



Animal:
canLie="False"
age;gender;color

Object

Calculator
add();subtract();
multiply();divide()
sin();cos()

Pet
Age;gender;
color;talk()

Base_Calculator
add();subtract();
multiply();divide()
sin();cos()

Class Inheritance

Cat
age;gender;color
talk()

Dog
age;gender;color
talk()

Regular_Calculator
add();subtract();
multiply();divide();
sin();cos()

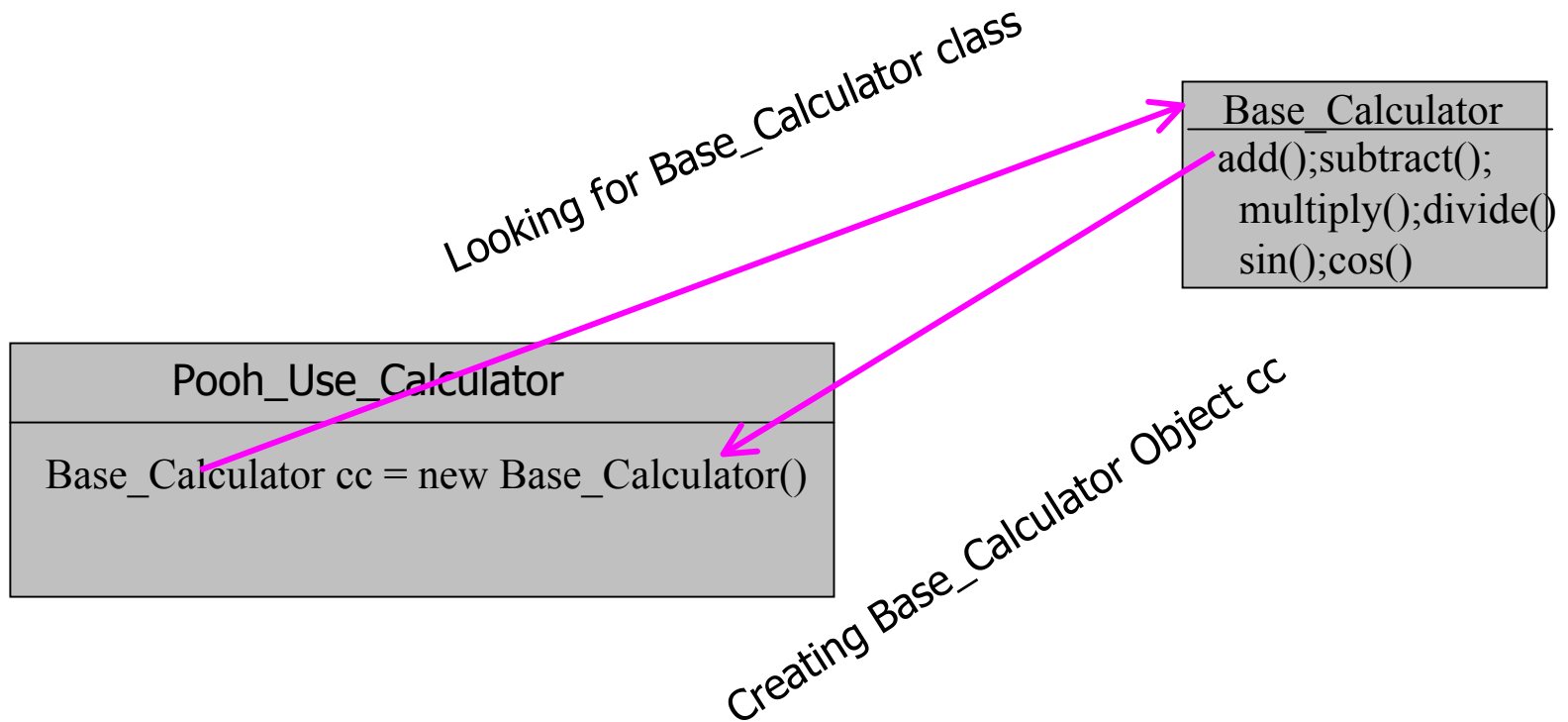
IntelligentCat
add();subtract()
talk()

IntelligentDog
add();subtract()
talk()

Powerful_Calculator
add();subtract();
multiply();divide()
sin();cos()

