Inheritance

Last updated 4/10/19

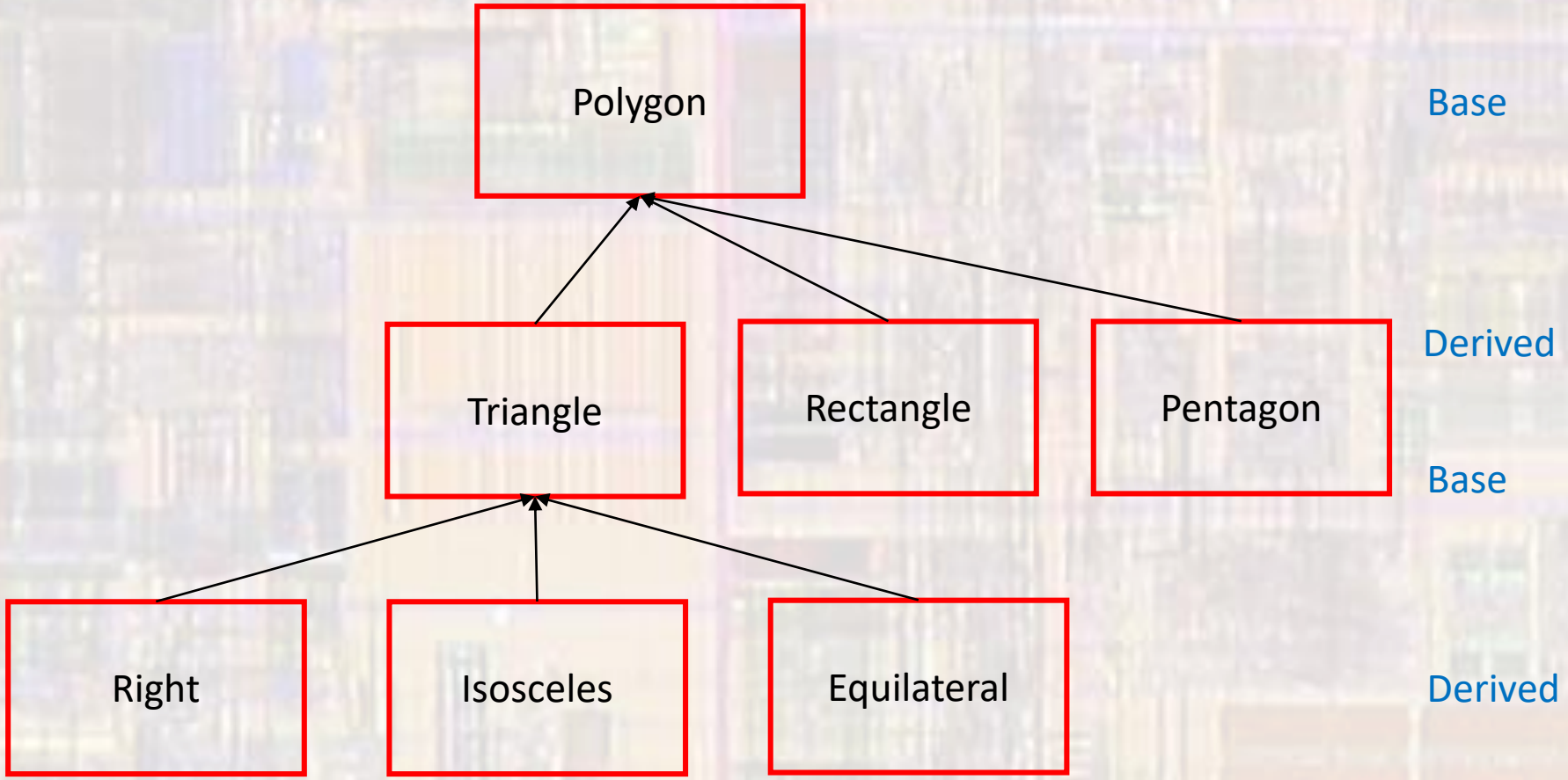
Inheritance

- Motivation
  - When creating Classes that have member variables and functions that are a subset of another Class

  two options:
  - Recreate the variables for each new Class
  - Inherit the existing Class

