CSCI 136:
Data Structures
and
Advanced Programming

Lecture 4

Exceptions and classes

Instructor: Kelly Shaw

Williams

Topics

- Study tip: growth mindset
- Exceptions
- Classes and objects

Study tip #1: growth mindset

Have you ever thought:
"I'm not good at [x]"

Study tip #1: growth mindset

If you are motivated and study effectively,
there is nothing you cannot learn.

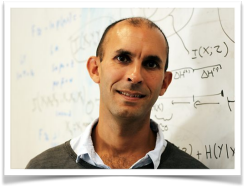
In fact, **you learn whether you want to or not**.

Proof (demo). Again. Ungarbled. One more time.

Notice that you can understand the garbled sound!

Study tip #1: growth mindset

Every brain is an **amazing learning machine**.



Anil Seth,
Professor of Cognitive and Computational Neuroscience, University of Sussex

Your brain is capable of rewiring itself in **milliseconds**.

Learning how to use your brain is a **skill** that requires **practice**!

Your to-dos

1. Lab 1, **due Tuesday 9/20 by 10pm**.
2. Quiz on Fri/Sat.
   Material: nuts and bolts discussed in class this week (command line arguments, `Scanner`, classes/objects)
3. Read **before Mon**: Bailey, Ch 3-3.1 & Ch 4-4.2.2.
   Suggestion: read *actively*.

Announcements

- CS Colloquium **today @ 2:35pm in Wege Auditorium (TCL 123)**

Ina Fiterau Brostean (UMass Amherst)

Machine Learning for Healthcare

Fiterau's research lies at the intersection of machine learning and healthcare. Her Information Fusion Lab is currently working on a project combining features extracted from brain MRIs with patient demographics, test results, and contextual information, to detect Alzheimer's disease earlier than traditional diagnostics can.

Nim

- Game starts with **random** piles.
- Each player removes **one or more** objects from **ONE** pile.
- The last player to remove the **last object loses**.

Initializing board randomly

Reading input: Scanner

Exceptions

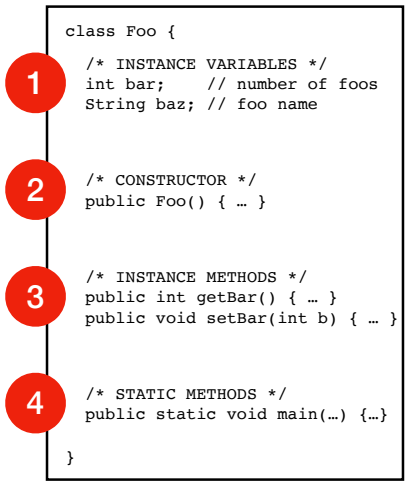
Exceptions

A **software exception** is a mechanism **for signaling errors**. When an exception is **thrown** in a program, the program will cease running ("crash") unless the program **catches** and **handles** the error.

We will talk more about how this mechanism works when we discuss the **call stack** in the near future.

Example in Nim using Scanner

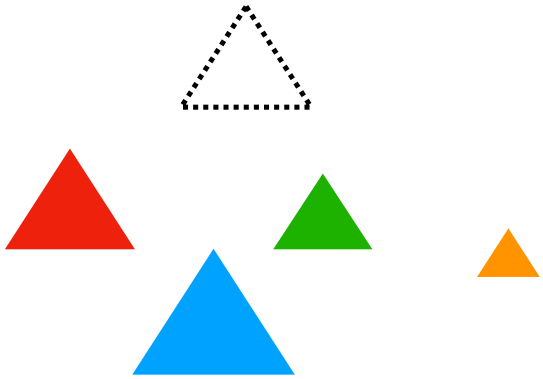
How I organize a class



Classes and objects

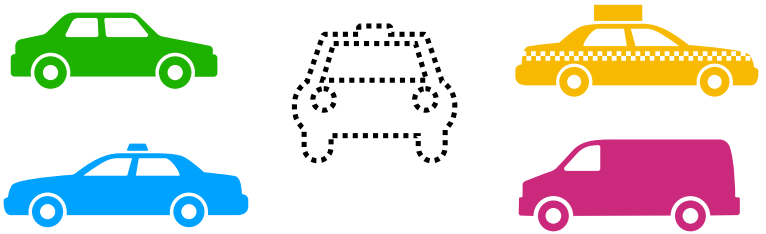
Classes are prototypes.

Objects are copies (“instances”).



“Car” is a prototype.

There are many instances of cars.

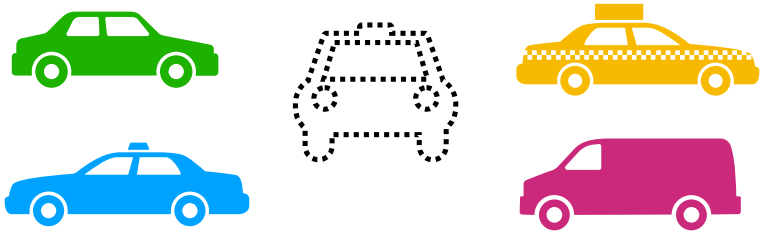


All cars have the same interface.

(wheels, doors, steering wheel, etc.)

“Car” is a prototype.

There are many instances of cars.



But most cars vary in the details

(wheels, doors, steering wheel, etc.)

Methods are functions that are tied to either:

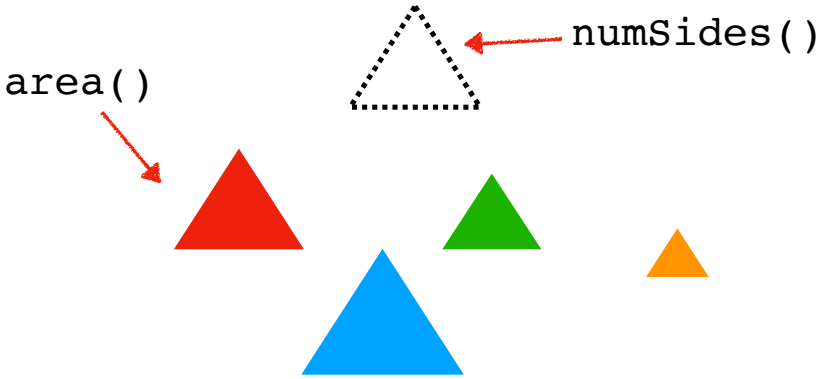
1. a class, or
2. an instance of a class (an object).

static method

instance method

`static` methods are “attached” to class.

instance methods are “attached” to object.



`static` methods are “attached” to class.

instance methods are “attached” to object.