



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

មេរៀនទី៣៖ ប្រភេទ Modifier របស់ Java

៣.១ តើ Modifiers ជាអ្វី ?

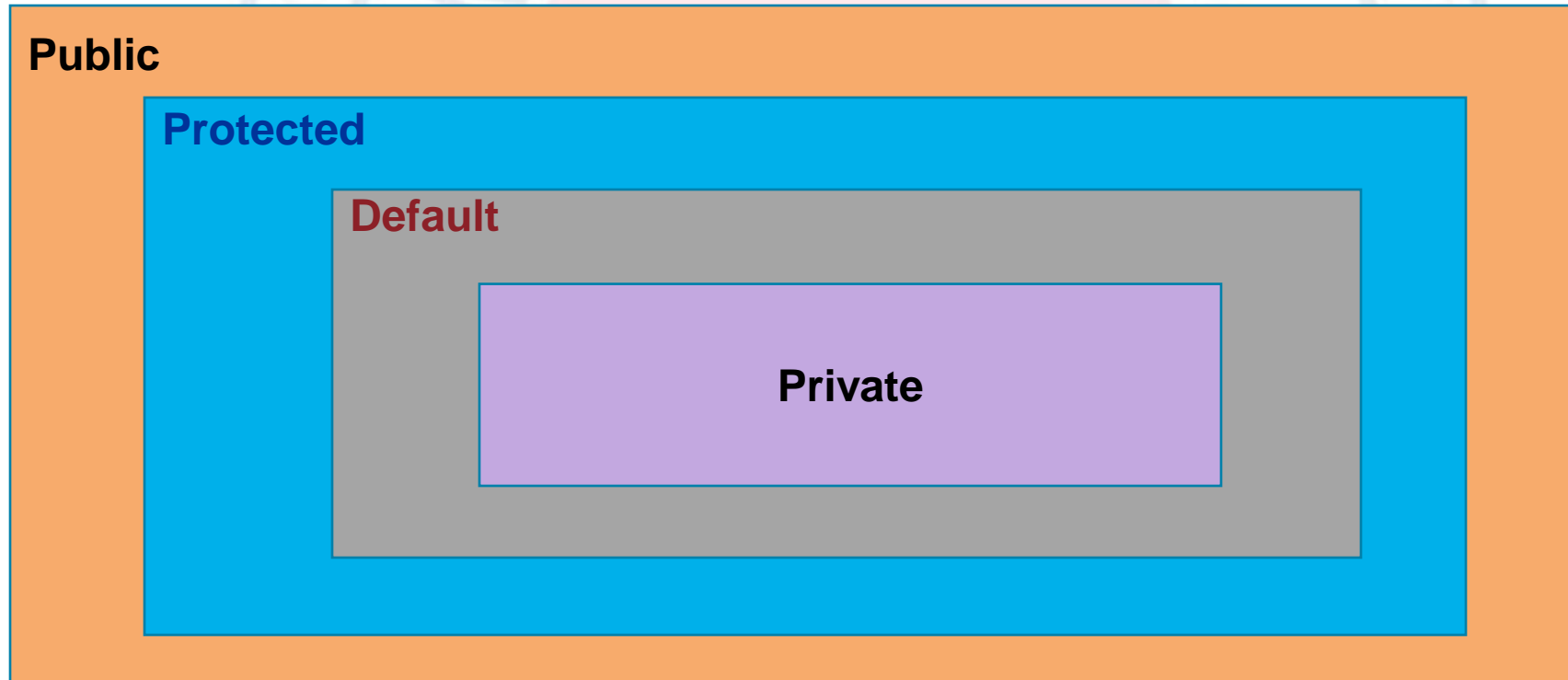
៣.២ ប្រភេទ Modifiers

៣.៣ អ្វីទៅជា Access Modifiers ?

៣.៤ អ្វីទៅជា Non-Access Modifiers ?