Inheritance

- Example



Inheritance

- Details
  - Derived classes inherit all the member variables and member functions from it's base class
    - Access is limited
  - Refer to them as if they were part of the derived class

Inheritance

- Syntax
  - In the base class –
    - Make anything you want to be inheritable either
      - private → protected (this allows inheritance)
      - public
  - In the derived class (.h file)
    - Include the base class .h file
    - Indicate the base class in the Class definition line

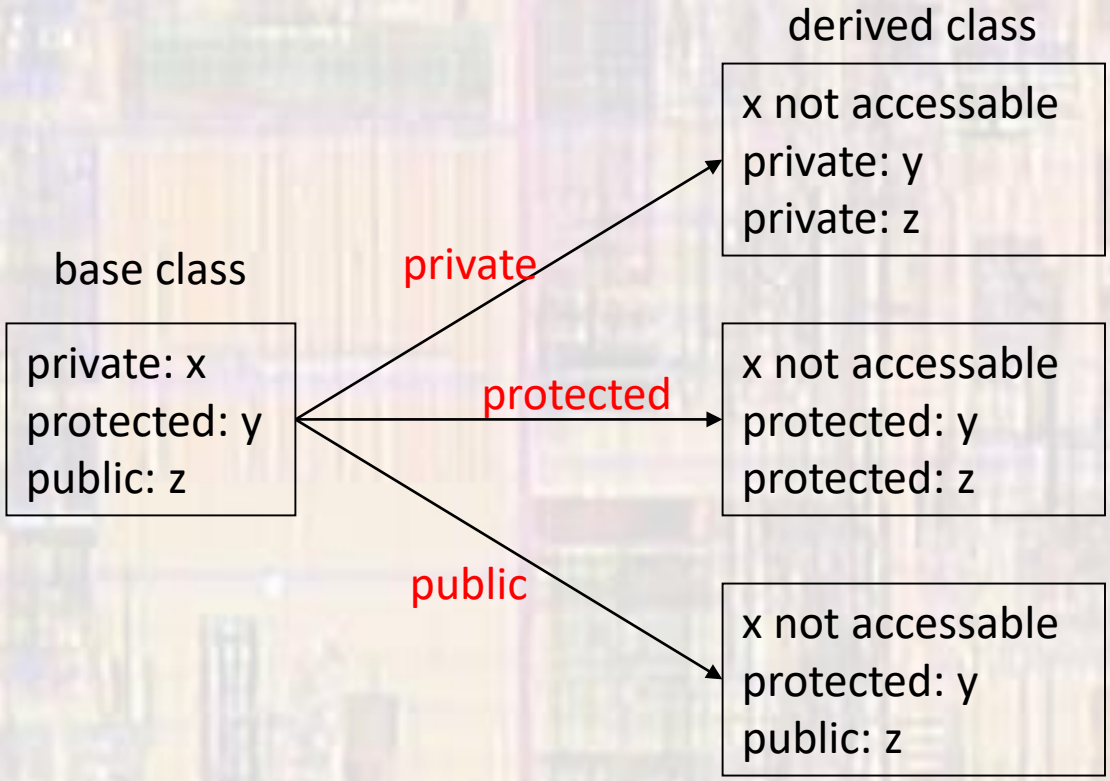
      class Rectangle : public Polygon { …
  - In the derived class (.cpp file)
    - If desired – call the base class constructor from the derived class constructor

      Rectangle::Rectangle() : Polygon() { …

Inheritance

- Syntax

  class Rectangle : public Polygon { …



Inheritance

- Syntax
  - Base class members and functions can be overwritten in a derived class

Inheritance

- Type compatibility
  - A derived class pointer can be assigned to a base class pointer
  - A type cast is required to assign a base class pointer to a derived class pointer