CECS 277 – Lecture 14 – Passing a Method as a Parameter

There are times when you may need a method to decide to call another method, this is usually accomplished by using if statements and calling the correct method. However, there may be times when you do not know the methods ahead of time, usually due to encapsulation. In these cases you would want to pass a method to the method so that it could be called at the correct time. Java does not let us directly pass a method as a parameter like some other languages do, but we can effectively do the same thing by wrapping the method inside of a class or interface.

Example: Create the method wrapper.

```
public interface Command {
     void execute( );
}
```

Example: Define the methods.

```
public class Speak implements Command {
     public void execute(){
          System.out.println(" ARF!  ARF! ");
     }
}

public class Sit implements Command {
     public void execute(){
          System.out.println(" Sits on the ground ");
     }
}
```

Example: Call the method.

```
public class Dog {
     public void executeCmd( Command cmd ){
          cmd.execute();
     }
}
```

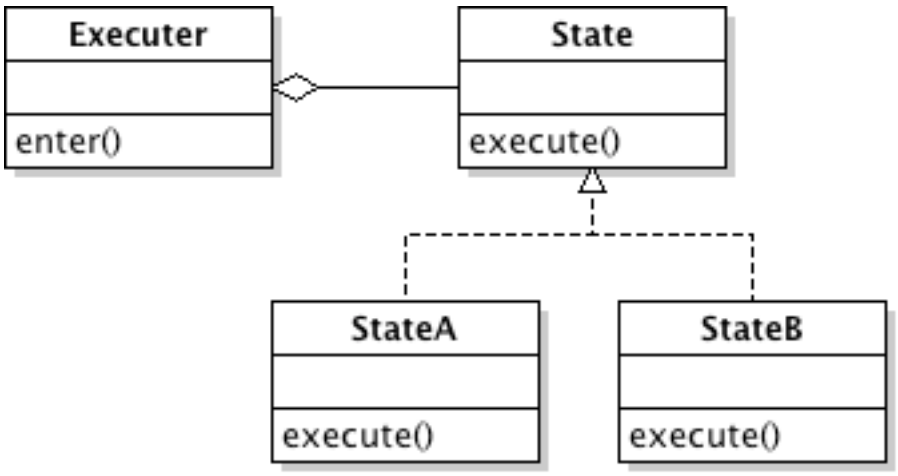
Example: Create a Dog in main and give it commands.

```
public class Trainer {
     public static void main( String [] args ){
          Dog d = new Dog();
          Command speak = new Speak();
          Command sit = new Sit();
          d.executeCmd(speak);
          d.executeCmd(sit);
     }
}
```

State Machines –

We can think of the dog as being in one of two different states, either Speaking or Sitting. This concept comes in very handy when creating a program that executes different functions depending on what state it is currently in. Alarm Clocks, Microwaves, TVs, Cars, and ATM Machines are just a few objects that rely on states.

Class Hierarchy of a State Machine:



When creating a state machine, it is often helpful to create a state transition diagram for the object. A state transition diagram allows you to enumerate the different possible states of the object along with the events that trigger the state changes.

Example: State Diagram of the Life of a Light Bulb.