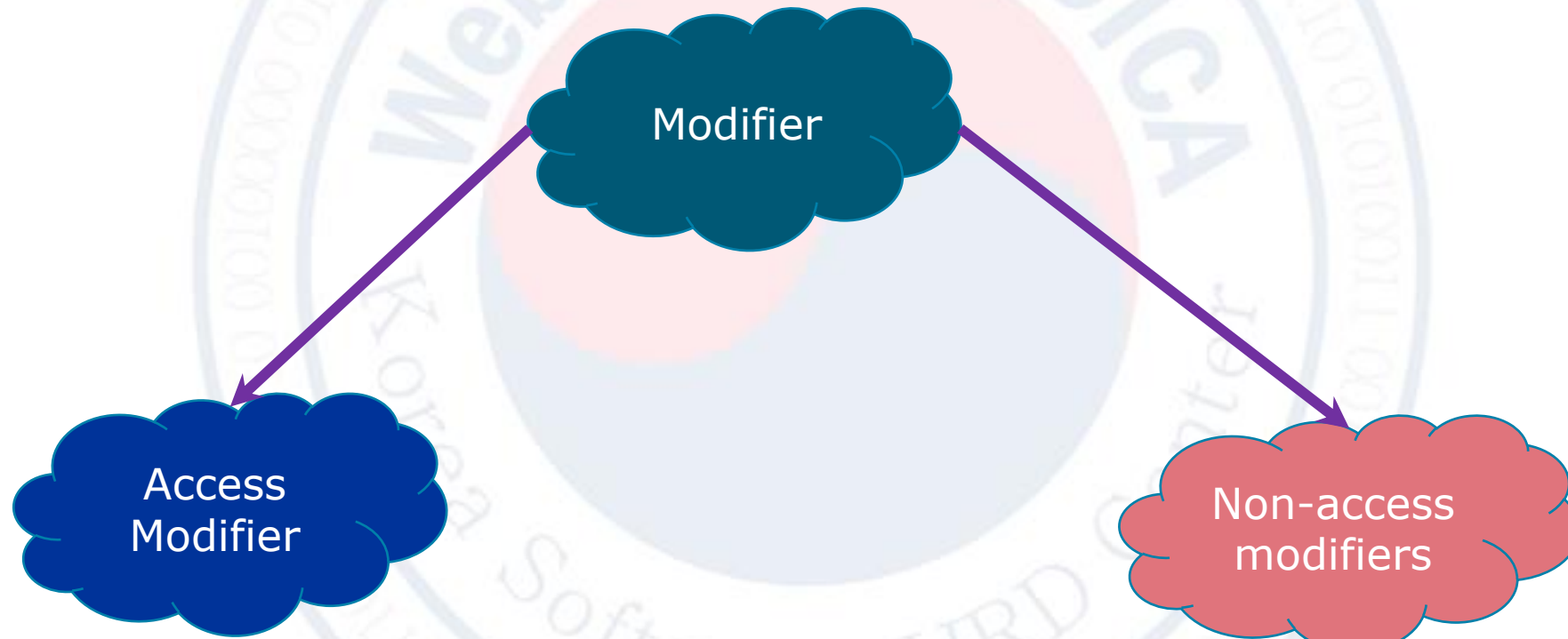
៣.១ តើ Modifiers ជាអ្វី ?

- **Modifier** គឺជាប្រភេទ **Keyword** ដែលប្រើសម្រាប់ កំណត់ អត្ថន័យ (Definition) ទៅឲ្យ **Class, Method** ឬ **Variable** ។



៣.២ ប្រភេទ Modifiers

- នៅក្នុងភាសា Java គេបានបែកចែក Modifier ជា ២ប្រភេទ៖



៣.៣ អ្វីទៅជា Access Modifiers ?

- **Access Modifier** គឺជាប្រភេទ **Keyword** ដែលប្រើសម្រាប់ កំណត់ កំរិតនៃ ការ **Access** ទៅឲ្យ **Class, Method** ឬ **Variable** ។
- **Access Modifier** មាន៤ ដូចជា៖
 - ☞ **Default** : អាច **Access** បានក្នុង **Class** ឬ **Package** ជាមួយគ្នា ។
 - ☞ **Public** : អាច **Access** បានគ្រប់ទីកន្លែង ។
 - ☞ **Protected** : អាច **Access** បានក្នុង **Class** ឬ **Package** ជាមួយគ្នា និង subclass របស់វា។
 - ☞ **Private** : អាច **Access** បានតែក្នុង **Class** ជាមួយគ្នាប៉ុណ្ណោះ។

៣.៣ អ្វីទៅជា Access Modifiers ?

- ដើម្បីឲ្យកាន់តែងាយស្រួលក្នុងការស្វែងយល់អំពី **Access Modifier** សូមពិនិត្យមើលតារាងខាងក្រោម:

Access Modifier	នៅក្នុង class ជាមួយគ្នា	នៅក្នុង package ជាមួយគ្នា	នៅក្នុង subclass	នៅក្នុង package ផ្សេងគ្នា
Public	✓	✓	✓	✓
Protected	✓	✓	✓	X
Default	✓	✓	X	X
Private	✓	X	X	X

៣.៣ អ្វីទៅជា Access Modifiers ?

