



# Week9 – Inheritance

COMP90041 Programming and software  
development

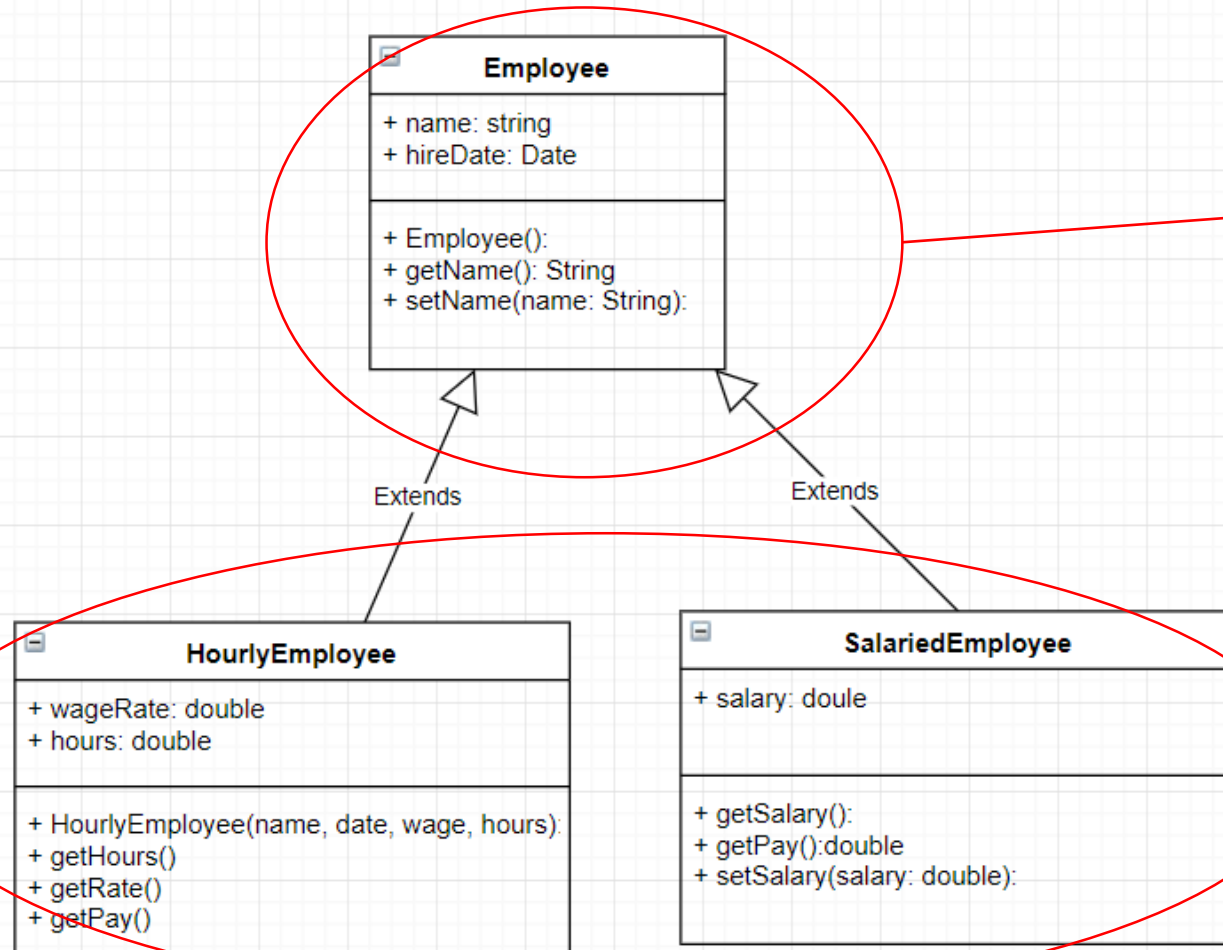




# Most important concepts of OOP

1. Abstraction
2. Encapsulation
3. Inheritance
4. Polymorphysm

# Overview of Inheritance



**Base class/parent class/  
super class**

**Derived class/ child class/  
sub class**



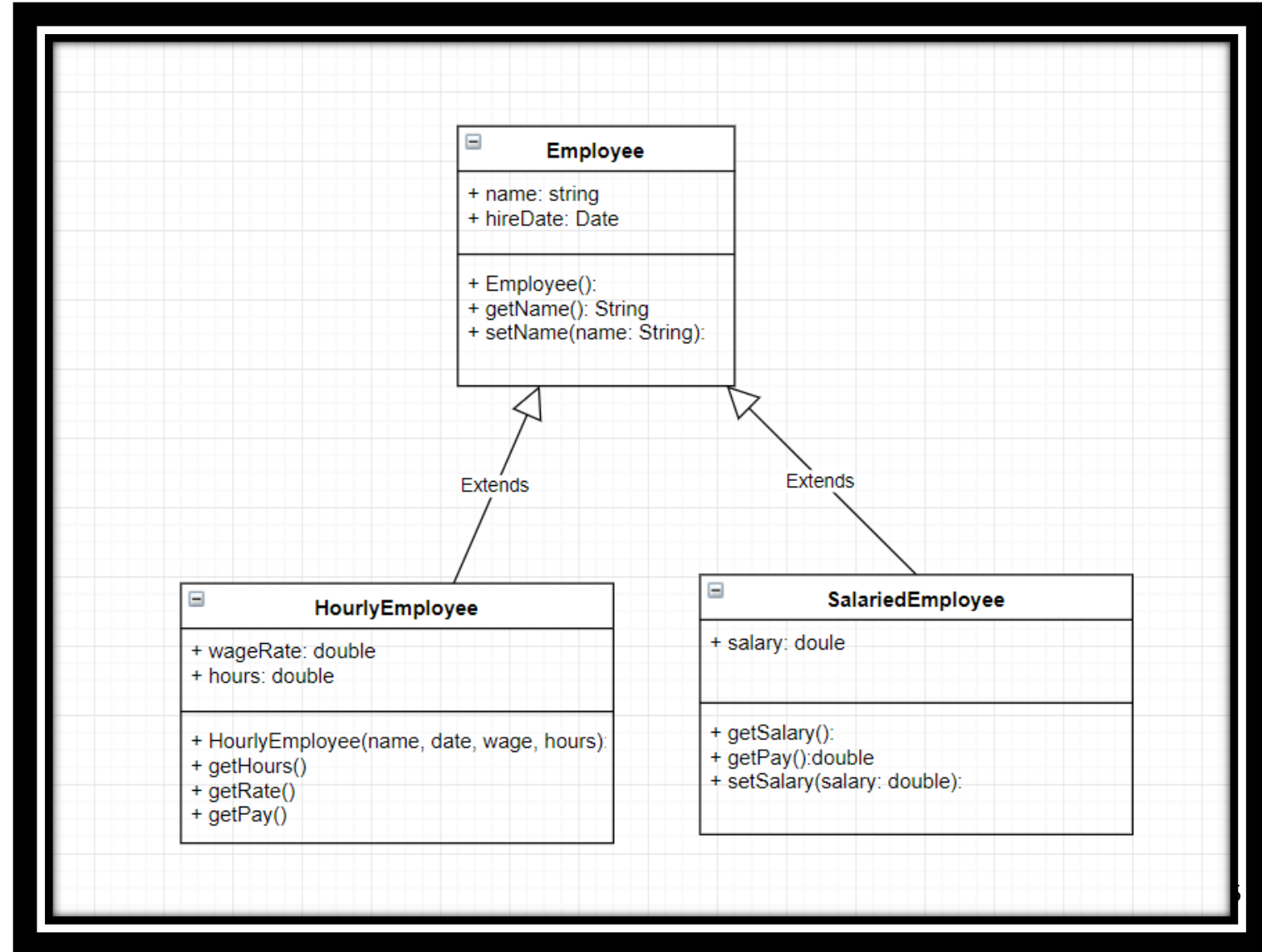
# Rules of Inheritance

1. The Derived class **inherits**:
  - all the methods
  - all the instance variables
2. The Derived class can its own have additional variables and methods

# Rules of Inheritance

1. The Derived class **inherits**:
  - all the methods
  - all the instance variables
2. The Derived class can have additional variables and methods

First Demo.....





# Override

1. The derived class can override the parent class
2. Rule:
  - Same method name
  - Same parameters
  - Can change from private to public but can not change from public to private

Second Demo.....

# Modifiers

| Modifier           | Class | Package | Subclass | World |
|--------------------|-------|---------|----------|-------|
| public             | Y     | Y       | Y        | Y     |
| protected          | Y     | Y       | Y        | N     |
| <i>no modifier</i> | Y     | Y       | N        | N     |
| private            | Y     | N       | N        | N     |



# The **Object** Class

Every class is a decendent of the class Object

- equals()
- toString()



# What is a well designed class looks like?

1. Define several constructor and each of them serves different purpose
2. Define getter and setter for each instance variables
3. Define toString(), equals()

# What is a well designed class looks like?

1. Define several constructor and each of them serves different purpose
2. Define getter and setter for each instance variables
3. Define toString(), equals()

However, it is not always necessary, it all depends on the requirements of your program