

# 327: Object-oriented programming

Lecture 9

9/29/2021

Professor Barron

# Today

- Callback functions
- Event-driven GUI framework
- Database ORM

# Functions as objects

- Functions can be passed around, modified, called later, partially applied
- Any object can act like a function by making it callable
- `foo(arg1, arg2)`
  - `foo.__call__(arg1, arg2)`
- Methods are attributes and can be changed during runtime
  - “Monkey patching”
  - Useful for testing, but should be avoided most of the time

# First order functions in event-driven frameworks

- Event-driven programming
- Flow of the program comes from user interaction (or other form of input)
  - Unfamiliar feeling for many inexperienced programmers
- Almost all GUIs are event-driven
- One main loop
- Listens for events
- Triggers **callback functions**
- OOP is a nice complement when building with a Widget toolkit

# tkinter

- There are dozens of commonly used widget toolkits/GUI frameworks
- tkinter is the main one in python
  - comes built-in
- OOP interface
- Attach callback functions to objects representing elements in a window
- Widget objects can stay put, move around, be deleted
- Drawing each frame is abstracted away
- Example
  - Building on Notebook (and Bank) without changing existing code

# Object Relational Mapper

- Useful technique in object-oriented languages
- Translate high-level objects into database tables/queries/transactions
- Storage model can be mixed with functioning model!
- Python has SQLAlchemy and DjangoORM
- Instead of manually saving and pickling objects, let's store them in a database and synchronize it as we use the app