

What is it?

- Inheritance can be thought of as software reusability where one class inherits another classes' data and functions and adds new functionality of its own
- Parts:
 - superclass the existing class
 - subclass the new class with inherited members and additional behaviors

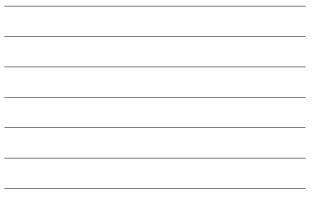
CS250 Introduction to Computer Science II

Public Inheritance

- Every derived class object is also an object of the superclass.
- As an example, if the superclass is "Vehicle" then a subclass might be "Cars" and "Trucks." Cars inherit the members and behaviors of a Vehicle and add other behaviors and members
- Members of a subclass cannot directly access the private members of a superclass

CS250 Introduction to Computer Science II

Example Class Person { private: string name; public: Person() { setName(""); } Person(string name) { setName(name); } void setName(string name) { this->name = name; } string getName() { return name; } };



Enumerated Data Types (4.13)

 Enumerated data types are programmerdefined data type that contain a set of named integer constants

enum Roster{ Bart, Maggie, Homer, Lisa, Marge };

CS250 Introduction to Computer Science II

Roster student;

student = Lisa;

Enumerations

• We are to create two enumerations to be used with the Person class as follows:

enum Discipline { MATH, BIOLOGY, COMPUTER_SCIENCE };

enum Classification { FRESHMAN, SOPHOMORE, JUNIOR, SENIOR };

CS250 Introduction to Computer Science II

Another Class	
class Student : public Person	-
£	
private:	
Discipline major;	
Person *advisor;	
public:	
<pre>void setMajor(Discipline d) { major = d; }</pre>	
<pre>Discipline getMajor() { return major; }</pre>	
<pre>void setAdvisor(Person *p) { advisor = p; }</pre>	
<pre>Person *getAdvisor() { return advisor; }</pre>	
};	
CS250 Introduction to Computer Science II	7

