

public class Calisthenics
{



Object Calisthenics

21.02.2022

Timon Schmid

CSS

Object Calisthenics - Topics

1 Don't use the else keyword

2 Wrap all primitives and strings

3 First class collections

Don't use the else keyword

```
if (statement) {  
} else {  
}
```

Why you should not use the else keyword

Promotes a main execution lane with few special cases.

Try using the Null Object pattern to express that a result has no value

Suggests polymorphism to handle complex conditional cases, making the code more explicit

Wrap all primitives and strings

```
/** some class **/  
int distanceInKm;  
  
/** instead use own type **/  
public class Distance {  
  
    private int distanceInKm;  
  
    public int getDistance() {  
        return distanceInKm;  
    }  
  
}
```

Why you should wrap all primitives and strings

Primitives can express anything so we tend to use them for everything

Cure for Primitive Obsession code smell

Make the code more explicit by expressing intent through types that become behavior attractors.

First class collections

```
/** instead of this **/  
public class TicTacToe {  
  
    private EnumMap<Field, Player> grid = new EnumMap<>(Field.class);  
    private Player currentPlayer = Player.X;  
  
    public void placeMark(Field field) {  
        grid.put(field, currentPlayer);  
    }  
  
}
```

```
/** do this **/  
public class TicTacToe {  
  
    private Grid grid = new Grid();  
    private Player currentPlayer = Player.X;  
  
    public void placeMark(Field field) {  
        grid.place(field, currentPlayer);  
    }  
  
}  
  
public class Grid {  
  
    private EnumMap<Field, Player> fieldPlayerMap = new EnumMap<>(Field.class);  
  
    public void place(Field field, Player player) {  
        fieldPlayerMap.put(field, player);  
    }  
  
}
```

Why you should use first class collections

Consider collections as primitives; this way any behavior specification to your collection will be attracted to a single place

Filters will become part of the class

Joins or special rules applied to collection elements will be contained within the class

Changing the internal representation won't affect the clients that improve decoupling

Thank you 😊