

JAVA BEGINNER

INHERITANCE



Abstract of New Technology

Website <http://www.antkh.com>

Tel 010 / 016 66 66 53

Prepared By Tum Sakal

Tel 087 36 31 30

Mail tumsakal.ts@gmail.com

Facebook [sak kal](#)

WHAT IS INHERITANCE?

- Inheritance គឺជាលក្ខណៈពិសេសមួយរបស់ OOP
- Inheritance អនុញ្ញាតអោយ Programmer សរសេររបង្កើត Class ថ្មីមួយដោយផ្អែកទៅលើ Class ចាស់ដែលមានស្រាប់

```
[modifier] class SubClass extends SupperClass{  
  
}
```

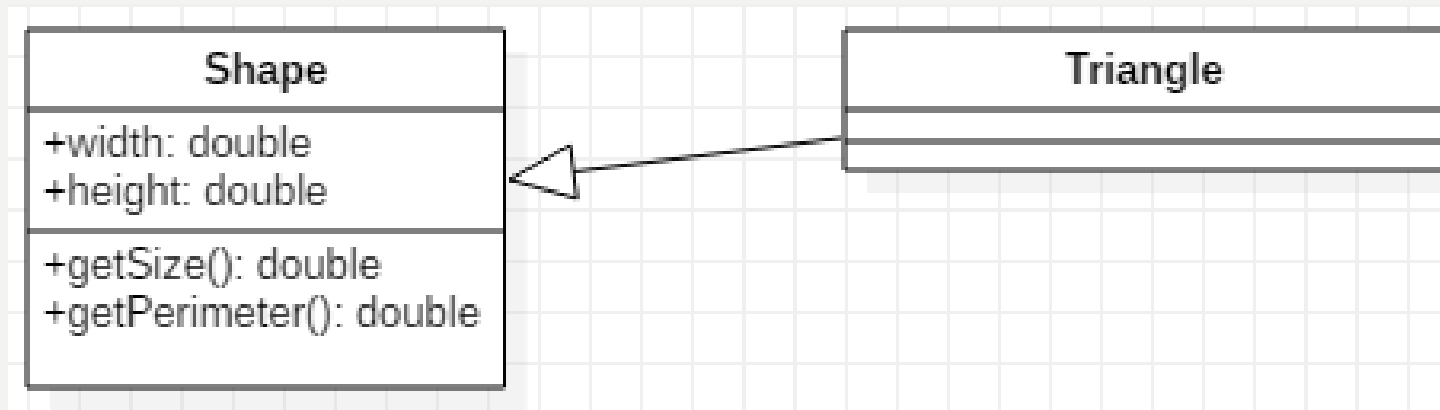
- “*extends*” ជា Keyword ប្រើសំរាប់អនុញ្ញាតអោយ Sub Class ធ្វើការ Inherit ទៅកាន់ Supper Class

WHEN & WHY

- Inheritance អនុញ្ញាតិអោយអោយអ្នកសរសេរកូដធ្វើរឿងបីគឺ
 - Reuse
 - Extend
 - And Modify

REUSE

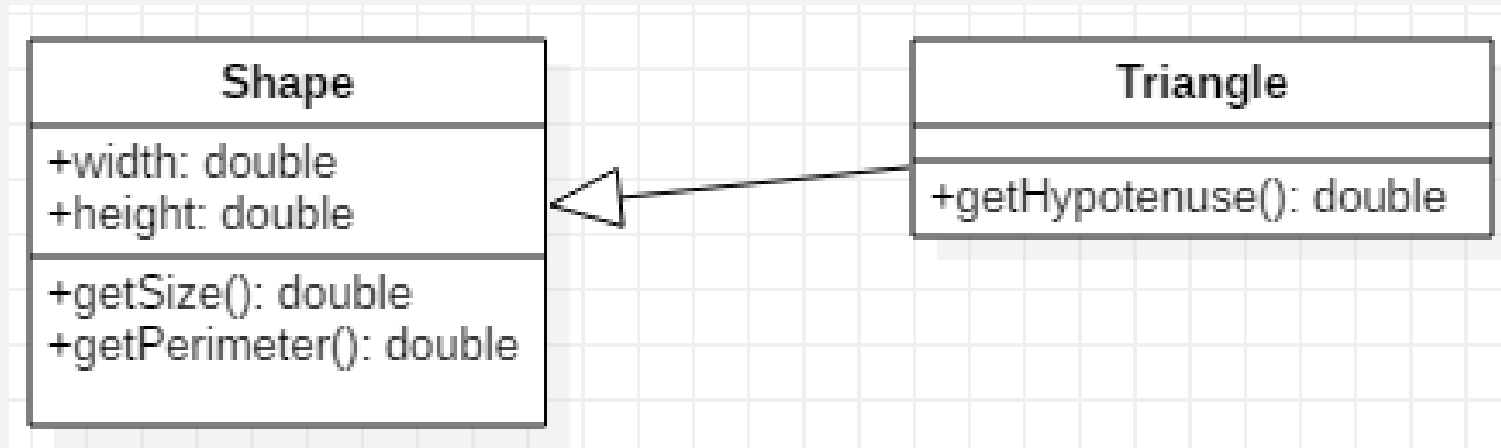
- សំដៅទៅលើការ ទាញយក Member របស់ Class ចាស់មកធ្វើជា Member របស់ Class ថ្មីដោយមិនចាំបាច់សរសេរបង្កើតឡើងវិញ