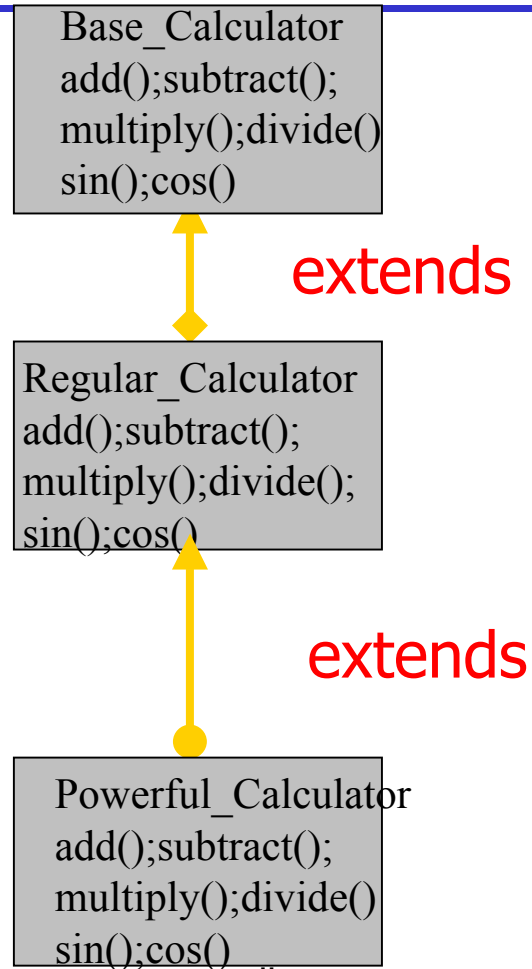


Class and Object





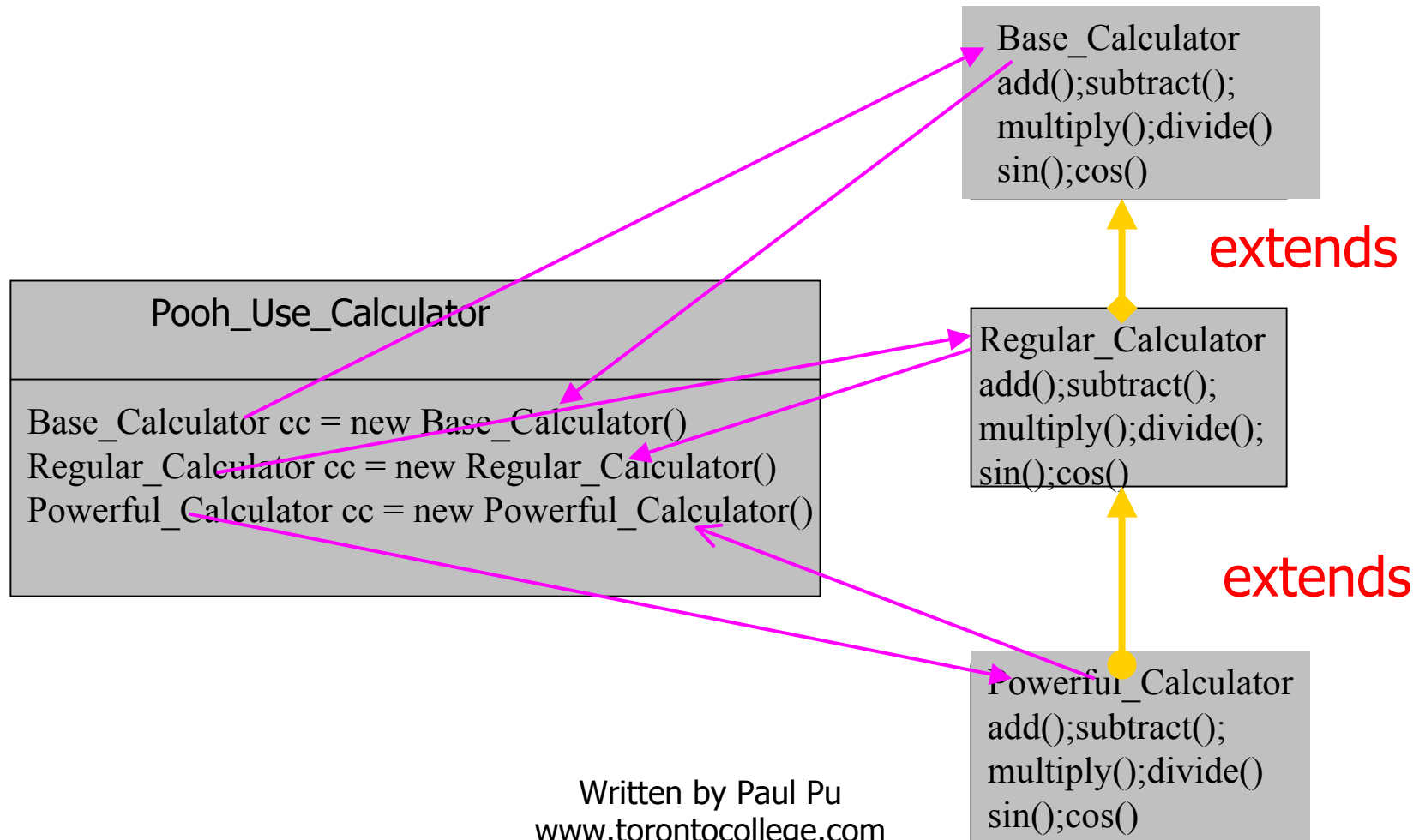
Class Inheritance





Class Inheritance

Class and Object



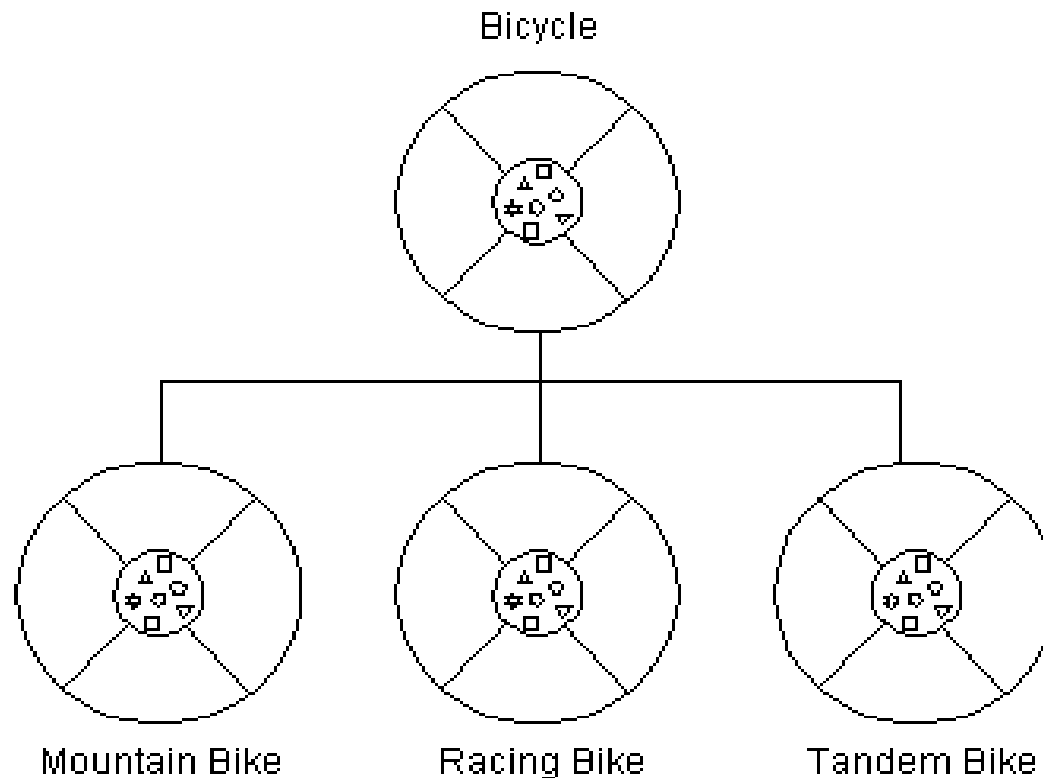


What Is Inheritance?

- **Object-oriented systems** take a step further and allow classes to be defined in terms of other classes.
- For example, mountain bikes, racing bikes, and tandems are all kinds of bicycles. In object-oriented terminology, mountain bikes, racing bikes, and tandems are all subclasses of the bicycle class. Similarly, the bicycle class is the superclass of mountain bikes, racing bikes, and tandems. This relationship is shown in the following figure.



What Is Inheritance?





What Is Inheritance?

inheritance

- The concept of classes automatically containing the variables and methods defined in their *supertypes*.



What Is Inheritance?

subclass

- A class that is derived from a particular class, perhaps with one or more classes in between.

superclass

- A class from which a particular class is derived, perhaps with one or more classes in between.

Each subclass inherits state (in the form of variable declarations) from the superclass. Mountain bikes, racing bikes, and tandems share some states:



What Is Inheritance?

Inheritance offers the following benefits:

- Subclasses provide specialized behaviors from the basis of common elements provided by the superclass. Through the use of inheritance, programmers can reuse the code in the superclass many times.
- Programmers can implement superclasses called *abstract classes* that define "generic" behaviors. The abstract superclass defines and may partially implement the behavior, but much of the class is undefined and unimplemented. Other programmers fill in the details with specialized subclasses.



What Is Inheritance?

- *hierarchy*

A classification of relationships in which each item except the top one (known as the root) is a specialized form of the item above it. Each item can have one or more items below it in the hierarchy. In the Java(TM) class hierarchy, the root is the Object class. **You are not limited to just one layer of inheritance.**