## Program 1: Abstraction

This program will track Youtube videos and their comments. There will be a video class which will have an attribute for its title, author, length, and list of comments. There will also be a comment class which will have an attribute for the name of the commentor and for the text of the comment. The videos will be placed into a list and the video along with its comments will be displayed.

* Video Class
  + Attributes
    - \_title: string
    - \_author: string
    - \_length: int
    - \_comments: List<Comment>
  + Behaviors
    - Add a comment
    - Get comments
* Comment Class
  + Attributes
    - \_commenter: string
    - \_text: string
  + Behaviors
    - Hold the information about the comment

[Program 1 PlantUML Diagram](https://www.plantuml.com/plantuml/png/NO_12i8m44Jl-OezjgW_44IK7hnu4xnAQPSsa2RGJ44G_ztItKZjSIQJDpldX7h4D7ZJ-3ebUhYM8xqDqO6USF1iAM5qeHEhdj37yT_p73hqbbo064qS1Wv8bcukuLY9F9dvTISDXQBl2sPF6Plr2baklovjgvX2wPRKaC2LeRgAqvmgvPwFqM7wcASfX3TJm2_yd8MxQUNLaH8MLSvDNm00)

A screenshot of a computer

Description automatically generated

## **Program 2: Encapsulation**

This program will have classes for Products, Customers, Addresses, and orders. Getters and Setters will be used to pull information from each of these classes to create shipping labels.

* Order Class
  + Attributes
    - \_products: List<Product>
    - \_customer: Customer
  + Behaviors
    - Add a product to the order
    - Calculate the total cost of the order
    - Get the packing label
    - Get the shipping label
* Product Class
  + Attributes
    - \_name: string
    - \_product: string
    - \_pricePerUnit: double
    - \_quantity: int
  + Behaviors
    - Get the total cost of the product
* Customer Class
  + Attributes
    - \_name: string
    - \_address: Address
  + Behaviors
    - Contain the information about the customer
* Address Class
  + Attributes
    - \_street: string
    - \_city: string
    - \_stateOrProvince: string
    - \_country: string
  + Behaviors
    - Contain the information about the address
    - Check whether it is in the USA or not

[Program 2 PlantUML Diagram](https://www.plantuml.com/plantuml/png/TLFHQWCX47o_h_3nqYO_S9II46W917AGvhaONLgfqLJNGYdzzzxTkiXThc-wCyxihBgEA0EcYwsKbJ6AGz0Gn4ybn4gyNeFNIM5inDv4V6Xf-qYWIX7z1K8ZjdbLTS0zATItw690DrfdcJghDo8NYB2LLYKh4LuyIhlr4Uj58xHFPmi3uHcmbUh3kBUzF8FjqOYXsp9wV3VNwmpyMr56zkEKJbw0IQFSEpqj6mKjX9Cpo2qHyfcaGuFVZJ0EQGoSajIFWynI5ELIcb5TYf7YWgFDpAJ4ua7_dqjg7I1srxcX1NNAvoQjJidKn2xktEcuwVtFtjlYdbbitja2uDXUzNboXLaIuH3QuB-CKp2X--Gmb1EzFzlKhC_TaZRlRdGPo9fSuC7YKxAMfKSlXlx3tMh55regvTtJ8bTvjcjmklzNVm00)

A screenshot of a computer

Description automatically generated

## Program 3: Inheritance

This program will have a base class for Events, and will have inherited classes like lectures, receptions, and outdoor gatherings. The program will handle the common event details and the individual details for specific types of events to create marketing messages.

* Event Class
  + Attributes
    - \_title: string
    - \_description: string
    - \_date: string
    - \_time: string
    - \_address: Address
  + Behaviors
    - Get the standard details
    - Get the full details
    - Get the short description
* Lecture Class
  + Attributes
    - \_speaker: string
    - \_capacity: int
  + Behaviors
    - Hold the details for the event
* Reception Class
  + Attributes
    - \_rsvpEmail: string
  + Behaviors
    - Hold the details for the event
* OutdoorGathering Class
  + Attributes
    - \_weatherForecast: string
  + Behaviors
    - Hold the details for the event
* Address Class
  + Attributes
    - \_street: string
    - \_city: string
    - \_stateOrProvince: string
    - \_country: string
  + Behaviors
    - Hold the details about the address

[Program 3 PlantUML Diagram](https://www.plantuml.com/plantuml/png/jPJFIWCn4CRlVOeSMsrVe7Yeq3yNeM8VG8PaiC5jiancAwA-kzlDJjWEAWeMUcY--JBV_7QpMIG6ukPGZMm5APdL4GERjv4nC_F8dYkScyJamrEM72PBlcOVWoe0AolxWrB0ESAKvkOs_sdLzdUTGyTzsgxRC3N3f2Bc55db15dfxfEkzmPvnn0Sa5iYWw_IU29JYM7TLDMtnTq-4Y-n33DmV8pwftQ7bXj2UMwfHdX6EeUtK8Fr_3etFh2Gznl_dNrgoWnYESkV_8nUm1xGOZU6e54wrgj3QnUla1Jd1LXAg9X-E_wsOHSZRO3tUAe8nGjsoZeIMaYiMVIs2o2f2Vu89ds4foqZiZflxRiMHLp0kALxYaSVB2fxR0BJOCTf4iaPIu00vUQoqerBGPf-HPTRAxfy_ToypsRoQGobSiQ6edvPVUsgBSariy3WJbVS9m00)

A close-up of a computer screen

Description automatically generated

## Program 4: Polymorphism

This program will have a base Activity class, and will have inherited classes for Running, Biking, and Swimming. The program will list the different activities and show calculations and other details about the activities. Each derived class will override the methods for calculating the distance, speed, and pace.

* Activity Class
  + Attributes
    - \_date: string
    - \_lengthinMinutes: int
  + Behaviors
    - Get the distance
    - Get the speed
    - Get the pace ()
    - Get the summary of info about the activity.
* Running Class
  + Attributes
    - \_distance: double
  + Behaviors
    - Get the distance (override)
    - Get the speed (override)
    - Get the pace (override)
    - Get the summary of info about the activity (override)
* Cycling Class
  + Attributes
    - \_speed: double
  + Behaviors
    - Get the distance (override)
    - Get the speed (override)
    - Get the pace (override)
    - Get the summary of info about the activity (override)
* Swimming Class
  + Attributes
    - \_laps: int
  + Behaviors
    - Get the distance (override)
    - Get the speed (override)
    - Get the pace (override)
    - Get the summary of info about the activity (override)

[Program 4 PlantUML Diagram](//www.plantuml.com/plantuml/png/lPD1QiCm44NtFiKiRHfVm7IHqaBfeb3g0vI9FAG2QHgiKOj9S_TAZIAI81Al0bwD_lleOStI2OxYhQcKGUVWGOd-rZB1jW9euMD0eGwSZ9hN-uaXNild2xzgzaAk0yqIJi9tb-bwZ_N_s09AI1E0BQv2CohP1VQPv4axGLPKDnqCNtvbg13gDqJ3nSGRNglmrk8urKqssrN9_zqpXq7MJpVAPL7p49fXkO3pXk1zGRPiMDQwwl8uANFaucBjYKXAp1D9-Cqj-XzjxP66mKrwT15EnpCTCdkJ--TTklzjsyEZEPkcFt0wpLBLadY8o_a7)

A screenshot of a computer

Description automatically generated