



**មជ្ឈមណ្ឌលកូរ៉េ សហ្វ្វែរ អេច អ ឌី**

**Korea Software HRD Center**

**កម្មវិធីបង្រៀន Java Programming ជាកាសាខ្មែរ**

**Online Java Training Course**

**Advisor: Dr. Kim Tae Kyung**

---



[www.kshrd.com.kh](http://www.kshrd.com.kh)

# មេរៀនទី៣៖ Polymorphism

៣.១. អ្វីជា Polymorphism?

៣.២. លក្ខណៈ: Overloading

៣.៣. លក្ខណៈ: Overriding

## ៣.១. អ្វីជា Polymorphism?

- **Polymorphism:** ជាពាក្យក្រិច (Greece) ដែល Poly មានន័យថា " ច្រើន " និង morph មានន័យថា " ទម្រង់ " ។
- **Polymorphism** ជា Concept ដែលអាចអោយ Object ( **Class** ) មួយ មានច្រើនទម្រង់ ។
- លក្ខណៈច្រើនទម្រង់៖
  - ☞ ជា **Method** ដែលមានឈ្មោះដូចគ្នាអាចធ្វើការផ្សេងគ្នា
  - ☞ លក្ខណៈច្រើនទម្រង់ ត្រូវបានបែងចែកជា២
    - ✍ **Overloading**
    - ✍ **Overriding**

## ៣.១. អ្វីជា Polymorphism?

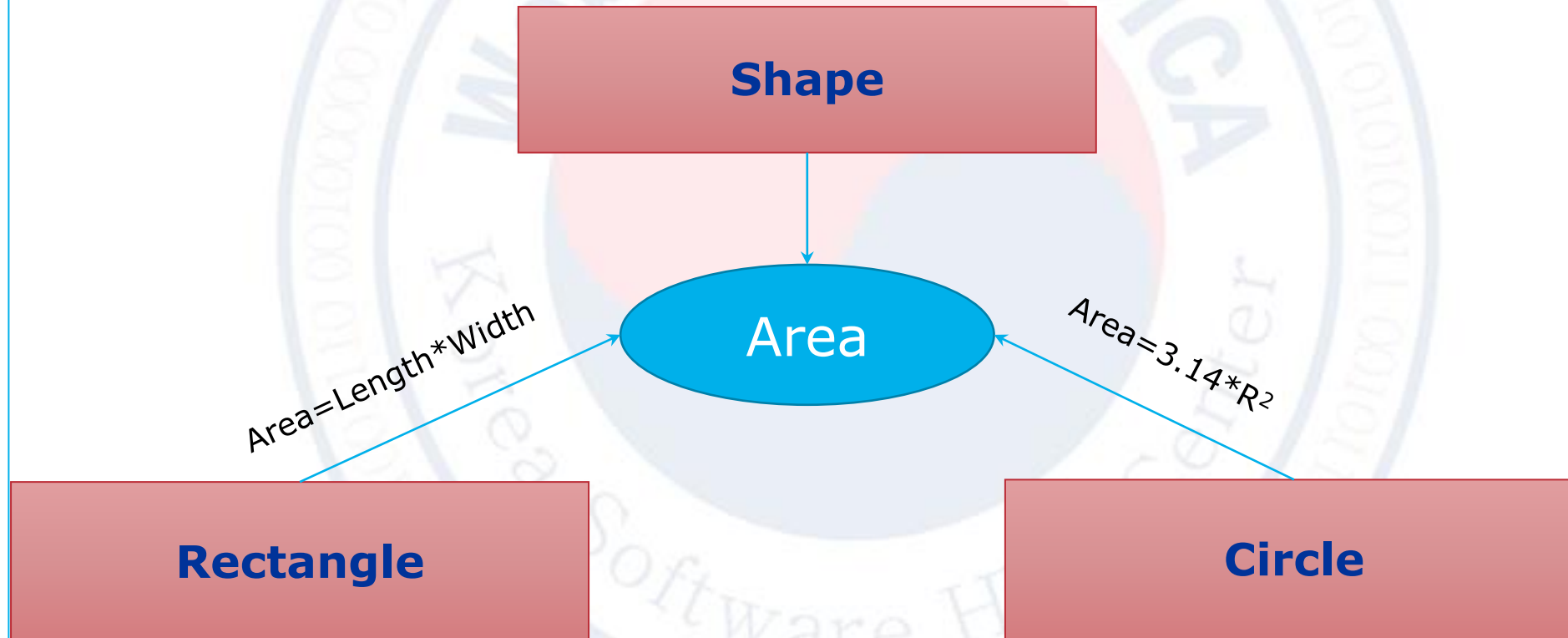
- លក្ខណៈ **Polymorphism** គឺអនុញ្ញាត អោយ **Sub Class** មួយមានមុខងារផ្ទាល់ខ្លួនមួយ ដែលបន្ថែមនូវលក្ខណៈរបស់ **Super Class** ឬ ស្រដៀងទៅនឹង **Super Class** ។

