Assignment 1

Web Technologies

1. 4 principles of Object Oriented Programing

The 4 principles of Object Oriented Programming are Encapsulation, Data Abstraction, Polymorphism and Inheritance.

1. Encapsulation: Encapsulation means that the internal representation of an object is generally hidden from view outside of the object’s definition. Typically, only the object’s own methods can directly inspect or manipulate its fields.

Accessor is a method that is used to ask an Object about itself.

Mutator is a public method that is used to modify the state of an object, while hiding the implementation of exactly how the data gets modified.

Encapsulation provides reduced system complexity.

1. Data Abstraction: Encapsulation and Abstraction are tied together. Because a simple definition of data abstraction is the development of classes, objects, types in terms of their interfaces and functionality, instead of their implementation details. Abstraction means working with something we know how to use without knowing how it works internally. Abstraction is one of the most important concepts in programming and OOP. It allows us to write code, which works with abstract data structures.
2. Inheritance: It’s the fundamental principle of object oriented programming. This let a class inherits or carry the behavior from the parental class. It’s a way of reusing codes or to generate a subtype from an existing object. The classes can exhibit properties from existing classes where existing classes called as parental class, base class, ancestor class or super class. This relationship will produce hierarchies.
3. Polymorphism: Polymorphism is for the things with same name but different forms. In OOP Polymorphism means multiple methods with same name but functionality varies slightly.

There are 2 basic types of Polymorphism. The first is overriding which is also called as run time polymorphism. The second is overloading which is compile time polymorphism.

Ref: [www.introprogramming.info](http://www.introprogramming.info), www.wordpress.com

1. Benefits of 4 principles

* Software Reusability and Recycling: It’s the fundamental benefit of Inheritance. Inheritance provides extendibility, reusability, Provides abstraction and reduce redundancy.
* Extendibility: The programmer can easily make derived class from parental class by adding new variables or methods. Child class can extend parent class using Extendibility.
* Maintainability: OOP is easier to maintain and modify since the design is modular. In case of some troubles with the system, part of it can be modified without disturbing the other part.