- Member ដែលអាចទាញយកមកបានមាន **Non-Private Instance Field** និង **Non-Private Instance Method**

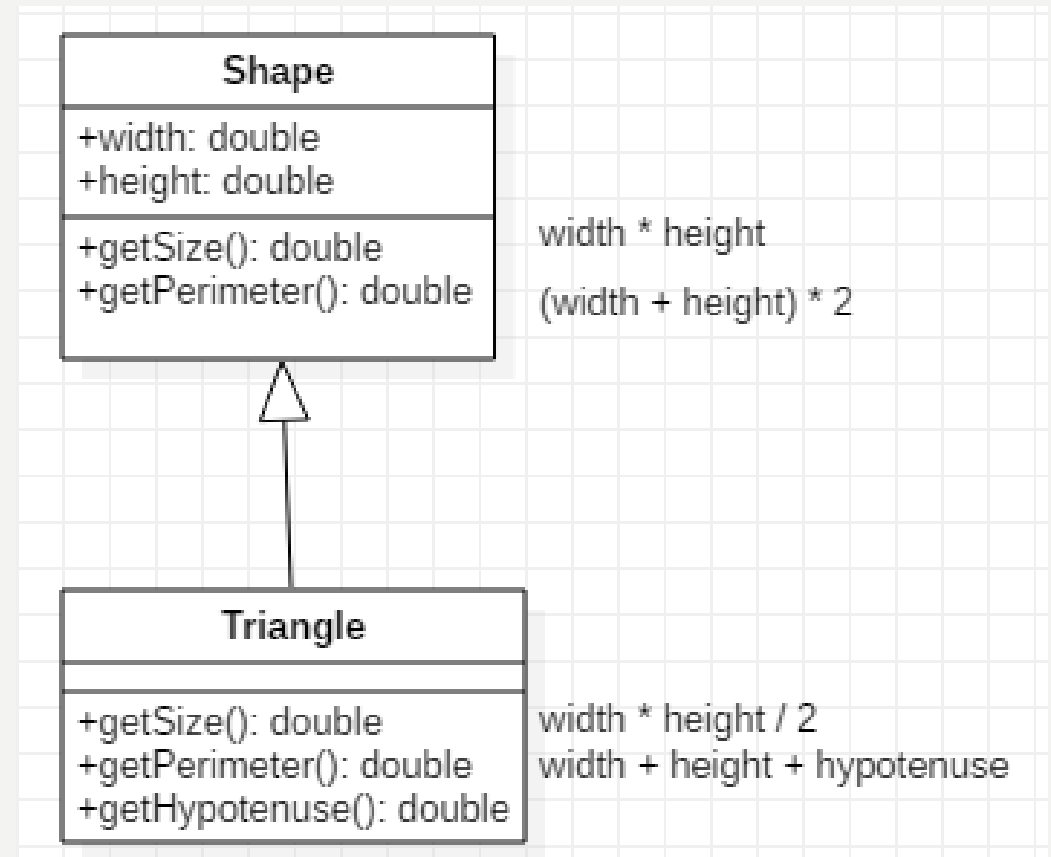
EXTEND

- សំដៅទៅលើ ការបន្ថែមមុខងារថ្មីដែល Class ចាស់មិនមាន



MODIFY

- សំដៅទៅលើការ Override នូវមុខងារចាស់របស់ Supper Class



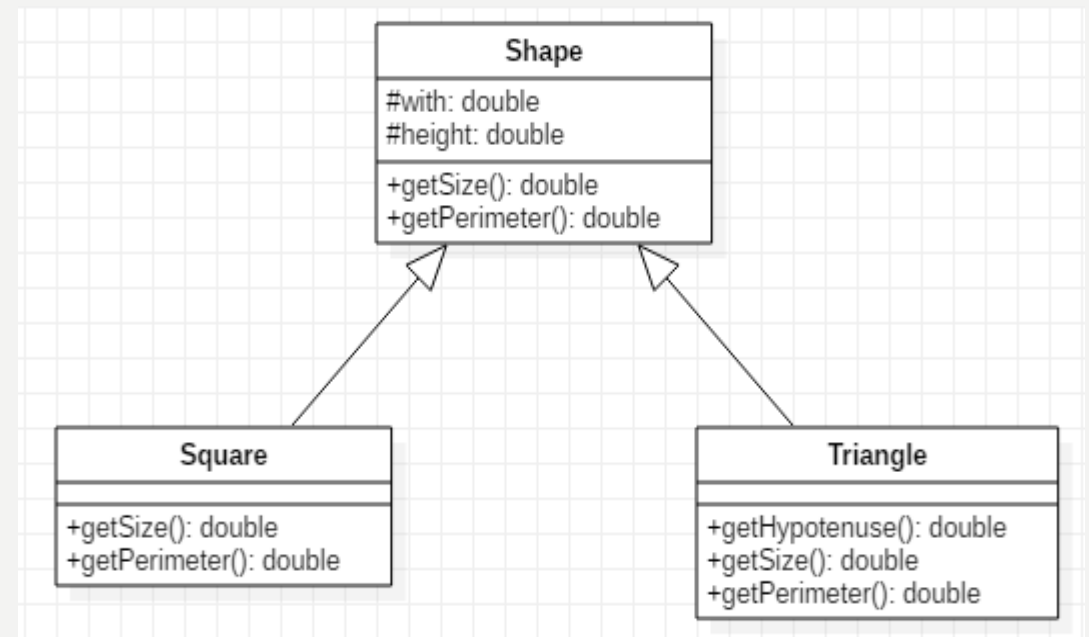
PROTECTED MEMBER

- Member ដែលមាន Access Level ជា Protected ត្រូវបានការពារសំរាប់អោយតែ Sub Class អាចហៅ និង Inherit បានតែប៉ុន្មោះ
- មានន័យថា មិនអនុញ្ញាតអោយមានការហៅ Member នោះទៅប្រើដោយផ្ទាល់តាមរយៈ Object បានទេ

Shape sh = new Shape();

~~sh.width = 100D;~~

width & height is protected

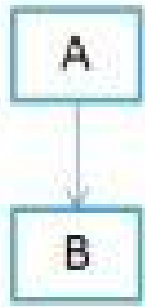


OVERRIDE ANNOTATION

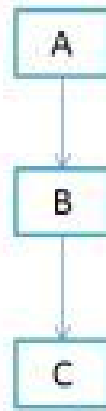