**Example:**

```
public class Shape{}  
public class Rectangle extend Shape{}  
public class Square extend Shape{}
```

ដែល **Class Shape** — មាន **Method Area** ដូច្នោះ **Class Rectangle** និង **Class Circle** ក៏មាន **Method Area** ដែលមានមុខងារខុសគ្នា ។

## ៣.១. អ្វីជា Polymorphism?



## ៣.២. លក្ខណៈ: Overloading

□ **Overloading:** ជា **Methods** ដែលមានឈ្មោះ: ដូចគ្នាតែមាន Signature

ខុសគ្នា ។

```
class DisplayOverloading {  
  
    public void disp(char c) {  
        System.out.println(c);  
    }  
  
    public void disp(char c, int num){  
        System.out.println(c + " "+num);  
    }  
  
}
```

Answer:

a

a 10



## ៣.៣. លក្ខណៈ: Overriding

- **Overriding**: ជា **Method** ដែលមានឈ្មោះដូចគ្នា ហើយមាន **Signature** ដូចគ្នា។

```
class Animal{
    public void move(){
        System.out.println("Animals can move");
    }
}

class Dog extends Animal{
    public void move(){
        System.out.println("Dogs can walk and run");
    }
}

public class TestDog{
    public static void main(String args[]){
        Animal a = new Animal();
        Animal b = new Dog();
        a.move(); // runs the method in Animal class
        b.move(); //Runs the method in Dog class
    }
}
```

# សមាជិក

## ក្រុមអ្នកស្រាវជ្រាវ



ល. តាំង បូរិក

[borith\\_tang@yahoo.com](mailto:borith_tang@yahoo.com)



ល. សៀង វិជ្ជាវិទ្ធ

[rith.magnificent.9@gmail.com](mailto:rith.magnificent.9@gmail.com)



ល. ចាន់ សិរិរតនៈ

[chanseryratanak@yahoo.com](mailto:chanseryratanak@yahoo.com)



ល. ដារ៉ា ពោញចិត្ត

[darapenhchet@gmail.com](mailto:darapenhchet@gmail.com)



ល. ចាន់ បូរ៉ា

[borachann@gmail.com](mailto:borachann@gmail.com)

## ក្រុមផលិតវីដេអូ



ល. ឈុន បញ្ញាភាន់

[chhunpanharath@gmail.com](mailto:chhunpanharath@gmail.com)



ល. ព្រាប វិទ្ធី

[Itpreap.vuthy@gmail.com](mailto:Itpreap.vuthy@gmail.com)



ល. ហង្ស បូរី

[hongboreyrupp@gmail.com](mailto:hongboreyrupp@gmail.com)



ល. ហួ ឈុនឡេង

[huochhunleng@yahoo.com](mailto:huochhunleng@yahoo.com)



មេរៀនបន្ទាប់ និងធ្វើការបង្ហាញពី Abstraction