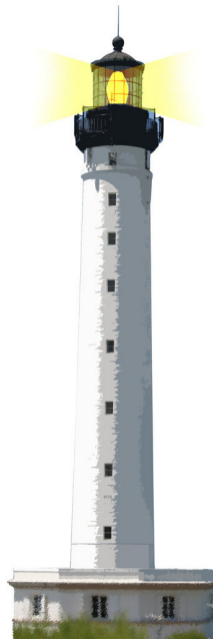


DieHandle new vs. self class new

When classes are first class citizen

S. Ducasse and L. Fabresse



Goal

- self represents the receiver
- Class receive messages



Imagine

To support

```
(DieHandle new add: (Die faces: 4); yourself)  
+ (DieHandle new add: (Die faces: 6); yourself)
```

We defined + as

```
DieHandle >> + aDieHandle  
| handle |  
handle := DieHandle new.  
self dice do: [ :each | handle addDie: each ].  
aDieHandle dice do: [ :each | handle addDie: each ].  
^ handle
```



What Is the difference...

Between

```
DieHandle >> + aDieHandle  
| handle |  
handle := DieHandle new.
```

And

```
DieHandle >> + aDieHandle  
| handle |  
handle := self class new.
```

Let us see....



What if we create a new subclass

```
DieHandle subclass: MemoDieHandle
```

```
....
```

```
(MemoDieHandle new add: (Die faces: 4); yourself)  
+ (MemoDieHandle new add: (Die faces: 6); yourself)  
> aDieHandle
```

We get a DieHandle instance back and not a MemoDieHandle instance!!!



Solution 1: Creating a hook

```
DieHandle >> + aDieHandle  
| handle |  
handle := self handleClass new.  
self dice do: [ :each | handle addDie: each ].  
aDieHandle dice do: [ :each | handle addDie: each ].  
^ handle
```

```
DieHandle >> handleClass  
^ DieHandle
```

A subclass may redefine `handleClass`

```
MemoDieHandle >> handleClass  
^ MemoDieHandle
```



Solution 1: Creating a hook

```
(MemoDieHandle new add: (Die faces: 4); yourself)
+ (MemoDieHandle new add: (Die faces: 6); yourself)
> aMemoDieHandle
```

We get an instance of the subclass!



But we can do better!

Let us see

- In each subclass we should redefine the hook method `handleClass`
- This is tedious



Solution 2

```
DieHandle >> + aDieHandle  
| handle |  
handle := self class new.  
self dice do: [ :each | handle addDie: each ].  
aDieHandle dice do: [ :each | handle addDie: each ].  
^ handle
```

- self class always returns the class of the receiver
- We get instances of the same kind of the receiver

Conclusion

If we define a subclass of DieHandle, and send the message + to an instance

- With DieHandle new, + **does not** return an instance of the subclass but of DieHandle
- With self class new, + **returns** an instance of the receiver: an instance of a potential subclass



A course by

S. Ducasse, L. Fabresse, 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/>