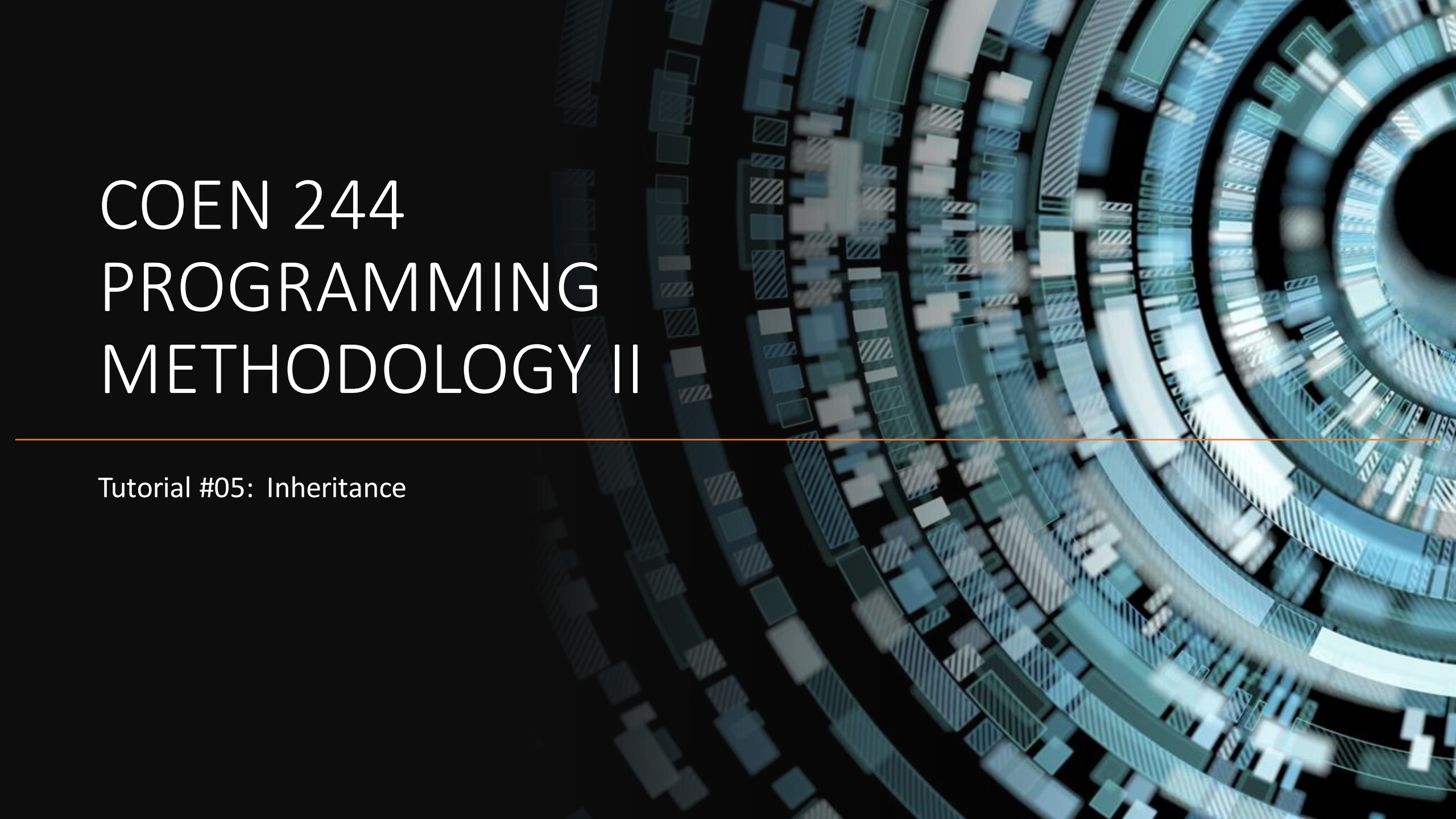
COEN 244 PROGRAMMING METHODOLOGY II

Tutorial #05: Inheritance



Exercise 1

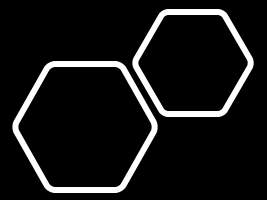
Multilevel Inheritance

- Create a class named '**FirstLevel**' with the following requirements:
  - Private attribute named 'levelID'
  - Constructor: print a message indicates the first level constructor
  - Public Setter and getter methods for the attribute 'levelID'
  - Destructor: print a message indicates the first level destructor
- Create a class named '**SecondLevel**' which inherits from the class '**FirstLevel**' with the following requirements:
  - Constructor: print a message indicates the second level constructor
  - Destructor: print a message indicates the second level destructor
- Create a class named '**ThirdLevel**' which inherits from the class '**SecondLevel**' with the following requirements:
  - Constructor: print a message indicates the third level constructor
  - Destructor: print a message indicates the third level destructor
  - Test your classes in the main method
    - Create one object from each above class (total three objects)
    - Set level ID value to 1 for the object of '**FirstLevel**' , to 2 for the object of '**SecondLevel**', to 3 for the object of '**ThirdLevel**'
    - Print the level ID for each object

Exercise 2

Friend Function

- Create class called '**Balance**' with the following requirements:
  - Private attribute named 'balance'
  - Constructor to initialize the balance with zero
  - Setter and getter methods for the balance attribute
  - Friend function named 'addSpecialAmont' that takes a reference to the balance object and amount as arguments. Add the amount to the balance of the referenced object if the amount>0, otherwise print a message indicates that can not add the amount to the balance.
- Test your class and friend function in the main method
  - Create a balance object
  - Set balance to 30$
  - Print current balance
  - Add 40$ to the balance using the friend function
  - Print the current balance (should be 70$)



Exercise 3

private vs public inheritance

3.1

- Create class called '**Parent**' with the following requirements:
  - Constructor
  - Method named printOne just to print 'one'
  - Method named printTwo just to print 'two'

- Create class called '**Child**' that inherits class 'Parent' privately with the following requirements:
  - Constructor
  - Method named printThree just to print 'three'

- Test your program in the main function as the following:
  - Create object c from class Child
  - Call printThree through c
  - Call printOne through c

3.2

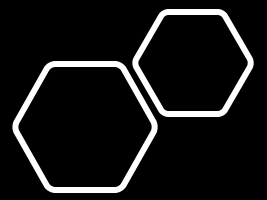
Try to fix any issue in 3.1 (**hint** type of inheritance)

Exercise 4

Practice Inheritance

- Create class named '**Employee**' with the following requirements:
  - Attributes: empID, firstName, lastName
  - Constructor
  - Setters and getters methods for all the attributes

- Create class named '**SalariedEmployee**' that inherits the '**Employee**' class with the following requirements:
  - Attributes: salary
  - Constructor
  - Setters and getters methods for all the attributes
- Create class named '**HourlyEmployee**' that inherits the '**Employee**' class with the following requirements:
  - Attributes: workingHours, hourWage
  - Constructor
  - Setters and getters methods for all the attributes
  - Add getSalary method that returns (workingHours *HourWage)



Exercise 4

Practice Inheritance

- Test the previous classes in main method as follows:
  - Create two salaried employees objects with their information
  - Create two hourly employees objects with their information
  - Print details information of all the employess
  - Find the average salary for all the employees