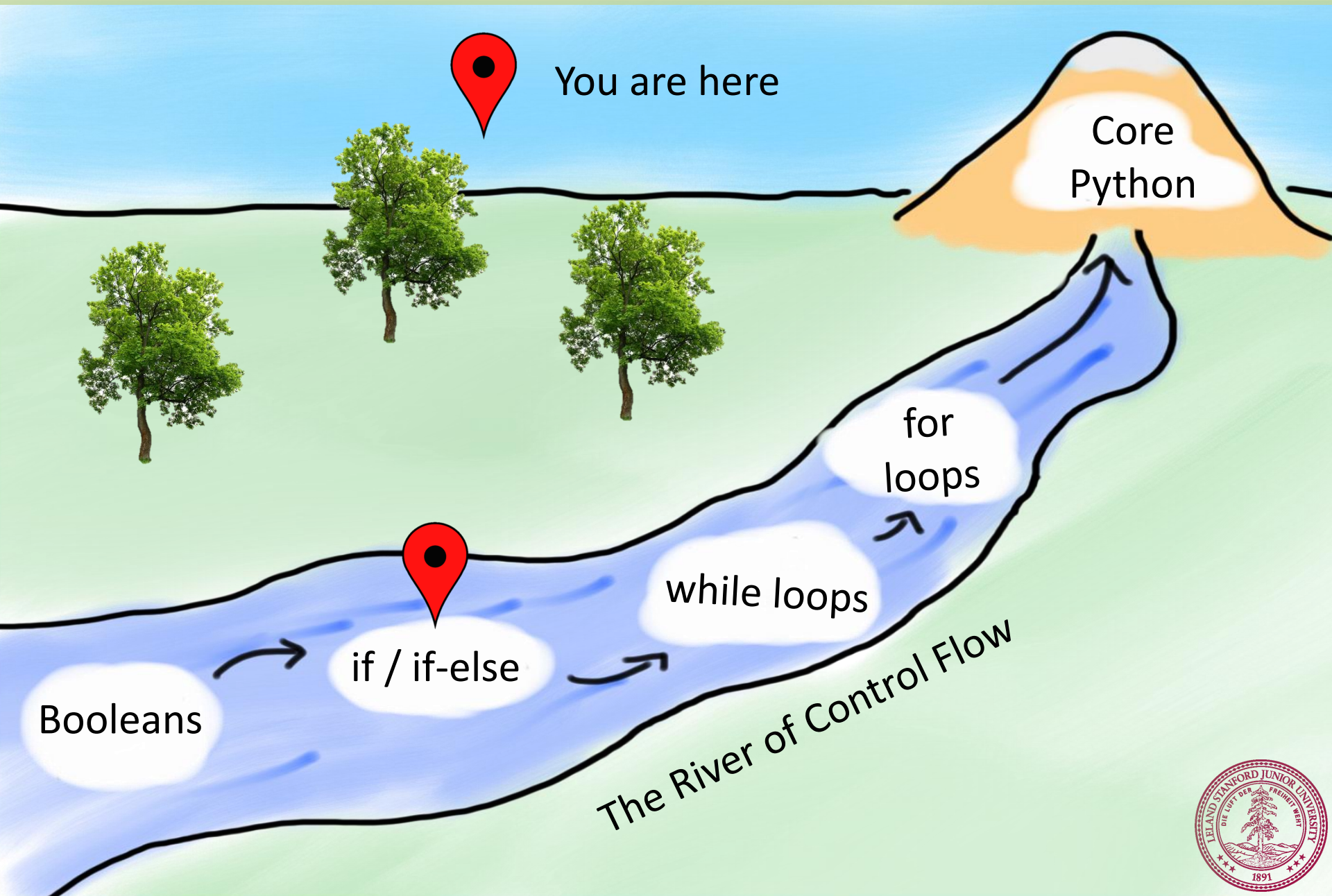


FreeSignPrinter.com

`is_allowed = (not food) or drinks`



Today's Route



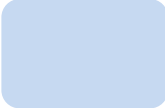
if statements, revisited

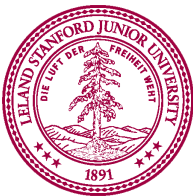
Any Boolean expression
or variable

if condition:

 statements
(block)

note indenting

```
x = int(input("Enter a number: "))  
if x < 10:  
     print(x, "is less than 10")  
print("Beat Cal!")
```



if statements, revisited

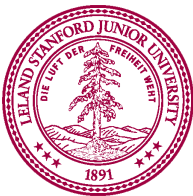
Any Boolean expression
or variable

if *condition*:
 statements
 (block)

note indenting

```
→ x = int(input("Enter a number: "))  
  if x < 10:  
      print(x, "is less than 10")  
  print("Beat Cal!")
```

Enter a number:



if statements, revisited

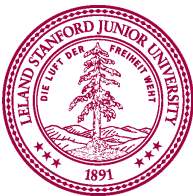
Any Boolean expression
or variable

if *condition*:
 statements
 (block)

note indenting

```
→ x = int(input("Enter a number: "))  
  if x < 10:  
      print(x, "is less than 10")  
  print("Beat Cal!")
```

```
Enter a number: 8
```



if statements, revisited

Any Boolean expression
or variable

if *condition*:
 statements
 (block)

note indenting

```
x = int(input("Enter a number: "))  
→ if x < 10:  
    print(x, "is less than 10")  
print("Beat Cal!")
```

Enter a number: 8



if statements, revisited

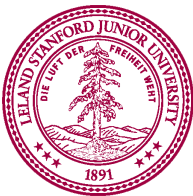
Any Boolean expression
or variable

if *condition*:
 statements
 (block)

note indenting

```
x = int(input("Enter a number: "))  
if x < 10:  
    print(x, "is less than 10")  
print("Beat Cal!")
```

```
Enter a number: 8  
8 is less than 10
```



if statements, revisited

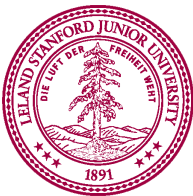
Any Boolean expression
or variable

if *condition*:
 statements
 (block)

note indenting

```
x = int(input("Enter a number: "))  
if x < 10:  
    print(x, "is less than 10")  
→ print("Beat Cal!")
```

```
Enter a number: 8  
8 is less than 10  
Beat Cal!
```



if statements, revisited

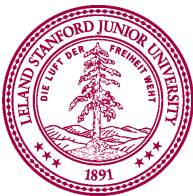
Any Boolean expression
or variable

if *condition*:
 statements
 (block)

note indenting

```
→ x = int(input("Enter a number: "))  
  if x < 10:  
      print(x, "is less than 10")  
  print("Beat Cal!")
```

Enter a number:



if statements, revisited

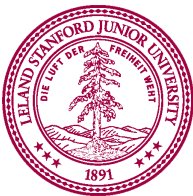
Any Boolean expression
or variable

if *condition*:
 statements
 (block)

note indenting

```
→ x = int(input("Enter a number: "))  
  if x < 10:  
      print(x, "is less than 10")  
  print("Beat Cal!")
```

Enter a number: 15



if statements, revisited

Any Boolean expression
or variable

if *condition*:
 statements
 (block)

note indenting

```
x = int(input("Enter a number: "))
```

```
→ if x < 10:  
    print(x, "is less than 10")  
print("Beat Cal!")
```

```
Enter a number: 15
```



if statements, revisited

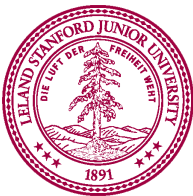
Any Boolean expression
or variable

if *condition*:
 statements
 (block)

note indenting

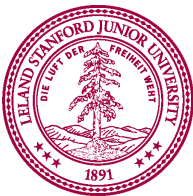
```
x = int(input("Enter a number: "))  
if x < 10:  
    print(x, "is less than 10")  
→ print("Beat Cal!")
```

```
Enter a number: 15  
Beat Cal!
```



Nested `if` statements

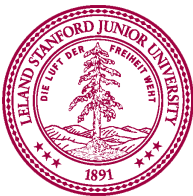
```
x = int(input("Enter a number: "))  
if x > 0:  
    print(x, "is positive")  
    if x > 5:  
        print(x, "is greater than 5")  
    print("I like to stay positive")  
print("Thank you")
```



Nested `if` statements

```
→ x = int(input("Enter a number: "))  
  if x > 0:  
      print(x, "is positive")  
      if x > 5:  
          print(x, "is greater than 5")  
      print("I like to stay positive")  
  print("Thank you")
```

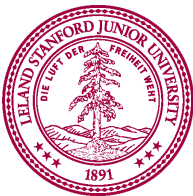
Enter a number:



