

		HAS-A	IS-A
		A → B	C
class A {	interface B{ }		
B ob;			
setOb(B ob){	class C implements B{		
this.ob=ob;	}		
}			
	Normal	Anonymous + Instance Block	
B ob = new C();		new A() {{	
		setOb(new C());	
A oa = new A();		}	
oa.setOb(ob);			

```
class FlatFileItemReader {
    Resource resource;
    void setResource(Resource r){
        this.resource=r;
    }
}

new FlatFileItemReader() {
    setResource(new ClassPathResource());
};
```

class FlatFileItemReader {	interface LineMapper{ }
LineMapper mapper;	class DefaultLineMapper
setLineMapper(LM m){	implements LineMapper {
mapper=m;	}
}	}
A → B (I)	F of = new F();
(C) ↑	E oe = new E();
C → D (I)	oe.setOf(of);
(C) ↑	C oc =new C();
E → F (C)	oc.setOd(oe);
(C)	A oa = new A();
	oa.setOb(oc);
	new A() {{
	setOb(new C()) {{
	setOd(new E() {{
	setOf(new F());
	}}
	}}
	}}