- **សំគាល់:** Class, Method, ឬ Variable ណាដែលយើងមិនបានដាក់ Access Modifier នឹងមាន **Default Modifier** ជាស្វ័យប្រវត្តិ ។

```
public class AccessModifier {  
  
    //Public modifier  
    public void publicMethod(){  
        privateMethod();  
    }  
    //Protected modifier  
    protected void protectedMethod(){  
        privateMethod();  
    }  
    //Default modifier  
    void defaultMethod(){  
        privateMethod();  
    }  
  
    //Private modifier  
    private void privateMethod(){  
    }  
  
}
```

៣.៤ អ្វីទៅជា Non-Access Modifiers ?

- **Non-Access Modifier** គឺជាប្រភេទ **Keyword** ដែលប្រើសម្រាប់ បន្ថែមមុខងារមួយចំនួនអោយ **Class, Method** ឬ **Variable** ។
- Java បានផ្តល់នូវ **Java Non-Access Modifiers** មួយចំនួន រួមមានដូចខាងក្រោម៖
 - ↳ **Static Modifier**
 - ↳ **Final Modifier**
 - ↳ **Abstract Modifier**
 - ↳ **Synchronized**

៣.៤ អ្វីទៅជា Non-Access Modifiers ?

☞ **Static Modifier** គឺជាប្រភេទ **Keyword** ដែលប្រើសម្រាប់បង្កើត Method និង Variable នៅក្នុង Class ដែលគេអាចប្រើវាបាន ដោយមិនចាំបាច់បង្កើត Object ចេញពី Class នោះទេ ។

```
//StaticModifier.java
public class StaticModifier {

    //Static variables
    public static int x=5;
    public static int y=2;

    //Static method
    public static int sum(){
        return x+y;
    }
}
```

```
//Test.java
public class Test {

    public static void main(String args[]){

        System.out.println("X = "+ StaticModifier.x);

        System.out.println("Y = "+ StaticModifier.y);

        System.out.println("Sum of X & Y is "
            + StaticModifier.sum());
    }
}
```

៣.៤ អ្វីទៅជា Non-Access Modifiers ?

☞ **Final Modifier** គឺជាប្រភេទ **Keyword** ដែលប្រើសម្រាប់បញ្ចប់ការ **Implement** ទៅលើ **Class, Method** ឬ **Variable** ។

```
//Final class - cannot be inherited.  
public final class FinalModifier {  
  
    //Final variable - cannot change its initial value  
    final int value = 10;  
  
    //Final method - cannot be overloaded or overridden  
    public final void changeValue(){  
        value = 12; //will give an error  
    }  
}
```

៣.៤ អ្វីទៅជា Non-Access Modifiers ?

👉 **Abstract Modifier** គឺជាប្រភេទ keyword ដែលប្រើសម្រាប់បង្កើត abstract class និង method។

```
//AbstractModifier.java
//Abstract class - cannot instantiate(cannot be used to create object)
public abstract class AbstractModifier {

    /* Abstract method - has no body and has
    to be implemented in subclass */
    abstract void changeColor();
}

//Test.java
public class Test extends AbstractModifier {
    void changeColor() {}
}
```

៣.៤ អ្វីទៅជា Non-Access Modifiers ?

☞ **Synchronize Modifier** គឺជាប្រភេទ **keyword** ដែលប្រើសម្រាប់គ្រប់គ្រង threads ក្នុងការប្រើប្រាស់ method ពេលគឺវាកំណត់ឲ្យមានតែ thread មួយប៉ុណ្ណោះ ដែលអាចប្រើ (**access**) method នោះ ក្នុងពេលតែមួយ ហើយវាអាចប្រើជាមួយ Access Modifier (**public, protected, default, private**) បាន ។

❖ អ្នកនឹងរៀនអំពី thread នៅមេរៀនទី ១០.១

សមាជិក

ក្រុមអ្នកស្រាវជ្រាវ



ល. ខេង ចាន់វិជ្ជា

vichea@rocketmail.com



ល. សែម ចិត្រា

sabaychitra84@gmail.com



ល. ឡៅ ស៊ីនឡេង

sunlenglao@gmail.com.com



ក. លីម សុខហេង

lim.sokheng1@gmail.com



ល. យ៉ូ វណ្ណវិទ្ធ

ravuthz@gmail.com

ក្រុមផលិតវីដេអូ



ល. ឈុន បញ្ញាភក្តី

chhunpanharath@gmail.com



ល. ព្រាប វិទ្ធ

Itpreap.vuthy@gmail.com



ល. ហង្ស បូរី

houngboreyrupp@gmail.com



ល. ហួ ឈុនឡេង

huochhunleng@yahoo.com



មេរៀនបន្ទាប់ និងធ្វើការបង្ហាញពី Basic Operators