Nested `if` statements

```
→ x = int(input("Enter a number: "))  
if x > 0:  
    print(x, "is positive")  
    if x > 5:  
        print(x, "is greater than 5")  
    print("I like to stay positive")  
print("Thank you")
```

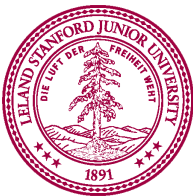
Enter a number: 3



Nested `if` statements

```
x = int(input("Enter a number: "))  
→ if x > 0:  
    print(x, "is positive")  
    if x > 5:  
        print(x, "is greater than 5")  
    print("I like to stay positive")  
print("Thank you")
```

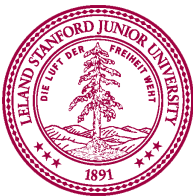
Enter a number: 3



Nested `if` statements

```
x = int(input("Enter a number: "))  
if x > 0:  
    → print(x, "is positive")  
        if x > 5:  
            print(x, "is greater than 5")  
        print("I like to stay positive")  
print("Thank you")
```

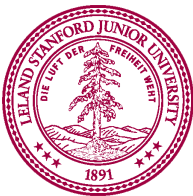
```
Enter a number: 3  
3 is positive
```



Nested `if` statements

```
x = int(input("Enter a number: "))  
if x > 0:  
    print(x, "is positive")  
→    if x > 5:  
        print(x, "is greater than 5")  
    print("I like to stay positive")  
print("Thank you")
```

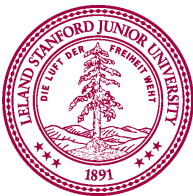
```
Enter a number: 3  
3 is positive
```



Nested `if` statements

```
x = int(input("Enter a number: "))
if x > 0:
    print(x, "is positive")
    if x > 5:
        print(x, "is greater than 5")
→    print("I like to stay positive")
print("Thank you")
```

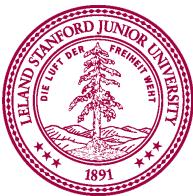
```
Enter a number: 3
3 is positive
I like to stay positive
```



Nested `if` statements

```
x = int(input("Enter a number: "))  
if x > 0:  
    print(x, "is positive")  
    if x > 5:  
        print(x, "is greater than 5")  
    print("I like to stay positive")  
→ print("Thank you")
```

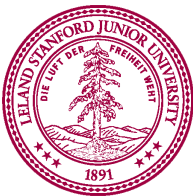
```
Enter a number: 3  
3 is positive  
I like to stay positive  
Thank you
```



Nested `if` statements

```
→ x = int(input("Enter a number: "))  
  if x > 0:  
      print(x, "is positive")  
      if x > 5:  
          print(x, "is greater than 5")  
      print("I like to stay positive")  
  print("Thank you")
```

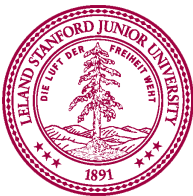
Enter a number:



Nested `if` statements

```
→ x = int(input("Enter a number: "))  
  if x > 0:  
      print(x, "is positive")  
      if x > 5:  
          print(x, "is greater than 5")  
      print("I like to stay positive")  
  print("Thank you")
```

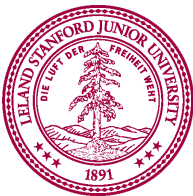
Enter a number: 12



Nested `if` statements

```
x = int(input("Enter a number: "))  
→ if x > 0:  
    print(x, "is positive")  
    if x > 5:  
        print(x, "is greater than 5")  
    print("I like to stay positive")  
print("Thank you")
```

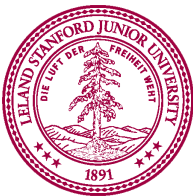
Enter a number: 12



Nested `if` statements

```
x = int(input("Enter a number: "))  
if x > 0:  
    print(x, "is positive")  
    if x > 5:  
        print(x, "is greater than 5")  
    print("I like to stay positive")  
print("Thank you")
```

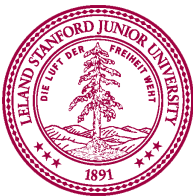
```
Enter a number: 12  
12 is positive
```



Nested `if` statements

```
x = int(input("Enter a number: "))  
if x > 0:  
    print(x, "is positive")  
→    if x > 5:  
        print(x, "is greater than 5")  
    print("I like to stay positive")  
print("Thank you")
```

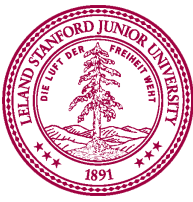
```
Enter a number: 12  
12 is positive
```



Nested `if` statements

```
x = int(input("Enter a number: "))
if x > 0:
    print(x, "is positive")
    if x > 5:
        print(x, "is greater than 5")
    print("I like to stay positive")
print("Thank you")
```

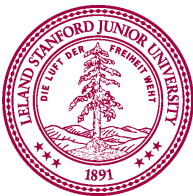
Enter a number: 12
12 is positive
12 is greater than 5



Nested `if` statements

```
x = int(input("Enter a number: "))  
if x > 0:  
    print(x, "is positive")  
    if x > 5:  
        print(x, "is greater than 5")  
→    print("I like to stay positive")  
print("Thank you")
```

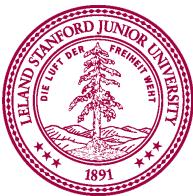
```
Enter a number: 12  
12 is positive  
12 is greater than 5  
I like to stay positive
```



Nested `if` statements

```
x = int(input("Enter a number: "))
if x > 0:
    print(x, "is positive")
    if x > 5:
        print(x, "is greater than 5")
    print("I like to stay positive")
→ print("Thank you")
```

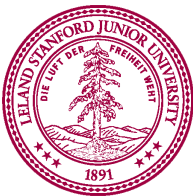
```
Enter a number: 12
12 is positive
12 is greater than 5
I like to stay positive
Thank you
```



Nested `if` statements

```
→ x = int(input("Enter a number: "))  
  if x > 0:  
      print(x, "is positive")  
      if x > 5:  
          print(x, "is greater than 5")  
      print("I like to stay positive")  
  print("Thank you")
```

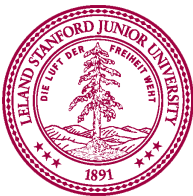
Enter a number:



Nested `if` statements

```
→ x = int(input("Enter a number: "))  
  if x > 0:  
      print(x, "is positive")  
      if x > 5:  
          print(x, "is greater than 5")  
      print("I like to stay positive")  
  print("Thank you")
```

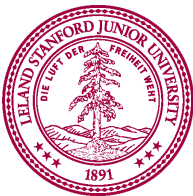
Enter a number: -2



Nested `if` statements

```
x = int(input("Enter a number: "))  
→ if x > 0:  
    print(x, "is positive")  
    if x > 5:  
        print(x, "is greater than 5")  
    print("I like to stay positive")  
print("Thank you")
```

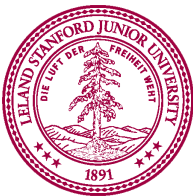
Enter a number: -2



Nested `if` statements

```
x = int(input("Enter a number: "))
if x > 0:
    print(x, "is positive")
    if x > 5:
        print(x, "is greater than 5")
    print("I like to stay positive")
→ print("Thank you")
```

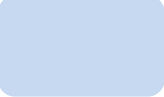
```
Enter a number: -2
Thank you
```



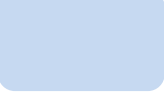
if-else statement

Any Boolean expression
or variable

if condition:

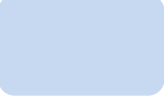
 statements (block) # condition True

else:

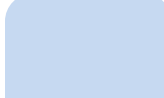
 statements (block) # condition False

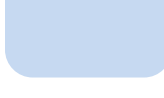
```
num = int(input("Enter a number: "))
```

```
if (num % 2) == 0:
```

```
 print(num, "is even")
```

```
else:
```

```
 print(num, "is odd")
```

```
 print("and so are you!")
```

if-else statement

Any Boolean expression
or variable

if condition:

statements (block) # condition True

else:

statements (block) # condition False

```
→ num = int(input("Enter a number: "))
if (num % 2) == 0:
    print(num, "is even")
else:
    print(num, "is odd")
    print("and so are you!")
```

Enter a number:

if-else statement

Any Boolean expression
or variable

if condition:

statements (block) # condition True

else:

statements (block) # condition False

```
→ num = int(input("Enter a number: "))
if (num % 2) == 0:
    print(num, "is even")
else:
    print(num, "is odd")
    print("and so are you!")
```

