6. Object Oriented Concepts and Techniques

Lesson 4:
Polymorphism

6.9. Polymorphism

- The Polymorphism is the quality of having more than one form.

- Combines Greek Words Poly and Morphism - Poly meaning Many and Morphism meaning Forms.

- In the context of Object Oriented Programming, polymorphism refers to the fact:

- A single operation can have different behavior in different objects.

6.9. Polymorphism
6.9.1. Overloading and Overriding

- It allows different forms of the same service to be defined. There are two common ways of implementing Polymorphism: Overloading and Overriding

- Overloading
  – Using the same method name with different parameter type lists

- Overriding
  – Using different implementations of the same method in sub classes.

6.10. *this* and *super* keywords
6.10.1. *this* keyword

- The this keyword is used to refer to the current object.

- It can be used to:
  - Refer to the current objects member variables
  - Refer to current objects methods
  - Pass a reference to the current object to a method
  - Return a reference current object

6.10. *this* and *super* keywords
6.10.1. *this* keyword (Cont …)

- The following refers to the member variable **x** in **this** object

    t = this.x;

- The following refers calla the **myMethod** defined in this class and pass its **this** object as a parameter

    this.myMethod(this);

- The following return the **this** Object

    return this;

6.10. *this* and *super* keywords
6.10.2. *super* keyword

- The super keyword is used to refer to the super or parent class.

- The following refers invokes super class Constructor
  super();

- The following refers invokes super class Constructor with **x** and **y** arguments
  super(x,y);

- Calls the super class function called **f()**
  super.f();