- “@Override” វាជា Annotation មួយដែលប្រើសំរាប់ ប្រាប់ Java Compiler ថា Member នោះជា Member របស់ Supper Class ដែល Sub Class បាន Override

INHERITANCE

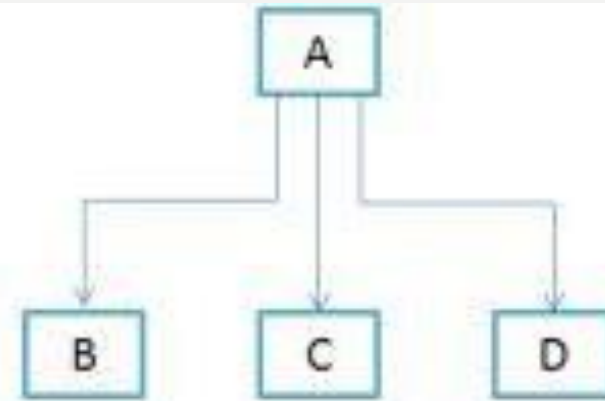
- លក្ខណៈរបស់ Inheritance នៅក្នុងភាសា Java មាន៖
 - **Single Inheritance** :
Sub Class មួយអាច Inherit បានតែ Super Class មួយ
 - **Multilevel Inheritance**: Sub Class មួយអាច Inherit ពី Sub Class មួយទៀតបាន
 - និង **Hierarchy Inheritance** :
Class មួយអាចផ្តល់ Inheritance ទៅកាន់ Class កូនច្រើនមិនកំណត់



(a) Single Inheritance



(d) Multilevel Inheritance



(c) Hierarchical Inheritance

PREVENT INHERITANCE

- “**final**” ជា Keyword មួយដែលប្រើដើម្បីកំណត់អោយ Class មួយក្លាសទៅជាប្រភេទ Class ដែលមិនអាចអោយ Class ដទៃ Inherit ពីវាបាន
- Sub Class មិនអាច Inherit ពី **final class** បានទេ

```
[modifier] final class ClassName{  
  
}
```