Enter a number: 4

if-else statement

Any Boolean expression
or variable

if condition:

statements (block) # condition True

else:

statements (block) # condition False

```
num = int(input("Enter a number: "))
```

```
→ if (num % 2) == 0:  
    print(num, "is even")
```

```
else:  
    print(num, "is odd")  
    print("and so are you!")
```

```
Enter a number: 4
```

if-else statement

Any Boolean expression
or variable

if condition:

statements (block) # condition True

else:

statements (block) # condition False

```
num = int(input("Enter a number: "))
```

```
if (num % 2) == 0:
```

```
    print(num, "is even")
```

```
else:
```

```
    print(num, "is odd")
```

```
    print("and so are you!")
```

```
Enter a number: 4
```

```
4 is even
```


if-else statement

Any Boolean expression
or variable

if condition:

statements (block) # condition True

else:

statements (block) # condition False

```
num = int(input("Enter a number: "))
```

```
if (num % 2) == 0:
```

```
    print(num, "is even")
```

```
else:
```

```
    print(num, "is odd")
```

```
    print("and so are you!")
```

```
Enter a number: 4
```

```
4 is even
```

if-else statement

Any Boolean expression
or variable

if condition:

statements (block) # condition True

else:

statements (block) # condition False

```
→ num = int(input("Enter a number: "))
if (num % 2) == 0:
    print(num, "is even")
else:
    print(num, "is odd")
    print("and so are you!")
```

Enter a number:

if-else statement

Any Boolean expression
or variable

if condition:

statements (block) # condition True

else:

statements (block) # condition False

```
→ num = int(input("Enter a number: "))
if (num % 2) == 0:
    print(num, "is even")
else:
    print(num, "is odd")
    print("and so are you!")
```

Enter a number: 11

if-else statement

Any Boolean expression
or variable

if condition:

statements (block) # condition True

else:

statements (block) # condition False

```
num = int(input("Enter a number: "))
```

```
→ if (num % 2) == 0:  
    print(num, "is even")
```

```
else:  
    print(num, "is odd")  
    print("and so are you!")
```

```
Enter a number: 11
```

if-else statement

Any Boolean expression
or variable

if condition:

statements (block) # condition True

else:

statements (block) # condition False

```
num = int(input("Enter a number: "))
```

```
if (num % 2) == 0:
```

```
    print(num, "is even")
```

→ else:

```
    print(num, "is odd")
```

```
    print("and so are you!")
```

Enter a number: 11

if-else statement

Any Boolean expression
or variable

if condition:

statements (block) # condition True

else:

statements (block) # condition False

```
num = int(input("Enter a number: "))
```

```
if (num % 2) == 0:
```

```
    print(num, "is even")
```

```
else:
```

```
    print(num, "is odd")
```

```
    print("and so are you!")
```

```
Enter a number: 11
```

```
11 is odd
```

if-else statement

Any Boolean expression
or variable

if condition:

statements (block) # condition True

else:

statements (block) # condition False

```
num = int(input("Enter a number: "))
```

```
if (num % 2) == 0:
```

```
    print(num, "is even")
```

```
else:
```

```
    print(num, "is odd")
```

```
    print("and so are you!")
```

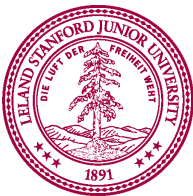
```
Enter a number: 11
```

```
11 is odd
```

```
and so are you!
```

Nested `if-else` statements

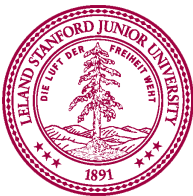
```
num = int(input("Enter a number: "))
if num == 0:
    print("Zero is my hero!")
else:
    if num > 0:
        print(num, "is positive")
    else:
        print(num, "is negative")
```



Nested `if-else` statements

```
→ num = int(input("Enter a number: "))  
if num == 0:  
    print("Zero is my hero!")  
else:  
    if num > 0:  
        print(num, "is positive")  
    else:  
        print(num, "is negative")
```

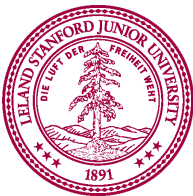
Enter a number:



Nested `if-else` statements

```
→ num = int(input("Enter a number: "))  
if num == 0:  
    print("Zero is my hero!")  
else:  
    if num > 0:  
        print(num, "is positive")  
    else:  
        print(num, "is negative")
```

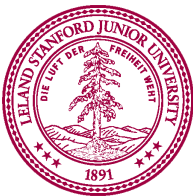
Enter a number: 6



Nested `if-else` statements

```
num = int(input("Enter a number: "))  
→ if num == 0:  
    print("Zero is my hero!")  
else:  
    if num > 0:  
        print(num, "is positive")  
    else:  
        print(num, "is negative")
```

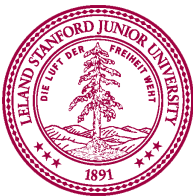
Enter a number: 6



Nested `if-else` statements

```
num = int(input("Enter a number: "))
if num == 0:
    print("Zero is my hero!")
→ else:
    if num > 0:
        print(num, "is positive")
    else:
        print(num, "is negative")
```

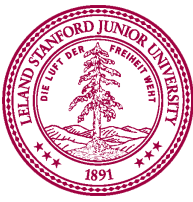
Enter a number: 6



Nested `if-else` statements

```
num = int(input("Enter a number: "))
if num == 0:
    print("Zero is my hero!")
else:
    if num > 0:
        print(num, "is positive")
    else:
        print(num, "is negative")
```

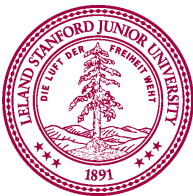
Enter a number: 6



Nested `if-else` statements

```
num = int(input("Enter a number: "))
if num == 0:
    print("Zero is my hero!")
else:
    if num > 0:
        print(num, "is positive")
    else:
        print(num, "is negative")
```

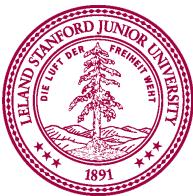
Enter a number: 6
6 is positive



Nested `if-else` statements

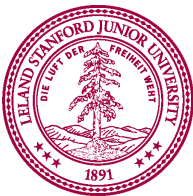
```
num = int(input("Enter a number: "))
if num == 0:
    print("Zero is my hero!")
else:
    if num > 0:
        print(num, "is positive")
    else:
        print(num, "is negative")
```

```
Enter a number: 6
6 is positive
```



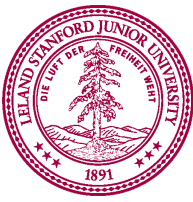
Bring me the `elif`

```
num = int(input("Enter a number: "))
if num == 0:
    print("Zero is my hero!")
else:
    if num > 0:
        print(num, "is positive")
    else:
        print(num, "is negative")
```



Bring me the `elif`

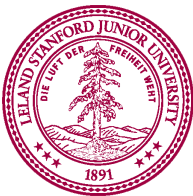
```
num = int(input("Enter a number: "))  
if num == 0:  
    print("Zero is my hero!")  
else:  
    if num > 0:  
        print(num, "is positive")  
    else:  
        print(num, "is negative")
```



Bring me the `elif`

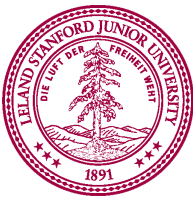
```
num = int(input("Enter a number: "))
if num == 0:
    print("Zero is my hero!")
else:
    if num > 0:
        print(num, "is positive")
    else:
        print(num, "is negative")
```

→ `elif num > 0:`



Bring me the `elif`

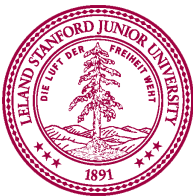
```
num = int(input("Enter a number: "))  
if num == 0:  
    print("Zero is my hero!")  
elif num > 0:  
    print(num, "is positive")  
else:  
    print(num, "is negative")
```



Bring me the `elif`

```
→ num = int(input("Enter a number: "))  
if num == 0:  
    print("Zero is my hero!")  
elif num > 0:  
    print(num, "is positive")  
else:  
    print(num, "is negative")
```

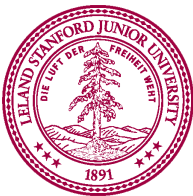
Enter a number:



Bring me the `elif`

```
→ num = int(input("Enter a number: "))  
if num == 0:  
    print("Zero is my hero!")  
elif num > 0:  
    print(num, "is positive")  
else:  
    print(num, "is negative")
```

