## **INFSCI 0201 – Intermediate Programming with Python (Spring 2023)**

## **Lab 4 - Object Oriented Programming**

Follow the tutorial or do the micro project! [Lab Guide](https://docs.google.com/document/u/0/d/1K2Yge7XyILhpik_1uDGE9VqQIQfPvbARAKpdJsRsqls/edit)

## Topics Reviewed

1. Class
2. Object

## **Lab Report**

\*) Copy this doc; fill this; and post the link in Canvas

Student 1: Kendall Foy

*\*) Both students can submit the same report to Canvas*

GitHub project folder Url: \_\_\_\_\_

*\*) For all labs, you can use monorepo or a single project in GitHub. Submit the folder, e.g., https://github.com/numpy/numpy/tree/main/numpy*

Total Notes: \_\_\_\_\_

| **Checkpoints** | **Notes \*)** |
| --- | --- |
| [Python Object Oriented Programming (OOP) - For Beginners](https://www.youtube.com/watch?v=JeznW_7DlB0) [What is an Object 0:38](https://www.youtube.com/watch?v=JeznW_7DlB0&t=38s)[Methods 4:17](https://www.youtube.com/watch?v=JeznW_7DlB0&t=257s)[Creating a Class 5:46](https://www.youtube.com/watch?v=JeznW_7DlB0&t=346s)[Defining Methods 8:48](https://www.youtube.com/watch?v=JeznW_7DlB0&t=528s)[Attributes 13:01](https://www.youtube.com/watch?v=JeznW_7DlB0&t=781s)[Modifying Attributes 15:27](https://www.youtube.com/watch?v=JeznW_7DlB0&t=927s)[Multiple Classes 19:24](https://www.youtube.com/watch?v=JeznW_7DlB0&t=1164s)[Inheritance 28:10](https://www.youtube.com/watch?v=JeznW_7DlB0&t=1690s)[Create another class 34:40](https://www.youtube.com/watch?v=JeznW_7DlB0&t=2080s)[Class attributes 40:50](https://www.youtube.com/watch?v=JeznW_7DlB0&t=2450s)[Static methods 48:50](https://www.youtube.com/watch?v=JeznW_7DlB0&t=2930s) | * # whats the point of this attribute? * # its stored permenatly for each specific object * # self. can be referenced later on from methods in our class * # once you create a class you can have an infinite number of instances of that class without changing anything * # child class * # inherits what’s inside of the parent class (super class * Can pass whatever is in parent   + Ex. \_\_init\_\_ |

Alternative resource:

[Python Object Oriented Programming (OOP) - For Beginners](https://www.youtube.com/watch?v=JeznW_7DlB0)

\*) Taking **notes**

1. Resources you use (copy-paste the URL or quote)
2. Questions, for example
   1. Do I understand this correctly? (Re-explain ideas or concepts in your own words and ask your peer/instructor)
   2. Which part does not make sense to you
      * What is happening here?
      * Why are we doing it this way?
   3. BEING EFFECTIVE: Why are we doing this? Connect this with concepts you learn from other courses, for example, “project management”.
      * The point is not getting the most accurate/precise answer but having the discussion.
   4. BEING EFFICIENT: Is there a faster or easier way of doing this?
      * Usually, students are juggling between using the convenience of IDE or the memorized command in the terminal.
      * Knowing the available options is a good thing.