

Data Class

Blazej.Pietrzak@cs.put.poznan.pl

What is Data Class?

- A class with public fields that are not final
- A class having getter/setter methods that are not used in the code
- A class with improper collection encapsulation
- A class with only getter/setter methods (structure equivalent)

Data Class –Motivation

- Improper encapsulation allow clients to manipulate the contents without the owning class knowing what is going on
- No behavior usually makes the class not paying for itself and should be removed by moving methods or parts of code from classes that operate on the Data Class's data to the Data Class

Data Class –Motivation –cont.

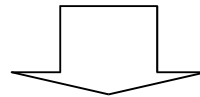
Removing Data Class reduces:

- Cost of maintenance
- Duplicates in code
- Code size and readability
- Errors in code

Data Class -examples

Public fields

```
public class NamesCollection {  
    public String collectionName;  
    ...  
}
```



```
public class NamesCollection {  
    private String collectionName;  
    public String getCollectionName() {  
        return collectionName;  
    }  
    public String setCollectionName(String name) {  
        collectionName = name;  
    }  
}
```

Data Class examples cont.

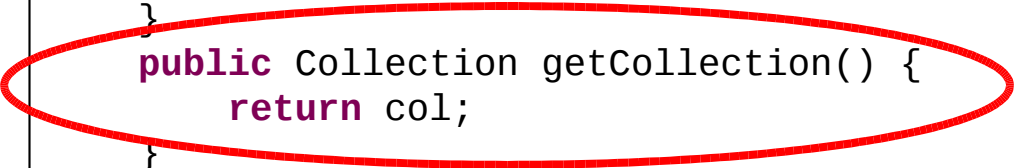
Improper collection encapsulation

```
public class NamesCollection {  
    private Set names = new HashSet();  
  
    public Set getNames() {  
        return names;  
    }  
  
    public void setNames(Set names) {  
        this.names = names;  
    }  
}
```

Data Class -examples cont.

Data Class -Error Example

```
class ServerClass {  
    private int size = 0;  
    private Collection col = new ArrayList();  
    public void add(Object obj) {  
        size++;  
        col.add(obj);  
    }  
    public void remove(Object obj) {  
        size--;  
        col.remove(obj);  
    }  
    public Collection getCollection() {  
        return col;  
    }  
    public int getSize() {  
        return size;  
    }  
}
```



Data Class -examples cont.

Data Class -Error Example -cont.

```
class ClientClass {  
    private ServerClass server;  
    ...  
    public printContents() {  
        Collection col = server.getCollection();  
        col.remove(someObject);  
  
        for (int i = 0; i < server.getSize(); i++){  
            out.println(col.get(i));  
        }  
    }  
    ...  
}
```

Data Class examples cont.

Proper collection encapsulation

```
public class NamesCollection {  
    private Set names = new HashSet();  
  
    public void addName(String name) {  
        return names.add(name);  
    }  
    public void removeName(String name) {  
        return names.remove(name);  
    }  
    ...  
}
```

Data Class examples cont.

Proper collection encapsulation -cont.

```
...
    public Set getNames() {
        return Collection.unmodifiableSet(names);
    }
    public void initializeNames(Set names) {
        this.names.addAll(names);
    }
}
```