Get to know each other!

What’s one hobby you’ve never tried, but always wanted to?
Announcements

Your first full prep!

If you just joined the course and missed Lab 1 and/or Prep 2, email csc148-2018-09@cs.toronto.edu.

Please double-check lab room assignments on Wednesday.
Object-oriented programming

CSC148, INTRODUCTION TO COMPUTER SCIENCE
DAVID LIU
The fundamental transition

The study of computer science transforms us from being users of technology to being creators of technology.
Key terms and phrases

- class
- instance (of a class)
- instance attribute
- method
- initializer
- dot notation
- id, type, value
- parameter
- local variable
- alias
Rebinding self

def mutate(self, x):
    self = NewObject(x)

Rebinding self doesn’t mutate anything!
Demo: Code analysis in action
Learning Tips: What to do after lecture

**Review:** summarize, question, re-explain

**Share:** meet with a friend or study group

**Get help:** come to office hours!
Composition of classes

HOW DO WE DESIGN CODE IN WHICH DIFFERENT CLASSES INTERACT WITH EACH OTHER?
Users are objects too

A Twitter user has an id, a biography, and tweets (among other things).
Learning Tips: What to do after lecture

**Review**: summarize, question, re-explain

**Share**: meet with a friend or study group

**Get help**: come to office hours!
The start of a `User` class

```python
class User:
    """A Twitter user."

    === Attributes ===
    userid: the userid of this Twitter user.
    bio: the bio of this Twitter user.
    tweets: a list of the tweets that this user has made.
    """

    userid: str
    bio: str
    tweets: List[Tweet]
```
User and Tweet

Composition

a relationship between two classes where instances of one class contain references to instances of the other

“has” relationship, e.g. “user has tweets”
Representation invariants

How do we document properties that must be true for every instance of a given class?
Tweets can have at most 280 characters

Every instance attribute has a *type annotation*, which restricts the kind of value this attribute can have.

But we often want to restrict attributes values even further; what do we call these restrictions, and how do we communicate them?
A representation invariant is a property of the instance attributes that every instance of a class must satisfy.

Example
- (in words) This tweet’s content is at most 280 characters.
- (in code) `len(self.content) <= 280`