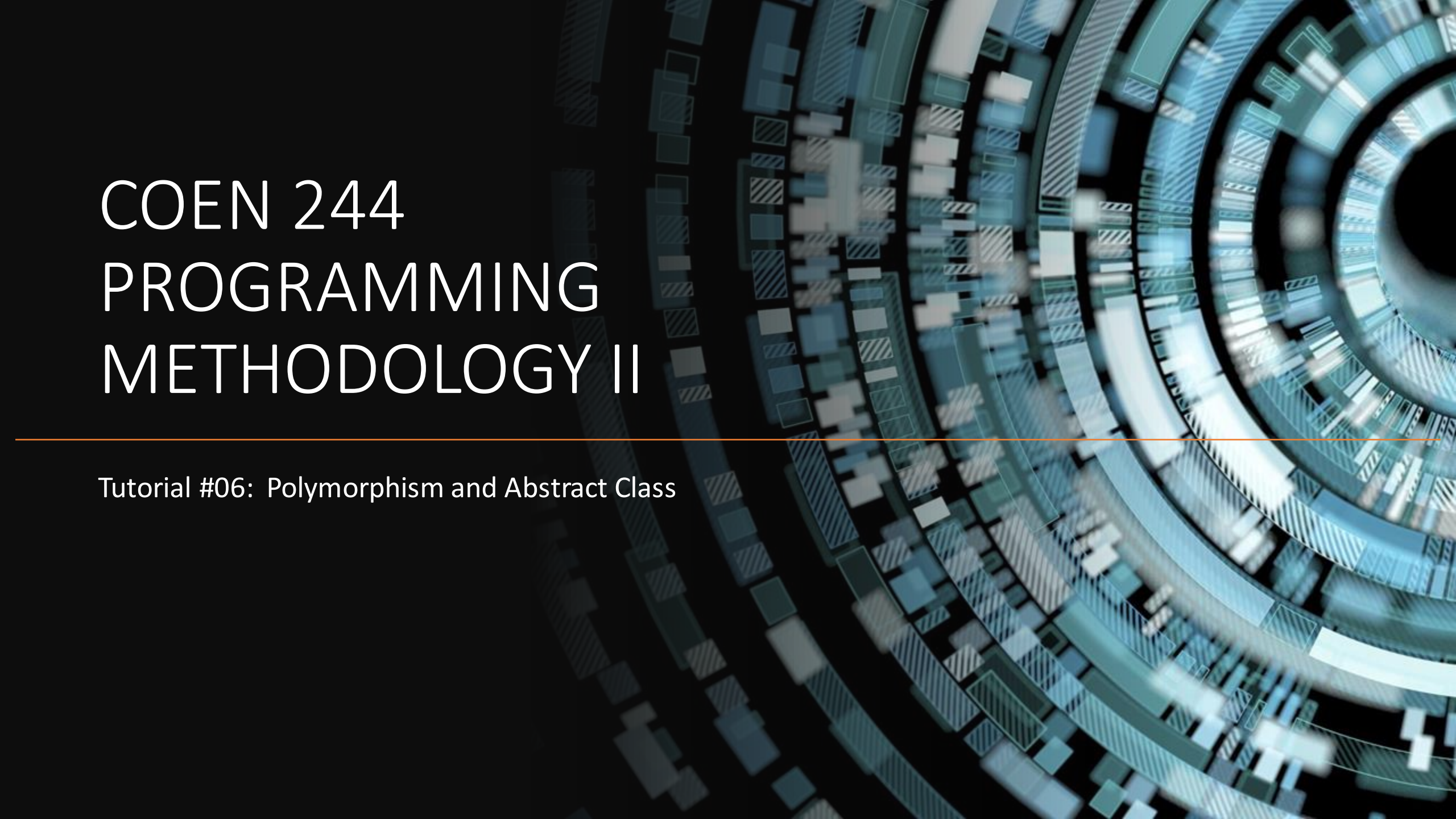
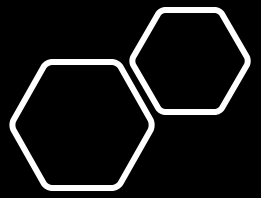
COEN 244 PROGRAMMING METHODOLOGY II