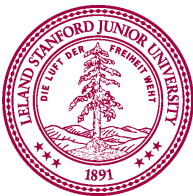
Enter a number: -2



Bring me the `elif`

```
num = int(input("Enter a number: "))  
→ if num == 0:  
    print("Zero is my hero!")  
elif num > 0:  
    print(num, "is positive")  
else:  
    print(num, "is negative")
```

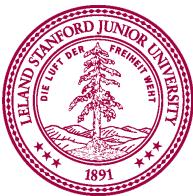
Enter a number: -2



Bring me the `elif`

```
num = int(input("Enter a number: "))  
if num == 0:  
    print("Zero is my hero!")  
→ elif num > 0:  
    print(num, "is positive")  
else:  
    print(num, "is negative")
```

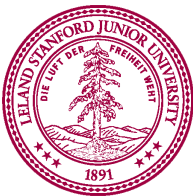
Enter a number: -2



Bring me the `elif`

```
num = int(input("Enter a number: "))  
if num == 0:  
    print("Zero is my hero!")  
elif num > 0:  
    print(num, "is positive")  
→ else:  
    print(num, "is negative")
```

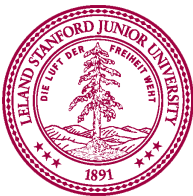
Enter a number: -2



Bring me the `elif`

```
num = int(input("Enter a number: "))  
if num == 0:  
    print("Zero is my hero!")  
elif num > 0:  
    print(num, "is positive")  
else:  
→    print(num, "is negative")
```

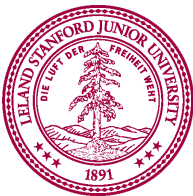
```
Enter a number: -2  
-2 is negative
```



Bring me the `elif`

```
num = int(input("Enter a number: "))  
if num == 0:  
    print("Zero is my hero!")  
elif num > 0:  
    print(num, "is positive")  
else:  
    print(num, "is negative")
```

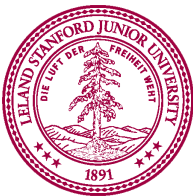
```
Enter a number: -2  
-2 is negative
```



Bring me the `elif`

```
→ num = int(input("Enter a number: "))  
if num == 0:  
    print("Zero is my hero!")  
elif num > 0:  
    print(num, "is positive")  
else:  
    print(num, "is negative")
```

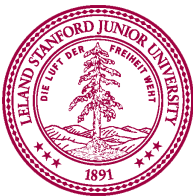
Enter a number:



Bring me the `elif`

```
→ num = int(input("Enter a number: "))  
if num == 0:  
    print("Zero is my hero!")  
elif num > 0:  
    print(num, "is positive")  
else:  
    print(num, "is negative")
```

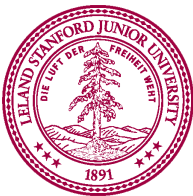
Enter a number: 0



Bring me the `elif`

```
num = int(input("Enter a number: "))  
→ if num == 0:  
    print("Zero is my hero!")  
elif num > 0:  
    print(num, "is positive")  
else:  
    print(num, "is negative")
```

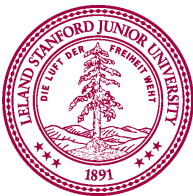
Enter a number: 0



Bring me the `elif`

```
num = int(input("Enter a number: "))  
if num == 0:  
→   print("Zero is my hero!")  
elif num > 0:  
    print(num, "is positive")  
else:  
    print(num, "is negative")
```

```
Enter a number: 0  
Zero is my hero!
```

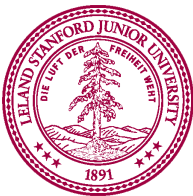


Bring me the `elif`

```
num = int(input("Enter a number: "))  
if num == 0:  
    print("Zero is my hero!")  
elif num > 0:  
    print(num, "is positive")  
else:  
    print(num, "is negative")
```

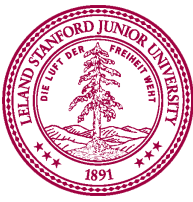
Done!

```
Enter a number: 0  
Zero is my hero!
```



It works for grades too!

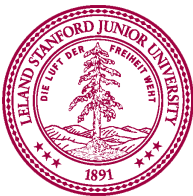
```
grade = int(input("Enter a score: "))  
if grade >= 90:  
    print("A")  
elif grade >= 80:  
    print("B")  
elif grade >= 70:  
    print("C")  
else:  
    print("Keep trying!")
```



It works for grades too!

```
→ grade = int(input("Enter a score: "))  
if grade >= 90:  
    print("A")  
elif grade >= 80:  
    print("B")  
elif grade >= 70:  
    print("C")  
else:  
    print("Keep trying!")
```

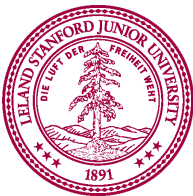
Enter a score:



It works for grades too!

```
→ grade = int(input("Enter a score: "))  
if grade >= 90:  
    print("A")  
elif grade >= 80:  
    print("B")  
elif grade >= 70:  
    print("C")  
else:  
    print("Keep trying!")
```

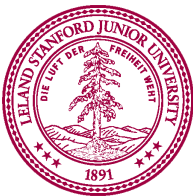
Enter a score: 82



It works for grades too!

```
grade = int(input("Enter a score: "))  
→ if grade >= 90:  
    print("A")  
elif grade >= 80:  
    print("B")  
elif grade >= 70:  
    print("C")  
else:  
    print("Keep trying!")
```

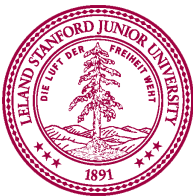
Enter a score: 82



It works for grades too!

```
grade = int(input("Enter a score: "))
if grade >= 90:
    print("A")
→ elif grade >= 80:
    print("B")
elif grade >= 70:
    print("C")
else:
    print("Keep trying!")
```

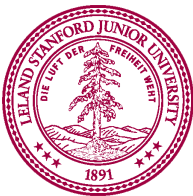
Enter a score: 82



It works for grades too!

```
grade = int(input("Enter a score: "))
if grade >= 90:
    print("A")
elif grade >= 80:
    → print("B")
elif grade >= 70:
    print("C")
else:
    print("Keep trying!")
```

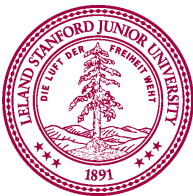
```
Enter a score: 82
B
```



It works for grades too!

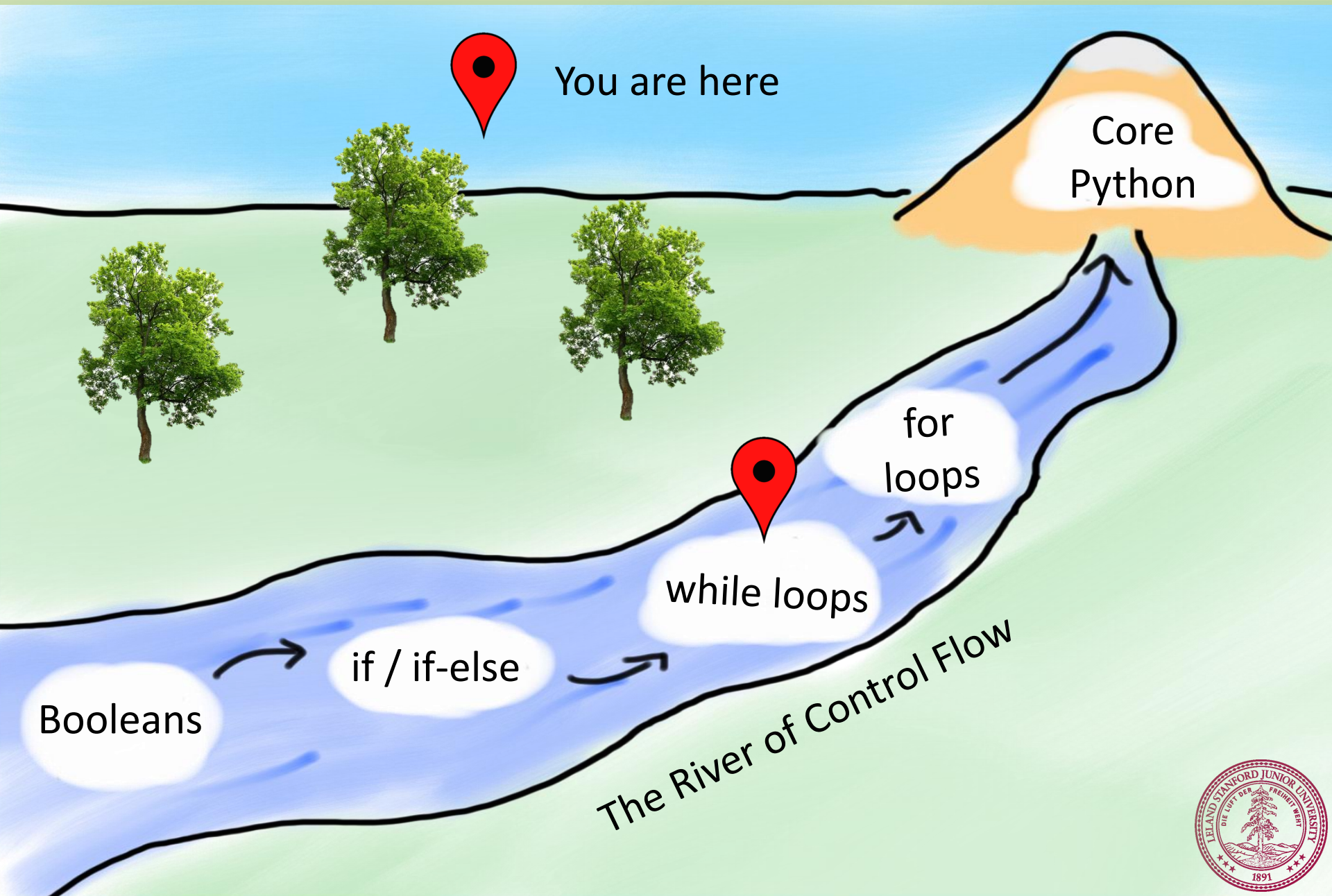
```
grade = int(input("Enter a score: "))  
if grade >= 90:  
    print("A")  
elif grade >= 80:  
    print("B")  
elif grade >= 70:  
    print("C")  
else:  
    print("Keep trying!")
```

```
Enter a score: 82  
B
```



