

# JAVA EVERYWHERE AGAIN

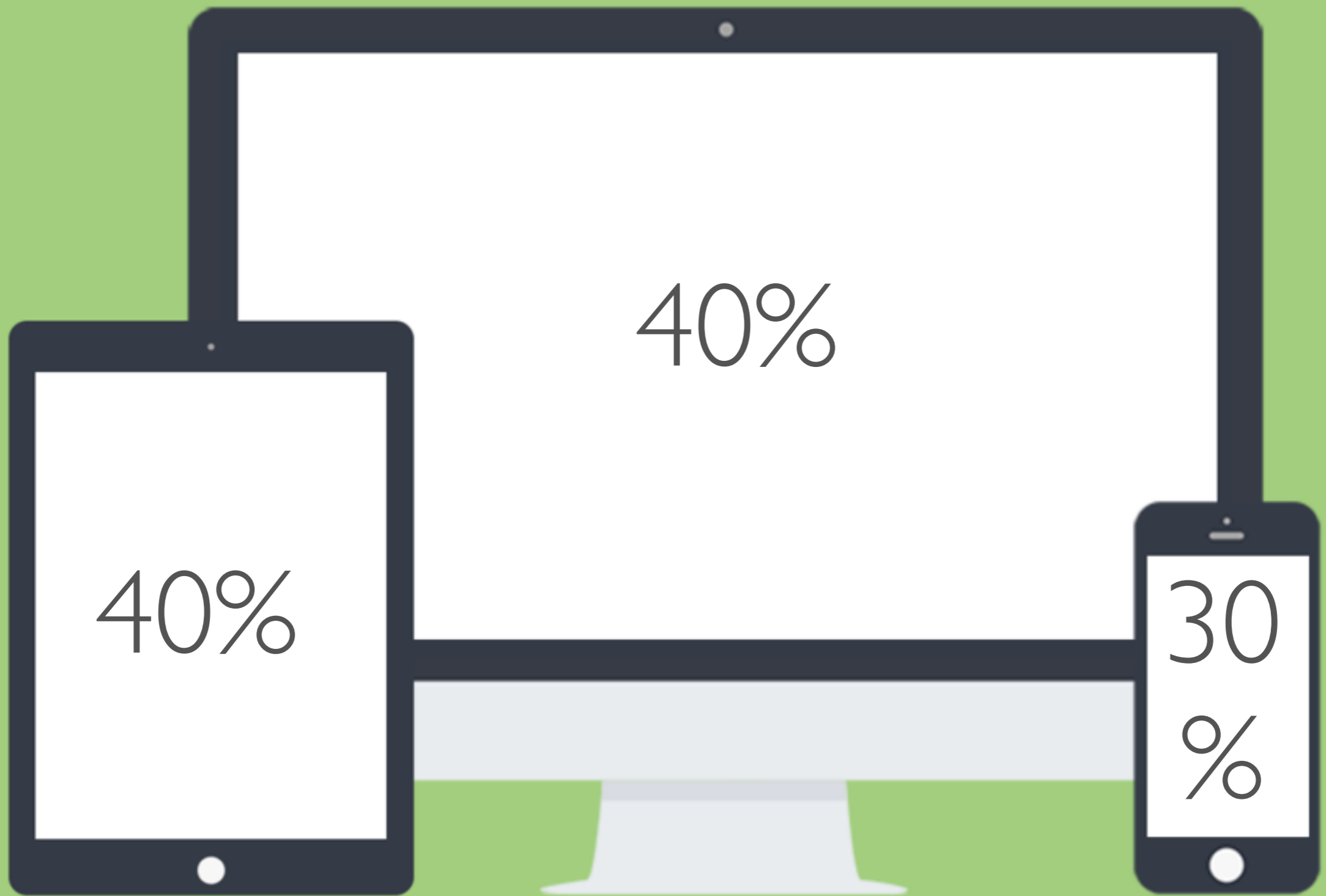
with DukeScript

# ANTON EPPLE

- Java Champion
- NetBeans Dream Team
- CEO Dukehoff GmbH
- @monacotoni



# ABOUT THIS LAB





**BINDING**



**BINDING**









@Model

# @Model

```
@Model(className = "Person",properties = {  
    @Property(name = "name", type = String.class),  
    @Property(name = "age", type = int.class)  
})  
  
public class ViewModel {  
  
}
```

# Generates:

```
public final class Person implements Cloneable {  
    public java.lang.String getName () {  
        //..  
    }  
    public void setName (java.lang.String v) {  
        //..  
    }  
    public int getAge() {  
        //..  
    }  
    public void setAge(int v) {  
        //..  
    }  
}
```