Tutorial #06: Polymorphism and Abstract Class

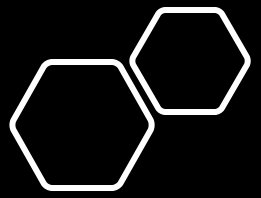




Exercise 1

Polymorphism

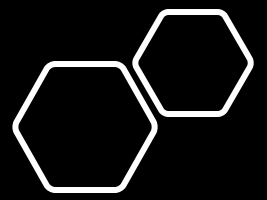
- Create a class named '**Student**' with the following requirements:
  - Private attributes named id, fName, lName
  - Constructor to initialize the attributes
  - Public Setter and getter methods for the attributes
  - Virtual method 'printInformation' to print student information
  - Destructor
- Create a class named '**UndergraduateStudent**' which inherits from the class '**Student**' with the following requirements:
  - Private attribute named status
  - Constructor to initialize the attributes including the base class
  - Public Setter and getter methods for the attributes
  - Override the method 'printInformation' to print student Information and its status (undergradtute)
  - Destructor
- Create a class named '**GraduateStudent**' which inherits from the class '**Student**' with the following requirements:
  - Private attribute named status
  - Constructor to initialize the attributes including the base class
  - Public Setter and getter methods for the attributes
  - Override the method 'printInformation' to print student Information and its status (gradtute)
  - Destructor



Exercise 1

Polymorphism

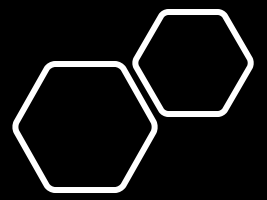
- Test your program as follows:
  - Create dynamic object from Student class
  - Create dynamic reference from Student class that reference UnderGradtuateStudent object
  - Create dynamic reference from Student class that reference GradtuateStudent object
  - Print information for each object



Exercise 2

Polymorphism

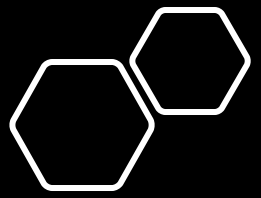
- Update the previous exercise 1 as follows:
  - Modify the **Student** class to add method named print that takes pointer to Student object and call method printInformation
  - Create dynamic object from UnderGradtuateStudent class
  - Create dynamic object from GradtuateStudent class
  - Create dynamic object from Student class
  - Use created object from the Student class to call print method and pass reference to UnderGradtuateStudent object, call print method again and pass reference to GradtuateStudent object



Exercise 3

Abstract Class

- Create a class named '**Animal**' with the following requirements:
  - Constructor
  - Virtual method named show to print 'animal'
  - Pure method named eat
- Create a class named '**Cat**' which inherits from the class '**Animal**' with the following requirements:
  - Private attributes named type and color
  - Constructor to initialize the attributes
  - Public Setter and getter methods for the attributes
  - Override method 'show' to print cat information
  - Override method eat to print 'cat likes to eat tuna fish'
- Create a class named '**Dog**' which inherits from the class '**Animal**' with the following requirements:
  - Private attributes named type and color
  - Constructor to initialize the attributes
  - Public Setter and getter methods for the attributes
  - Override method 'show' to print dog information
  - Override method eat to print 'dog likes to eat meat'
- Test your program as follows:
  - Create Cat object through reference to Animal class
  - Create Dog object through reference to Animal class
  - Print the information of the objects



Exercise 4

Polymorphism

- Create a class named '**Employee**' with the following requirements:
  - Constructor
  - Private attributes named id, fname, lname
  - Setters and getters
  - Virtual method named 'print'
- Create a class named '**FullTimeEmployee**' which inherits from the class '**Employee**' with the following requirements:
  - Private attributes named salary
  - Constructor to initialize the attributes
  - Public Setter and getter methods for the attributes
  - Override method 'print' to print full time employ information with salary
- Create a class named '**PartTimeEmployee**' which inherits from the class '**Employee**' with the following requirements:
  - Private attributes named salary
  - Constructor to initialize the attributes
  - Public Setter and getter methods for the attributes
  - Override method 'print' to print part time employ information with salary
- Test your program as follows:
  - Create FullTimeEmployee object through reference to Employee class
  - Create PartTimeEmployee object through reference to Employee class
  - Print the information of the employees