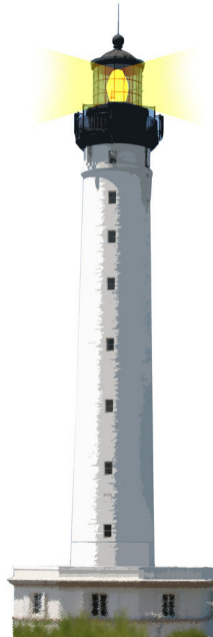


About Types and Lookup (Interface variation)

S. Ducasse



Remember: Static vs. Dynamic Types

```
A a = new B();
```

- The static type of variable `a` is `A` i.e., the statically declared class to which it belongs.
 - The static type never changes.
- The dynamic type of `a` is `B` i.e., the class of the object currently bound to `a`.
 - The dynamic type may change throughout the program.



Setting the stage

```
public interface Acceptable {  
    public void accept();  
}
```

```
public class Person implements Acceptable {  
    public void accept(){  
        System.out.println("accept");  
    }  
    public void agree(){  
        System.out.println("agree");  
    }  
}
```



Normal

```
Person p = new Person();  
p.accept();  
p.agree();
```

```
accept  
agree
```



Normal too

```
Person p = new Person();  
Acceptable r = p;  
r.accept;
```

```
accept
```



Influence of static type

```
Person p = new Person();  
Acceptable r = p;  
r.agree(); >>> BREAK!
```

```
java: cannot find symbol  
symbol: method agree()  
location: variable a of type designCorner.Acceptable
```

- At compile time, the typechecker does not use the dynamic type of the object.
- Within the static type `Acceptable` there is no method `agree()`.
- So the compiler rejects the program as invalid (even though at runtime no error would occur).



What you should know

- Static types are used to identify at compile time which methods to lookup
- Lookup will look for such method at runtime



A course by

S. Ducasse, G. Polito, and Pablo Tesone



Except where otherwise noted, this work is licensed under CC BY-NC-ND 3.0 France
<https://creativecommons.org/licenses/by-nc-nd/3.0/fr/>