# Week 2 Class Task 1

Create a .doc file and write two real-life and IT world examples of (Inheritance, Polymorphism, Abstraction, and Encapsulation) (Theoretical)

**Student**: Diego Ortiz - c0816681

## Inheritance

1. We can have a **Contact** class with Phone, Address attributes
   1. **A person** who inherits from Contact adding a first name and last name attributes.
   2. **A company** that inherits from Contact adds Company Name, business website attributes.
2. We can have an **Animal** class with age, color, size
   1. **Mammals**: Have legs and arms, fur
   2. **Birds**: Have wings and legs, feathers

## Polymorphism

1. We can have a **BusinessContact** class that exposes the **Act**() behavior
   1. **Customer** child class: Can implement Act() by purchasing
   2. **Vendor** child class: Can implement Act() by selling
2. We can have an **Animal** class that exposes the **Move**() behavior
   1. **Mammals** child class: Can implement Move() by walking
   2. **Birds** child class: Can implement Move() by flying.

## Abstraction

1. We can have an **Order** class that exposes the behavior **Confirm.**
   1. The Confirm behavior does many sub-tasks that are transparent for the user of the class.
   2. It confirms, sends an email, notifies the dispatcher, notifies the delivery service.
   3. But the user only needs to know the order has been confirmed.
2. We can have an Animal class that exposes the behavior Eat()
   1. The animal requires a mouth, teeth, stomach that work in conjunction to complete.
   2. But the user only needs to know the animal is eating

## Encapsulation

1. We can have an **Order** class that exposes the behavior **Confirm.**
   1. To complete his behavior, the order has internal details on how to send emails, which vendor to notify, etc. Those details are encapsulated.
2. We can have an Animal class that exposes the behavior Eat()
   1. To eat the animal requires to have internal organs, and features that are internal details encapsulated.