# As you walk in

## Quiz will start at the beginning of lecture

- Have pencil/pen ready
- Don’t use your own scratch paper
  - We have some if you need it
- Silence phones

# Quiz

## Before we start

- Don’t open until we start
- Make sure your name and Andrew ID are on the front
- Read instruction page
- No questions (unless clarification on English)

## Additional info

- 25 min

# 15-112 Lecture 2

## Week 2 Tue Strings

Instructor: Pat Virtue

# Post-quiz Exercise

What is the correct response to the following?

```
pet = "manatee"
s = pet[:4]*2
```

Then Google search: `s`

# Announcements

## Assignments

Week 4 Pre-reading Checkpoint

- Pre-reading out today
- Checkpoint out Wed
- Due Fri 9/16, 8 pm

HW3

- Out this evening
- Due Saturday 9/17, 8 pm
- Points will be deducted for style going forward

## Quiz

Week 3 material

- Tue 9/20, in lecture

# Announcements

## CMU Course Add Deadline

- Today

## 15-112 → 15-110

- Friday

## Participation scores from polls

- Week 1, just practice, doesn't count
- Week 2 totals will be up on Autolab tonight (won't include today)
- Piazza post coming with more details
- Don't panic!

# Strings

# Poll 1

What does this print?

A. A

B. B

C. C

D. D

E. E

F. F

G. G

H. None of the above

```
def ct(s):
    n = ord(s)
    n += 2
    return chr(n)

print(ct('C'))
```

# Poll 2

## Which is better?

A)

```
# Given string s
for i in range(len(s)):
    # Do stuff
```

B)

```
# Given string s
for c in s:
    # Do stuff
```

# Poll 3

What does this code print?

A. abcde
B. edcba
C. bcdea
D. bcda
E. ba
F. ab
G. (Python crashes)
H. I have no idea

```
def ct(s):
    return s[1:-1] + s[0]

print(ct('abcde’))
```

# String indexing and slicing

Reference slide

## Indexing

```
c = s[index] # c will be character at position index
```

## Valid indices

- Positive: 0 to len(s)-1  (but not len(s))
- Negative: -len(s) to -1

## Slicing

```
s[start:end:step]
```

## Similar to range arguments

- Doesn’t include end
- There are default values if any of these are left blank
- (Gets a bit goofy with a negative step)

# Poll 4

What does this function do?

```
def mystery(s):
    return s[::-1]
```

A. Return a copy of s
B. Return the reverse of s
C. Return string that is only the last character of s
D. Return string that is only the first character of s
E. Return None
F. (Python crashes)
G. I have no idea

# Example: isPalindrome(s)

# Poll 5

What does this print?

A. dog
B. DOG
C. mog
D. MOG
E. mOG
F. (Python crashes)
G. I have no idea

```
s = 'dog'
s.upper()
s[0] = 'm'
print(s)
```

# Strings are immutable

Reference slide

Once a string object is created, we can't change it.

This is what we call "immutable"

Actually, everything we have used so far is immutable: ints, floats, etc. (they just aren't very interesting objects)

It might see as though you can change strings but we can't. It always ends up as some new string object.

This will be much more relevant once we get to our first mutable object type, lists!

# Poll 6

What does this print?

A. lil
B. nasx
C. lilnasx
D. (Python crashes)
E. I have no idea

```
s = 'lil'
t = s
s += 'nasx'

print(t)
```

# Strings and aliases

Two variables are “aliases” are when they reference the exact same object.

This happens when you assign a variable to another variable:

```
s = 'abc'
t = s
```

s and t are aliases referencing the same to the same exact string object 'abc’

But…strings are immutable. We can’t possibly change s without making a new string.

