Mike Taatgen
DPW
Final

1. When a class inherits from another class it gets it attributes that were written inside the superclass.
2. Polymorphism is when you have something from the superclass written in your sub class and it will overwrite that value but with the same functionname

```
Class Parent(object):
    Def __init__(self):
      Def talk:
          Print "I can talk normally"

    Class Child(object):
     Def __init__(self):
      Super(Child,self).__init__()
      Def talk:
          Print "Weeew Weeew, I talk like a baby"
```

3.
4. The difference between encapsulation en inheritance is that inheritance grabs the attributes and uses the new value for it while encapsulation only gives the attribute that value on line where you asked for it.
5. Encapsulation makes it so that you can't overwrite information getters and setters aid encapsulation in this process since it's a form of security for the value
6. An abstract class is a class that were you declare all your values but with empty information or 0. You're basically going to give all it's attributes with real values

```
class AbstractRapper(object):
 def __init__(self)
    self.__song = [] #Array with all his songs
    self.__nickname = "" #Nickname for the Rapper
```

7. The M in MVC stands for Model, This sets up all the information for your view. The view is set to display the page and the controller is set for the connection between these 2.