Amazing!

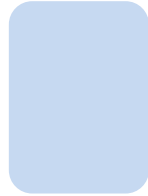
Today's Route



while loop

Any Boolean expression
or variable

while *condition*:

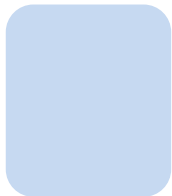


statements
(*block*)

note indenting

```
x = 10
```

```
while x > 1:
```



```
    x /= 2
```

```
    print(x)
```

while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

→ **x** = 10
while **x** > 1:
 x /= 2
 print(**x**)



while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

```
x = 10
```

```
→ while x > 1:  
    x /= 2  
    print(x)
```



while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

```
x = 10
```

```
while x > 1:
```

```
    x /= 2
```

```
    print(x)
```



while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

```
x = 10
while x > 1:
    x /= 2
    print(x)
```

5.0

while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

```
x = 10
```

```
→ while x > 1:  
    x /= 2  
    print(x)
```

```
5.0
```

while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (block)

note indenting

```
x = 10
while x > 1:
    x /= 2
    print(x)
```

5.0

while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

```
x = 10
while x > 1:
    x /= 2
    print(x)
```

5.0
2.5

while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

```
x = 10
```

```
→ while x > 1:  
    x /= 2  
    print(x)
```

```
5.0
```

```
2.5
```

while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

```
x = 10
while x > 1:
    x /= 2
    print(x)
```

```
5.0
2.5
```

while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

```
x = 10
while x > 1:
    x /= 2
    print(x)
```

```
5.0
2.5
1.25
```

while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

```
x = 10
```

```
→ while x > 1:  
    x /= 2  
    print(x)
```

```
5.0  
2.5  
1.25
```

while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

```
x = 10
while x > 1:
    x /= 2
    print(x)
```

```
5.0
2.5
1.25
```

while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

```
x = 10
while x > 1:
    x /= 2
    print(x)
```

```
5.0
2.5
1.25
0.625
```

while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

```
x = 10
```

```
→ while x > 1:  
    x /= 2  
    print(x)
```

```
5.0  
2.5  
1.25  
0.625
```

while loop

Any Boolean expression
or variable

while *condition*:
 statements
 (*block*)

note indenting

```
x = 10
```

```
while x > 1:
```

```
    x /= 2
```

```
    print(x)
```

Done!

```
5.0
```

```
2.5
```

```
1.25
```

```
0.625
```

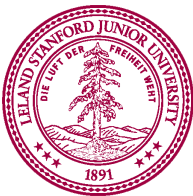

Conditions in Python



You can use **if**, **if-else**, and **while** statements in Python.

They are the same as in Karel, except that the *condition* can be any expression that evaluates to

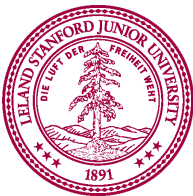
True or **False**
(Boolean)



A Loop-and-a-Half?

```
SENTINEL = -1
```

```
def main():  
    total = 0  
    value = int(input("Value? "))  
    while value != SENTINEL:  
        total += value  
        value = int(input("Value? "))  
    print("Total = " + str(total))
```



A Loop-and-a-Half?

SENTINEL = -1

total

value

0

def main():

→ total = 0

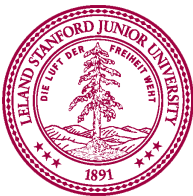
value = int(input("Value? "))

while value != SENTINEL:

total += value

value = int(input("Value? "))

print("Total = " + str(total))



A Loop-and-a-Half?

SENTINEL = -1

total

value

0

```
def main():
```

```
    total = 0
```

```
    value = int(input("Value? "))
```

```
    while value != SENTINEL:
```

```
        total += value
```

```
        value = int(input("Value? "))
```

```
    print("Total = " + str(total))
```

Value?



A Loop-and-a-Half?

SENTINEL = -1

total

0

value

4

```
def main():
```

```
    total = 0
```

```
    value = int(input("Value? "))
```

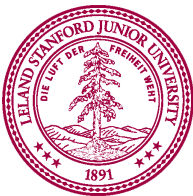
```
    while value != SENTINEL:
```

```
        total += value
```

```
        value = int(input("Value? "))
```

```
    print("Total = " + str(total))
```

Value? 4



A Loop-and-a-Half?

SENTINEL = -1

total

0

value

4

```
def main():
```

```
    total = 0
```

```
    value = int(input("Value? "))
```

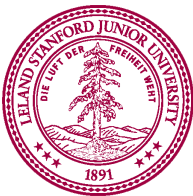
```
    while value != SENTINEL:
```

```
        total += value
```

```
        value = int(input("Value? "))
```

```
    print("Total = " + str(total))
```

Value? 4



A Loop-and-a-Half?

SENTINEL = -1

total

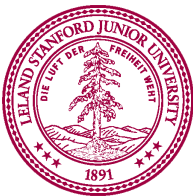
value

4

4

```
def main():  
    total = 0  
    value = int(input("Value? "))  
    while value != SENTINEL:  
        total += value  
        value = int(input("Value? "))  
    print("Total = " + str(total))
```

Value? 4



A Loop-and-a-Half?

SENTINEL = -1

total

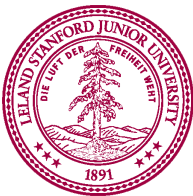
value

4

4

```
def main():  
    total = 0  
    value = int(input("Value? "))  
    while value != SENTINEL:  
        total += value  
        value = int(input("Value? "))  
    print("Total = " + str(total))
```

Value? 4
Value?



A Loop-and-a-Half?

SENTINEL = -1

total

4

value

6

```
def main():  
    total = 0  
    value = int(input("Value? "))  
    while value != SENTINEL:  
        total += value  
        value = int(input("Value? "))  
    print("Total = " + str(total))
```

Value? 4

Value? 6



A Loop-and-a-Half?

SENTINEL = -1

total

4

value

6

```
def main():
```

```
    total = 0
```

```
    value = int(input("Value? "))
```

```
→ while value != SENTINEL:
```

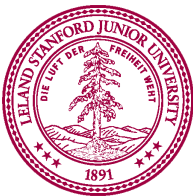
```
        total += value
```

```
        value = int(input("Value? "))
```

```
    print("Total = " + str(total))
```

Value? 4

Value? 6



A Loop-and-a-Half?

SENTINEL = -1

total

10

value

6

```
def main():  
    total = 0  
    value = int(input("Value? "))  
    while value != SENTINEL:  
        total += value  
        value = int(input("Value? "))  
    print("Total = " + str(total))
```

Value? 4

Value? 6



A Loop-and-a-Half?

SENTINEL = -1

total

10

value

6

```
def main():  
    total = 0  
    value = int(input("Value? "))  
    while value != SENTINEL:  
        total += value  
        value = int(input("Value? "))  
    print("Total = " + str(total))
```

Value? 4

Value? 6

Value?



A Loop-and-a-Half?