# Usage:

```
public static void onPageLoad() throws Exception {  
    Person person = new Person("Bob", 122);  
    person.setName("Robert");  
    person.setAge(person.getAge() + 1);  
    person.applyBindings();  
}
```

# Usage:

```
public static void onPageLoad() throws Exception {  
    Person person = new Person("Bob", 122);  
    person.setName("Robert");  
    person.setAge(person.getAge() + 1);  
    person.applyBindings();  
}
```

# person.applyBindings():

```
<html>
  <head>
    <title>PersonView</title>
  </head>
  <body>
    <div>
      <span data-bind="text: name"/> is
      <span data-bind="text: age"/> years old.
    </div>
  </body>
</html>
```

Robert is 123 years old.

<http://dew.apidesign.org/dew/#8d1cddc66f2a1712447b>



Robert is alone.

# Nested Model:

```
@Model(className = "PersonList", properties = {  
    @Property(name = "persons", type = Person.class,  
  
})  
public class ViewModel {  
    //..  
}
```

# Arrays:

```
@Model(className = "PersonList", properties = {  
    @Property(name = "persons", type = Person.class,  
              array = true)  
})  
public class ViewModel {  
    //..  
}
```

# Arrays:

```
public static void onPageLoad() throws Exception {  
    PersonList personList = new PersonList(new Person("Bob", 123));  
    personList.getPersons().add(new Person("Bill", 132));  
    personList.getPersons().add(new Person("Peter", 111));  
    personList.applyBindings();  
}
```

# person.applyBindings():

```
<html>
  <head>
    <title>PersonView</title>
  </head>
  <body>
    <div>
      <span data-bind="text: name"/> is
      <span data-bind="text: age"/> years old.
    </div>
  </body>
</html>
```

# person.applyBindings():

```
<html>
  <head>
    <title>PersonView</title>
  </head>
  <body>
    <div data-bind=„foreach: persons“>
      <span data-bind="text: name"/> is
      <span data-bind="text: age"/> years old.
    </div>
  </body>
</html>
```

Bob is 123 years old. Bill is 132 years old. Peter is 111 years old.

<http://dew.apidesign.org/dew/#4d95745cfd620eb7a919>

# Change the Design:

```
<html>  
  <head>  
    <title>PersonView</title>  
  </head>  
  <body>  
    <div data-bind=„foreach: persons“>  
      <span data-bind="text: name"/> is  
      <span data-bind="text: age"/> years old.  
    </div>  
  </body>  
</html>
```



# Change the Design:

```
<html>
  <head>
    <title>PersonView</title>
  </head>
  <body>
    <ol data-bind=„foreach: persons“>
      <li>
        <span data-bind="text: name"/> is
        <span data-bind="text: age"/> years old.
      </li>
    </ol>
  </body>
</html>
```

1. Bob is 123 years old.
2. Bill is 132 years old.
3. Peter is 111 years old.

<http://dew.apidesign.org/dew/#609b010d36f431ba1aaf>

# Function:

```
@Model(className = "Person", properties = {  
    @Property(name = "name", type = String.class),  
    @Property(name = "age", type = int.class)  
})
```

```
public static class PersonModel {
```

```
    @Function
```

```
    public static void increaseAge(Person person){  
        person.setAge(person.getAge()+1);  
    }
```

```
}
```

# Call a function:

```
<html>
  <head>
    <title>PersonView</title>
  </head>
  <body>
    <ol data-bind=„,foreach: persons“>
      <li>
        <span data-bind=„text: name“/> is
        <span data-bind=„text: age, click: increaseAge“/> years old.
      </li>
    </ol>
  </body>
</html>
```

**BINDING**

**databind**="<name>: <property> [...]"

# Style and Content

`databind="visible: showing"`

`databind="text: message"`

`databind="css: {'alarm': income() < 2}"`

`databind="style: {'color': temperature() < 0?  
                  'blue': 'red'}"`

`databind="attr: {href: url, title: details}"`



Responding to Input

`data-bind="click: increaseAge"`

`data-bind="event: { mouseover: enableDetails,  
mouseout: disableDetails}"`

`data-bind="enable: rotating"`

`data-bind="disable: rotating"`

`data-bind="value: property"`

`data-bind="textinput: text"`

`data-bind="hasFocus: isSelected"`

`data-bind="checked: wantsSpam"`

`data-bind="options: array"`

# Controlling Context

```
data-bind="foreach: persons"  
  data-bind="with: address"  
data-bind="click: $parent.remove"
```

## Lab Guide

<https://leanpub.com/hol3962>

<https://dukescript.com/javaone/hol3962.pdf>