

Object-Oriented Design and Analysis

Lecture 07 Summary

- What we know so far
- Objects
 - Encapsulation
 - Object Relationships
 - Object Model
 - Object-Oriented Analysis
- Classes
 - Class Model

By the end of this lecture, you will be able to analyse a problem by breaking it down into *objects*.

You will also be able to identify *classes* of objects.

What is analysis (in any context)?

a·nal·y·sis [*uh*-nal-*uh*-sis]

–noun, plural -ses [-seez].

1. the separating of any material or abstract entity into its **constituent elements** (opposed to SYNTHESIS).
2. this process as a method of studying the nature of something or of determining its **essential features** and their **relations**: *the grammatical analysis of a sentence.*
3. a presentation, usually in writing, of the results of this process: *The paper published an analysis of the political situation.*
4. a philosophical method of exhibiting **complex concepts** or propositions as **compounds** or **functions of more basic ones**.

...

Source: Dictionary.com

Procedural Analysis

- Information
- Processes (procedures)

What are the *constructs* of a process?

What is the advantage of writing functions or procedures?

What are the advantages/disadvantages of C/Python?

Modularization

- Helps to hide unnecessary details
 - e.g., `#include <math.h>`
- Can be *recursive*
 - Modules can contain modules
- Abstract Data Types are a form of module

Procedural vs. Object-Oriented

- Procedural Decomposition/Analysis
 - variables (information)
 - functions (processes)
- Object-Oriented Analysis
 - objects
 - object relationships

What is the advantage of hiding
implementation details?

Encapsulation

- Objects are said to *encapsulate* their implementation details
- To use an object, you don't need to know the details of how to manipulate its state
- E.g., inserting elements into a list in Python

Example: Tag Cloud

<http://www.wordle.net>



What object relationships exist in our *object model*?

Aggregation

- “*Has-a*” relationship between objects
- E.g.,
 - A location *has a* latitude
 - A location *has a* longitude

What are the aggregate relationships in our object model?

Classes

Classify the following words

- absent
- invent
- lapdog
- lasted
- mascot
- napkin
- rented
- rested
- sunlit
- sunset
- suntan
- zigzag
- insect
- sudden

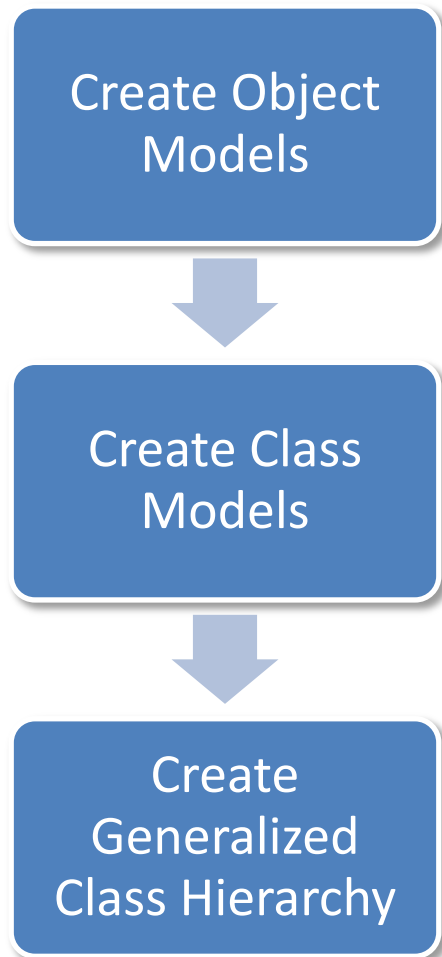
Linnaean Classification

- Life
 - Domain
 - Kingdom
 - Phylum
 - Class
 - Order
 - Family
 - Genus
 - Species
- Hierarchical
 - Most general to most specific
 - Called *generalization*
 - Use *inheritance* to create generalized classifications
 - But, not until later in the course

Linnaean Classification

- Method 1:
 - A biologist comes up with a category
 - Goes out and tries to find an instance of that new category.
- Method 2:
 - A biologist observes a form of life that may not exist within the classification system.
 - Once clear that the life form is not within the system, creates a new classification and adds it to the system.

Object-Oriented Design Process



- Based on observations
 - of entities & relationships
 - within the problem space
- Based on commonalities
 - similar objects belong to the same class
- Based on commonalities
 - within class model

What would the class model be for our tag cloud example?

Lecture 07 Summary

- What we know so far
- Objects
 - Encapsulation
 - Object Relationships
 - Object Model
 - Object-Oriented Analysis
- Classes
 - Class Model

Next Class

- Object-Oriented Design & Implementation
- Creating Classes in Java
 - i.e., implementing the class model