SENTINEL = -1

total

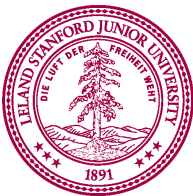
value

10

11

```
def main():  
    total = 0  
    value = int(input("Value? "))  
    while value != SENTINEL:  
        total += value  
        value = int(input("Value? "))  
    print("Total = " + str(total))
```

Value? 4
Value? 6
Value? 11



A Loop-and-a-Half?

SENTINEL = -1

total

value

10

11

```
def main():
```

```
    total = 0
```

```
    value = int(input("Value? "))
```

```
    while value != SENTINEL:
```

```
        total += value
```

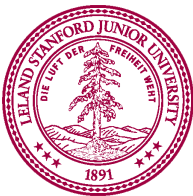
```
        value = int(input("Value? "))
```

```
    print("Total = " + str(total))
```

Value? 4

Value? 6

Value? 11



A Loop-and-a-Half?

SENTINEL = -1

total

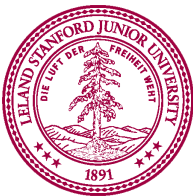
value

21

11

```
def main():  
    total = 0  
    value = int(input("Value? "))  
    while value != SENTINEL:  
        total += value  
        value = int(input("Value? "))  
    print("Total = " + str(total))
```

```
Value? 4  
Value? 6  
Value? 11
```



A Loop-and-a-Half?

SENTINEL = -1

total

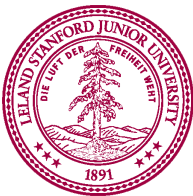
value

21

11

```
def main():  
    total = 0  
    value = int(input("Value? "))  
    while value != SENTINEL:  
        total += value  
        value = int(input("Value? "))  
    print("Total = " + str(total))
```

```
Value? 4  
Value? 6  
Value? 11  
Value?
```



A Loop-and-a-Half?

SENTINEL = -1

total

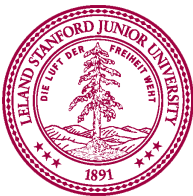
value

21

-1

```
def main():  
    total = 0  
    value = int(input("Value? "))  
    while value != SENTINEL:  
        total += value  
        value = int(input("Value? "))  
    print("Total = " + str(total))
```

```
Value? 4  
Value? 6  
Value? 11  
Value? -1
```



A Loop-and-a-Half?

SENTINEL = -1

total

21

value

-1

```
def main():
```

```
    total = 0
```

```
    value = int(input("Value? "))
```

```
→    while value != SENTINEL:
```

```
        total += value
```

```
        value = int(input("Value? "))
```

```
    print("Total = " + str(total))
```

Value? 4

Value? 6

Value? 11

Value? -1



A Loop-and-a-Half?

SENTINEL = -1

total

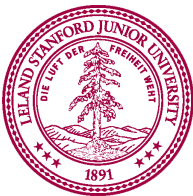
value

21

-1

```
def main():  
    total = 0  
    value = int(input("Value? "))  
    while value != SENTINEL:  
        total += value  
        value = int(input("Value? "))  
→ print("Total = " + str(total))
```

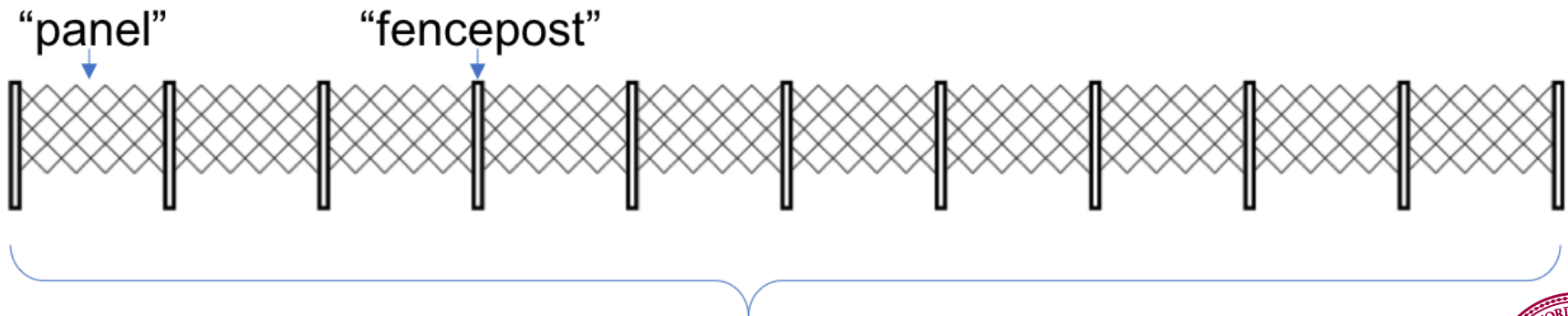
```
Value? 4  
Value? 6  
Value? 11  
Value? -1  
Total = 21
```



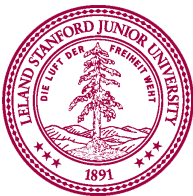
A Loop-and-a-Half?

```
SENTINEL = -1
```

```
def main():  
    total = 0  
    value = int(input("Value? "))  
    while value != SENTINEL:  
        total += value  
        value = int(input("Value? "))  
    print("Total = " + str(total))
```



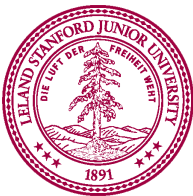
10 panels, 11 fenceposts



A Loop-and-a-Half?

```
SENTINEL = -1
```

```
def main():  
    total = 0  
    value = int(input("Value? "))  
    while value != SENTINEL:  
        total += value  
        value = int(input("Value? "))  
    print("Total = " + str(total))
```



A Loop-and-a-Half?

```
SENTINEL = -1
```

```
def main():
```

```
    total = 0
```

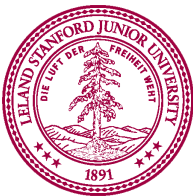
```
    while value != SENTINEL:
```

```
        total += value
```

```
        value = int(input("Value? "))
```

```
    print("Total = " + str(total))
```

UnboundLocalError: local variable 'value' referenced before assignment



A Loop-and-a-Half?

```
SENTINEL = -1
```

```
def main():
```

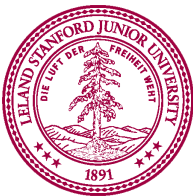
```
    total = 0
```

```
    while True:
```

```
        total += value
```

```
        value = int(input("Value? "))
```

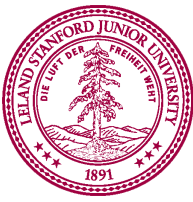
```
    print("Total = " + str(total))
```



A Loop-and-a-Half?

```
SENTINEL = -1
```

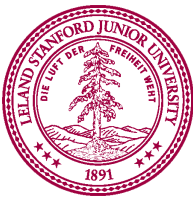
```
def main():  
    total = 0  
    while True:  
  
        total += value  
        value = int(input("Value? "))  
    print("Total = " + str(total))
```



A Loop-and-a-Half?

```
SENTINEL = -1
```

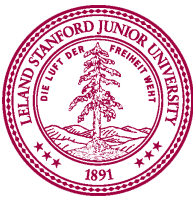
```
def main():  
    total = 0  
    while True:  
        value = int(input("Value? "))  
        total += value  
  
    print("Total = " + str(total))
```



A Loop-and-a-Half?

```
SENTINEL = -1
```

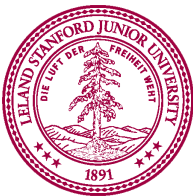
```
def main():  
    total = 0  
    while True:  
        value = int(input("Value? "))  
  
        total += value  
    print("Total = " + str(total))
```



A Loop-and-a-Half?

```
SENTINEL = -1
```

```
def main():  
    total = 0  
    while True:  
        value = int(input("Value? "))  
        if value == SENTINEL:  
            break  
        total += value  
    print("Total = " + str(total))
```



A Loop-and-a-Half?

```
SENTINEL = -1
```

```
def main():
```

```
    total = 0
```

```
    while True:
```

```
        value = int(input("Value? "))
```

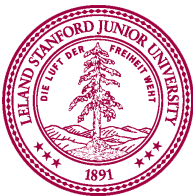
```
        if value == SENTINEL:
```

```
            break
```

```
        total += value
```

```
    print("Total = " + str(total))
```

**Immediately break out
of current loop**



A Loop-and-a-Half?

SENTINEL = -1

total

value

0

def main():

total = 0

while True:

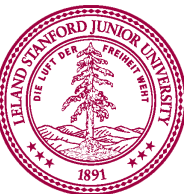
value = int(input("Value? "))

if value == SENTINEL:

break

total += value

print("Total = " + str(total))



A Loop-and-a-Half?