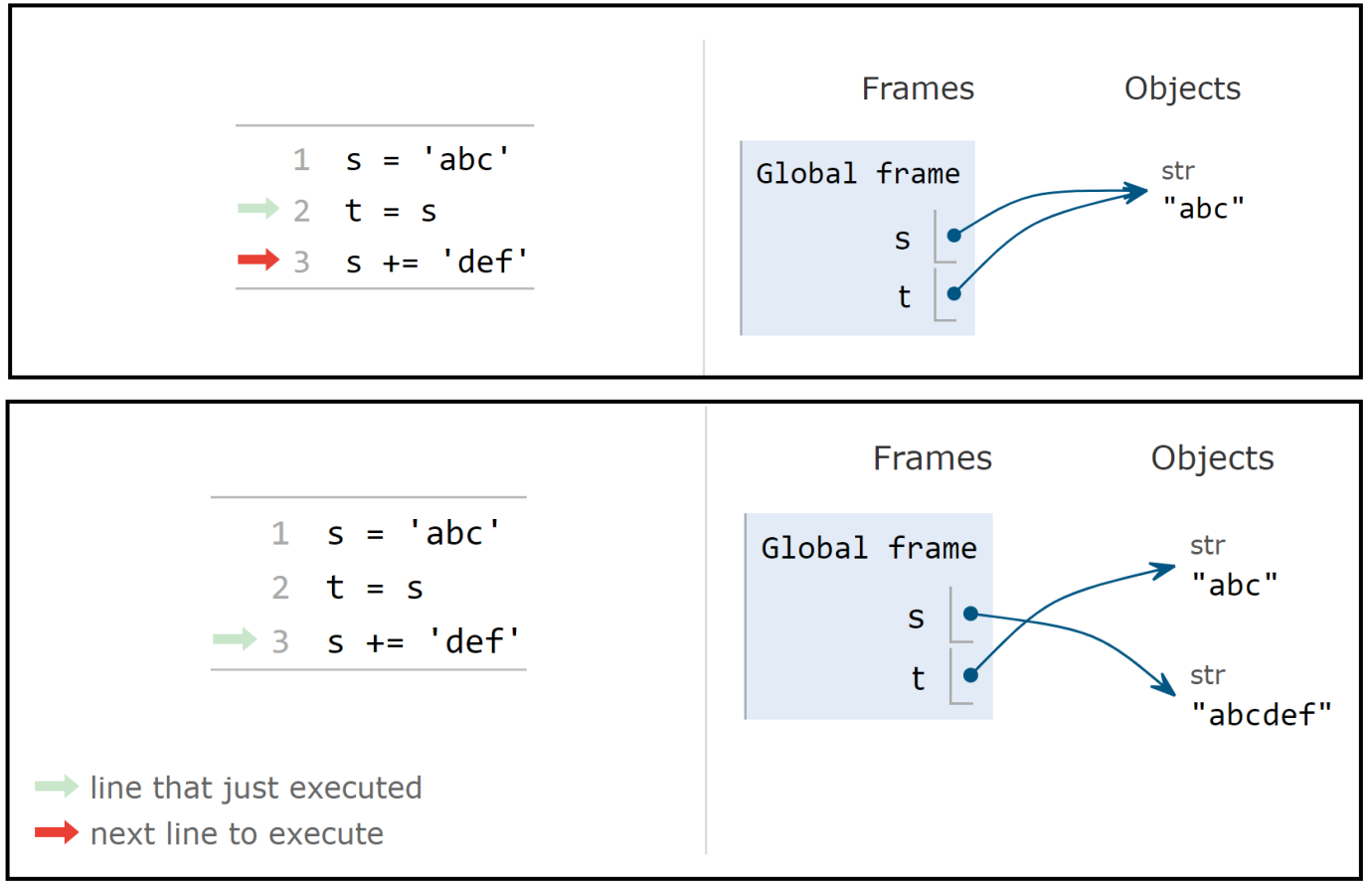
```
s += 'def' # Assigns s to a new string 'abcdef’
# The string t is referencing remains 'abc'
```

# Strings and aliases

Two variables are “aliases” are when they reference the exact same object.

This happens when you assign a variable to another variable

But…strings are immutable. We can’t possibly change s without making a new string.



# String methods

Reference slide

## From notes:

| s | isalnum | isalpha | isdigit | islower | isspace | isupper |
|---|---|---|---|---|---|---|
| ABCD | True | True | False | False | False | True |
| ABcd | True | True | False | False | False | False |
| abcd | True | True | False | True | False | False |
| ab12 | True | False | False | True | False | False |
| 1234 | True | False | True | False | False | False |
|  | False | False | False | False | True | False |
| AB?! | False | False | False | False | False | True |