PREVENT OVERRIDING

- “**final**” ជា Keyword មួយដែលប្រើដើម្បីកំណត់អោយ Method របស់ Class ក្លាយទៅជាប្រភេទ Method ដែលមិនអាចអោយ Override បាន

```
[modifier] final type methodName([parameter]){  
  
    statements...  
  
    return result;  
  
}
```

TYPE REFERENCE

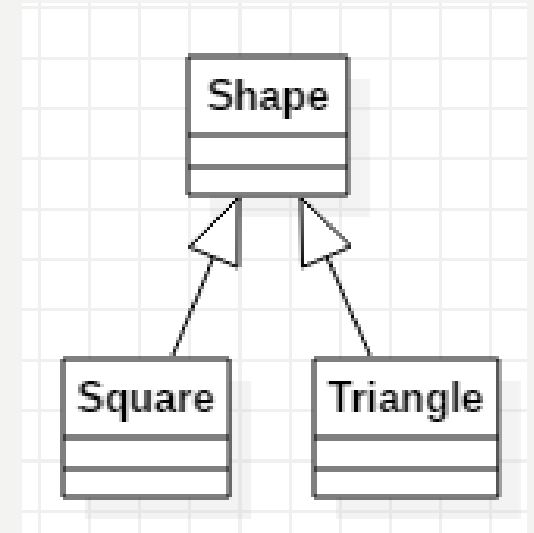
- នៅក្នុង Inheritance អនុញ្ញាតអោយមានការប្រកាស Object មួយដោយប្រើប្រាស់ប្រភេទ Supper Class និងបង្កើតចេញជា Object ដោយប្រើប្រាស់ Constructor របស់ Sub Class

Shape sh1 = new *Shape*();

Shape sh2 = new *Square*();

Shape sh3 = new *Triangle*();

- Object នៃប្រភេទ Supper Class អាចប្រើបានច្រើនទំរង់នៃ Sub Class

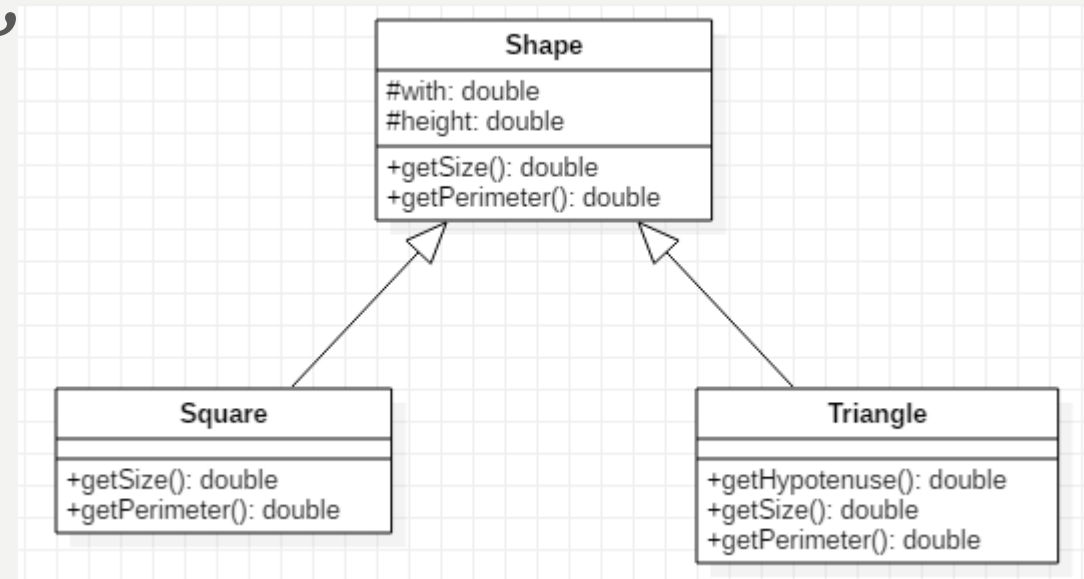


TYPE REFERENCE

- ដោយ Object ត្រូវបានប្រកាសដោយប្រើ Supper Class ជាប្រភេទទទួល នាំអោយយើងអាចហៅបានតែ Member ដូចដែលបានប្រកាសនៅក្នុង Supper Class តែប៉ុន្មោះ:

```
Shape sh = new Triangle();
```

```
sh.getHypotenuse();
```



TYPE REFERENCE

- នៅពេលជំនើការ Method នៃជំនាន់ Sub Class ដែលបាន Override ត្រូវបានហៅអោយជំនើការជំនួសអោយ Method របស់ Supper Class

```
Shape sh1 = new Triangle();  
sh1.getSize();  
  
Shape sh2 = new Square();  
sh2.getSize();
```

