

# Class Inheritance and Polymorphism

1. **SHIFT\_TYPE enum:** Create an enum *SHIFT\_TYPE* that holds the following two named values, Day and Night.

Day will have the integer value 1 and Night will have the integer value 2.

2. **Employee class:** Create an *Employee* class that has properties for the following data:  
Employee name  
Employee number
3. **ProductionWorker class:** Create a class named *ProductionWorker* that is derived from the *Employee* class. The *ProductionWorker* class should have properties to hold the following data:  
Shift (of type Enum *SHIFT\_TYPE*)  
Hourly pay rate

The workday is divided into two shifts: day and night. The Shift property will hold the enum value representing the shift that the employee works.

4. **ShiftSupervisor Class:** a shift supervisor is a salaried employee who supervises a shift. In addition to a salary, the shift supervisor earns a yearly bonus when his or her shift meets production goals. Create a *ShiftSupervisor* class that is derived from the *Employee* class. The *ShiftSupervisor* class should also have the following properties:
  - a. Shift (of type Enum *SHIFT\_TYPE*)
  - b. The annual salary
  - c. The annual production bonus that a shift supervisor earns
5. **TeamLeader Class:** A team leader is an hourly paid productionworker that leads a small team. In addition to hourly pay, team leaders earn a fixed monthly bonus. Team leaders are required to attend a minimum number of hours of training per year. Design a *teamLeader* class that is derived from the *ProductionWorker* class. The *teamLeader* class should have the following properties:
  - a. Monthly bonus amount
  - b. Required number of training hours.
  - c. The number of training hours attended

6. Each of the classes above should have a member method named *Display*, that is when called it should display the relevant information to object of that class and its parent classes.

**Note,** child classes need to override this method, and parent class need to allow the method to be overridden by a child class.

7. Create an application that allows the user to create two shift supervisors one for the day and one for the night. Allows the user to add as many production workers as possible to each shift. The ability to assign and change a team lead from the list of production workers.

**Important Note:** when creating any of the objects, you need to allow the user to enter the necessary data (fill the properties) that are associated with the object and its parents.

## Submission instructions:

Include your name at the title of the form.

Compress the entire project into a zip file.

Rename the zip file with “your name.zip”

Upload the zip file to Blackboard.