SENTINEL = -1

total

value

0

```
def main():
```

```
    total = 0
```

```
    while True:
```

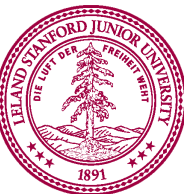
```
        value = int(input("Value? "))
```

```
        if value == SENTINEL:
```

```
            break
```

```
        total += value
```

```
    print("Total = " + str(total))
```



A Loop-and-a-Half?

SENTINEL = -1

total

value

0

```
def main():
```

```
    total = 0
```

```
    while True:
```

```
        value = int(input("Value? "))
```

```
        if value == SENTINEL:
```

```
            break
```

```
        total += value
```

```
    print("Total = " + str(total))
```

Value?



A Loop-and-a-Half?

SENTINEL = -1

total

value

0

4

```
def main():  
    total = 0  
    while True:  
        value = int(input("Value? "))  
        if value == SENTINEL:  
            break  
        total += value  
    print("Total = " + str(total))
```

Value? 4



A Loop-and-a-Half?

SENTINEL = -1

total

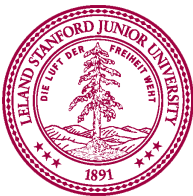
value

0

4

```
def main():  
    total = 0  
    while True:  
        value = int(input("Value? "))  
        if value == SENTINEL:  
            break  
        total += value  
    print("Total = " + str(total))
```

Value? 4



A Loop-and-a-Half?

SENTINEL = -1

total

value

4

4

```
def main():  
    total = 0  
    while True:  
        value = int(input("Value? "))  
        if value == SENTINEL:  
            break  
        total += value  
    print("Total = " + str(total))
```

Value? 4



A Loop-and-a-Half?

SENTINEL = -1

total

value

4

4

```
def main():
```

```
    total = 0
```

```
    while True:
```

```
        value = int(input("Value? "))
```

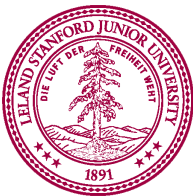
```
        if value == SENTINEL:
```

```
            break
```

```
        total += value
```

```
    print("Total = " + str(total))
```

Value? 4



A Loop-and-a-Half?

SENTINEL = -1

total

value

4

6

```
def main():  
    total = 0  
    while True:  
        value = int(input("Value? "))  
        if value == SENTINEL:  
            break  
        total += value  
    print("Total = " + str(total))
```

Value? 4

Value? 6



A Loop-and-a-Half?

SENTINEL = -1

total

value

4

6

```
def main():  
    total = 0  
    while True:  
        value = int(input("Value? "))  
        if value == SENTINEL:  
            break  
        total += value  
    print("Total = " + str(total))
```

Value? 4

Value? 6



A Loop-and-a-Half?

SENTINEL = -1

total

value

10

6

```
def main():  
    total = 0  
    while True:  
        value = int(input("Value? "))  
        if value == SENTINEL:  
            break  
        total += value  
    print("Total = " + str(total))
```

Value? 4

Value? 6



A Loop-and-a-Half?

SENTINEL = -1

total

value

10

6

```
def main():
```

```
    total = 0
```

```
    while True:
```

```
        value = int(input("Value? "))
```

```
        if value == SENTINEL:
```

```
            break
```

```
        total += value
```

```
    print("Total = " + str(total))
```

Value? 4

Value? 6



A Loop-and-a-Half?

SENTINEL = -1

total

10

value

11

```
def main():  
    total = 0  
    while True:  
→       value = int(input("Value? "))  
        if value == SENTINEL:  
            break  
        total += value  
    print("Total = " + str(total))
```

```
Value? 4  
Value? 6  
Value? 11
```



A Loop-and-a-Half?

SENTINEL = -1

total

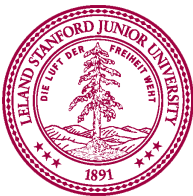
value

10

11

```
def main():  
    total = 0  
    while True:  
        value = int(input("Value? "))  
        if value == SENTINEL:  
            break  
        total += value  
    print("Total = " + str(total))
```

```
Value? 4  
Value? 6  
Value? 11
```



A Loop-and-a-Half?

SENTINEL = -1

total

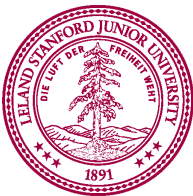
value

21

11

```
def main():  
    total = 0  
    while True:  
        value = int(input("Value? "))  
        if value == SENTINEL:  
            break  
        total += value  
    print("Total = " + str(total))
```

Value? 4
Value? 6
Value? 11



A Loop-and-a-Half?

SENTINEL = -1

total

value

21

11

```
def main():
```

```
    total = 0
```

```
    while True:
```

```
        value = int(input("Value? "))
```

```
        if value == SENTINEL:
```

```
            break
```

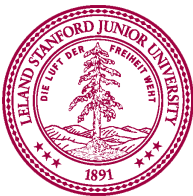
```
        total += value
```

```
    print("Total = " + str(total))
```

Value? 4

Value? 6

Value? 11



A Loop-and-a-Half?

SENTINEL = -1

total

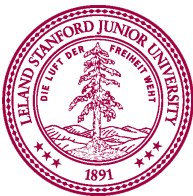
value

21

-1

```
def main():  
    total = 0  
    while True:  
→       value = int(input("Value? "))  
        if value == SENTINEL:  
            break  
        total += value  
    print("Total = " + str(total))
```

```
Value? 4  
Value? 6  
Value? 11  
Value? -1
```



A Loop-and-a-Half?

SENTINEL = -1

total

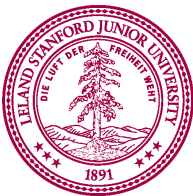
value

21

-1

```
def main():  
    total = 0  
    while True:  
        value = int(input("Value? "))  
        if value == SENTINEL:  
            break  
        total += value  
    print("Total = " + str(total))
```

```
Value? 4  
Value? 6  
Value? 11  
Value? -1
```



A Loop-and-a-Half?

SENTINEL = -1

total

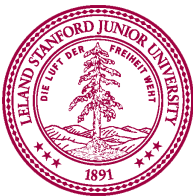
value

21

-1

```
def main():
    total = 0
    while True:
        value = int(input("Value? "))
        if value == SENTINEL:
            break
        total += value
    print("Total = " + str(total))
```

Value? 4
Value? 6
Value? 11
Value? -1



A Loop-and-a-Half?

SENTINEL = -1

total

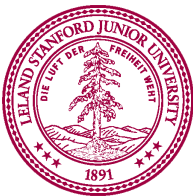
value

21

-1

```
def main():  
    total = 0  
    while True:  
        value = int(input("Value? "))  
        if value == SENTINEL:  
            break  
        total += value  
→ print("Total = " + str(total))
```

```
Value? 4  
Value? 6  
Value? 11  
Value? -1  
Total = 21
```



A Loop-and-a-Half?

SENTINEL = -1

total

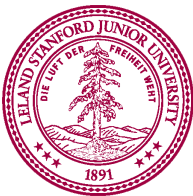
value

21

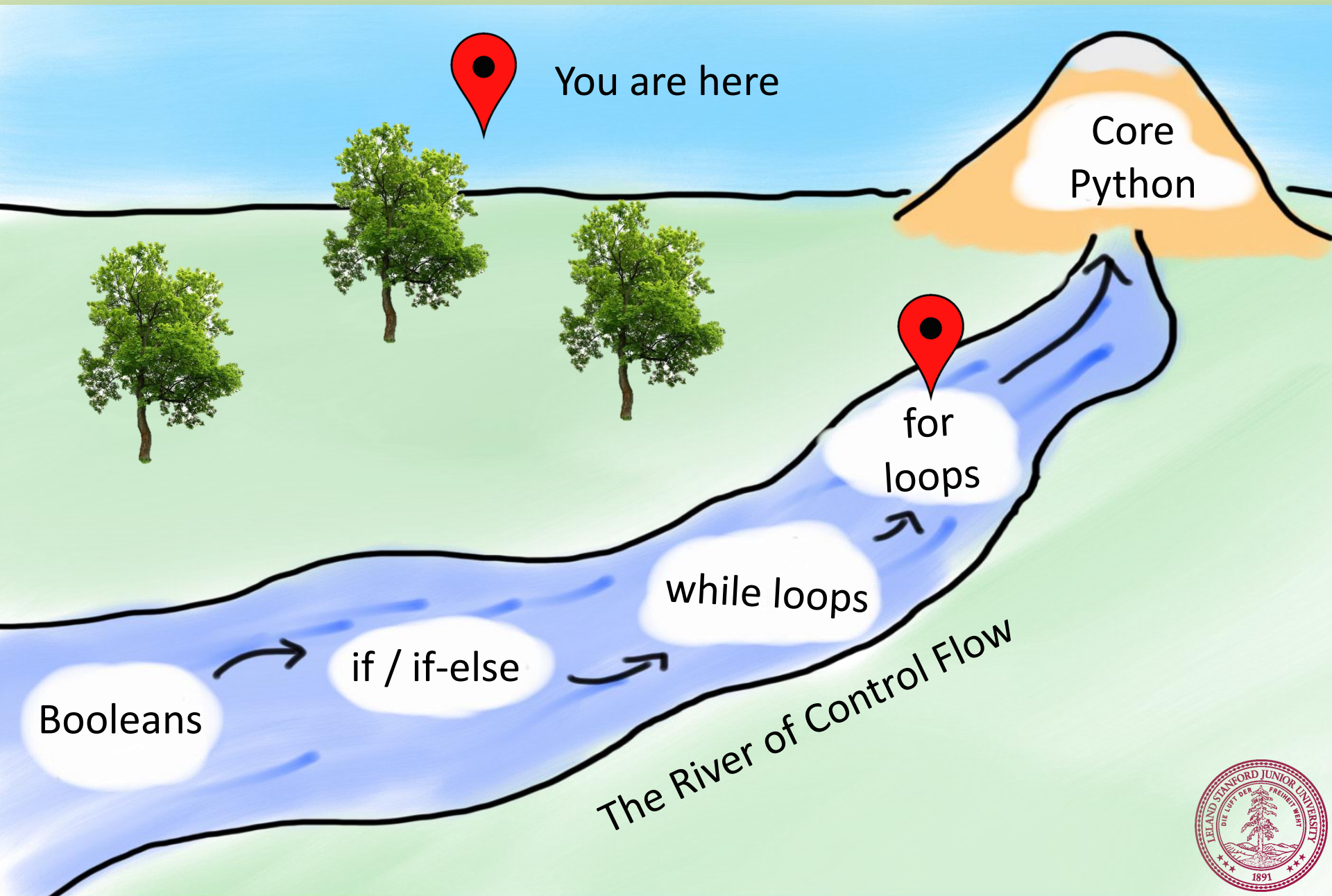
-1

```
def main():
    total = 0
    while True:
        value = int(input("Value? "))
        if value == SENTINEL:
            break
        total += value
    print("Total = " + str(total))
```

```
Value? 4
Value? 6
Value? 11
Value? -1
Total = 21
```



Today's Route



for loop

This is called an index variable.
Can have any variable name.



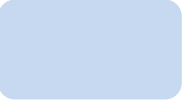
```
for i in range(count) :
```



```
    statements  
    (block)
```

```
# note indenting
```

```
for i in range(3) :
```



```
    print(i)
```

for loop

This is called an index variable.
Can have any variable name.



```
for i in range(count) :
```



```
    statements  
    (block)
```

note indenting

```
for i in range(3) :  
    print(i)
```

Index starts counting from 0
Counts up to (`count - 1`)

for loop

This is called an index variable.
Can have any variable name.



```
for i in range(count) :  
    statements                # note indenting  
    (block)
```



```
for i in range(3) :  
    print(i)
```

Index starts counting from 0
Counts up to (count - 1)

i
0



for loop

This is called an index variable.
Can have any variable name.



```
for i in range(count) :  
    statements                # note indenting  
    (block)
```



```
for i in range(3) :  
    print(i)
```

Index starts counting from 0
Counts up to (count - 1)

i
0

0

for loop

This is called an index variable.
Can have any variable name.



```
for i in range(count) :  
    statements                # note indenting  
    (block)
```



```
for i in range(3) :  
    print(i)
```

Index starts counting from 0
Counts up to (count - 1)

i
1

0

for loop

This is called an index variable.
Can have any variable name.



```
for i in range(count) :  
    statements                # note indenting  
    (block)
```



```
for i in range(3) :  
    print(i)
```

Index starts counting from 0
Counts up to (count - 1)

i
1

0
1

for loop

This is called an index variable.
Can have any variable name.



```
for i in range(count) :  
    statements                # note indenting  
    (block)
```



```
for i in range(3) :  
    print(i)
```

Index starts counting from 0
Counts up to (count - 1)

i

2

0

1

for loop

This is called an index variable.
Can have any variable name.



```
for i in range(count) :  
    statements                # note indenting  
    (block)
```



```
for i in range(3) :  
    print(i)
```

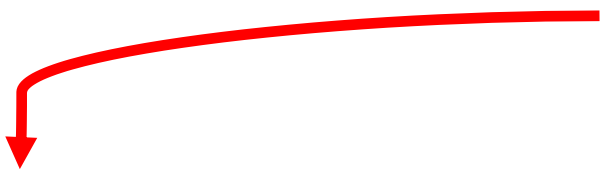
Index starts counting from 0
Counts up to (count - 1)

i
2

0
1
2

for loop

This is called an index variable.
Can have any variable name.



```
for i in range(count) :  
    statements                # note indenting  
    (block)
```

```
for i in range(3) :  
    print(i)
```

Index starts counting from 0
Counts up to (count - 1)

Done!

i
2

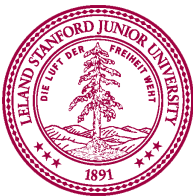
0
1
2

Write a program to print
“Python rocks my socks” 100 times

for loop redux

i starts at 0, counts up to 99 (inclusive)

```
def main():  
    for i in range(100):  
        print("Python rocks my socks")
```



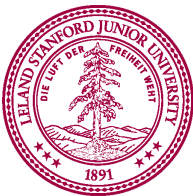
for loop redux

i starts at 0, counts up to 99 (inclusive)

```
def main():  
    for i in range(100):  
        print("Python rocks my socks")
```

Can think of it like this:

```
def main():  
    i = 0  
    while i < 100:  
        print("Python rocks my socks")  
        i += 1
```



for loops – the advanced course!

```
for i in range (start, end, step) :
```



```
    statements  
    (block)
```

Starts counting at start

Stops counting before end

Each iteration, add step to counter

```
for i in range (1, 7, 2) :  
    print(i)
```

for loops – the advanced course!

```
for i in range (start, end, step) :
```



```
    statements  
    (block)
```

Starts counting at start

Stops counting before end

Each iteration, add step to counter

```
→ for i in range (1, 7, 2) :  
    print(i)
```

i

1

for loops – the advanced course!

```
for i in range (start, end, step) :
```



```
    statements  
    (block)
```

Starts counting at start

Stops counting before end

Each iteration, add step to counter

```
for i in range (1, 7, 2) :
```



```
    print(i)
```

i

1

1

for loops – the advanced course!

```
for i in range (start, end, step) :
```



```
    statements  
    (block)
```

Starts counting at start

Stops counting before end

Each iteration, add step to counter

```
→ for i in range (1, 7, 2) :  
    print(i)
```

i

3

1

for loops – the advanced course!

```
for i in range (start, end, step) :
```



```
    statements  
    (block)
```

Starts counting at start

Stops counting before end

Each iteration, add step to counter

```
for i in range (1, 7, 2) :
```



```
    print(i)
```

i

3

1

3

for loops – the advanced course!

```
for i in range (start, end, step) :
```



```
    statements  
    (block)
```

Starts counting at start

Stops counting before end

Each iteration, add step to counter

```
→ for i in range (1, 7, 2) :  
    print(i)
```

i

5

1

3

for loops – the advanced course!

```
for i in range (start, end, step) :
```



```
    statements  
    (block)
```

Starts counting at start

Stops counting before end

Each iteration, add step to counter

```
for i in range (1, 7, 2) :
```



```
    print(i)
```

i

5

1

3

5

for loops – the advanced course!

```
for i in range (start, end, step) :
```



```
    statements  
    (block)
```

Starts counting at start

Stops counting before end

Each iteration, add step to counter

```
for i in range (1, 7, 2) :
```

```
    print(i)
```

Done!

i

5

```
1  
3  
5
```

Putting it all together:
guessnumber.py

Today's Goal

1. Be able to use For / While / If in Python



A close-up photograph of a baby with light brown hair and dark eyes, looking directly at the camera with a grumpy or pouting expression. The baby is wearing a white long-sleeved shirt with a bright green collar and green sleeves. They are holding a small handful of sand in their right fist. The background is a blurred outdoor setting, likely a beach with sand and some greenery in the distance.

Have a good
weekend!