

Von der Aufgabe zum Code

Javakurs 2011

Jörg Ferdinand

Freitagsrunde - TU Berlin

21. März 2011

4!

Der Datentyp char

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

```
1 public static void main(String[] args) {  
2     char zeichen = 'J';  
3     System.out.println("ingegebenes Zeichen:"  
4         + zeichen);  
5     zeichen = 'a';  
6 }
```

Der Datentyp char

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

```
1 public static void main(String[] args) {  
2     char zeichen = 'J';  
3     System.out.println("eingegebenes Zeichen:"  
4         + zeichen);  
5     zeichen = 'a';  
6 }
```

- ▶ char ist ein einzelnes Zeichen
- ▶ nicht nur Buchstaben
- ▶ Achtung - einzelne Hochkomma

Die Java API

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte



1

¹ <http://www.flickr.com/photos/striatic/729822/>

Die Java API

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte



-  **Alles**
-  **Bilder**
-  **Videos**
-  **News**

java api 6

java api 6

java api 6.0

java api 6 **download**

java api 6 **string**

java api 6 **file**

Die Java API

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

► [Java Platform SE 6](#) 🔍 - [[Diese Seite übersetzen](#)]

Java™ Platform, Standard Edition 6. API Specification. This document is the API specification for version 6 of the Java™ Platform, Standard Edition. ...
download.oracle.com/docs/cd/E17409_01/javase/6/.../api/ - Im Cache

Classes

[Java.awt.im.spi \(Java Platform SE 6\)](#)

[Java.awt.print](#)

[Java.beans](#)

[Java.beans.beancontext](#)

[Java.lang](#)

[Java.io \(Java Platform SE 6\)](#)

[Java.lang.management](#)

[Weitere Ergebnisse von oracle.com »](#)



4!

Die Java API

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

The screenshot shows the Java™ Platform, Standard Edition 6 API Specification page. The left sidebar lists 'All Classes' with a search bar and a list of packages including java.applet, java.awt, java.awt.color, java.awt.dnd, java.awt.event, java.awt.geom, java.awt.image, java.awt.print, java.io, java.lang, java.math, java.net, java.nio, java.rmi, java.security, java.sql, java.util, java.util.concurrent, java.util.logging, java.util.regex, java.util.zip, javax.swing, javax.swing.plaf, javax.swing.text, javax.swing.text.html, javax.swing.text.html.parser, javax.swing.undo, javax.xml, javax.xml.parsers, javax.xml.transform, javax.xml.transform.dom, javax.xml.transform.sax, javax.xml.xpath, and org.xml.sax. The main content area is titled 'Java™ Platform, Standard Edition 6 API Specification' and includes a table of packages with their descriptions.

Package	Description
java.applet	Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context.
java.awt	Contains all of the classes for creating user interfaces and for painting graphics and images.
java.awt.color	Provides classes for color spaces.
java.awt.dnd	Provides interfaces and classes for transferring data between and within applications.
java.awt.event	Provides interfaces and classes for defining and performing operations on objects related to two-dimensional geometry.
java.awt.geom	Provides classes and interfaces for defining and performing operations on objects related to two-dimensional geometry.
java.awt.image	Provides classes and interfaces for creating and modifying images.
java.awt.print	Provides classes and interfaces for producing rendering-independent images.
java.io	Provides classes and interfaces for defining and performing operations on objects related to two-dimensional geometry.
java.lang	Provides the Java 2D classes for defining and performing operations on objects related to two-dimensional geometry.
java.math	Provides interfaces that enable the development of user methods that can be used with any Java runtime environment.
java.net	Provides classes for creating and modifying images.
java.nio	Provides classes and interfaces for producing rendering-independent images.
java.rmi	Provides classes and interfaces for a general-purpose API.
java.security	Contains classes related to developing beans — components based on the JavaBeans™ architecture.
java.sql	Provides classes and interfaces relating to bean context.
java.util	Provides for system input and output through data streams, serialization, and the file system.
java.util.concurrent	Provides classes that are fundamental to the design of the Java programming language.
java.util.logging	Provides library support for the Java programming language annotation facility.
java.util.regex	Provides services that allow Java programming language agents to instrument programs running on the JVM.
java.util.zip	Provides the management interface for monitoring and management of the Java virtual machine as well as the operating system on which the Java virtual machine is running.
javax.swing	Provides reference-object classes, which support a limited degree of interaction with the garbage collector.
javax.swing.plaf	Provides classes and interfaces for obtaining reflective information about classes and objects.
javax.swing.text	Provides classes for performing arbitrary-precision integer arithmetic (BigInteger) and arbitrary-precision decimal arithmetic (BigDecimal).
javax.swing.undo	Provides the classes for implementing networking applications.
javax.xml	Defines buffers, which are containers for data, and provides an overview of the other NIO packages.

Die Java API

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Overview Package Class Use Tree Deprecated Index Help

Java™ Platform, Standard Edition 6
API Specification

This document is the API specification for version 6 of the Java™ Platform, Standard Edition.

See: [Description](#)

Packages	Description
java.applet	Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context.
java.awt	Contains all of the classes for creating user interfaces and for painting graphics and images.
java.awt.color	Provides classes for color spaces.
java.awt.datatransfer	Provides interfaces and classes for transferring data between and within applications.
java.awt.dnd	Drag and Drop is a direct manipulation gesture found in many Graphical User Interface systems that provides a mechanism to transfer information between two entities logically associated with presentation elements in the GUI.
java.awt.event	Provides interfaces and classes for dealing with different types of events fired by AWT components.
java.awt.font	Provides classes and interfaces relating to fonts.
java.awt.geom	Provides the Java 2D classes for defining and performing operations on objects related to two-dimensional geometry.
java.awt.image	Provides classes and interfaces for the input method framework.
java.awt.print	Provides interfaces that enable the development of print methods that can be used with any Java runtime environment.
java.awt.print.image	Provides classes for creating and modifying images.
java.awt.print.pdfable	Provides classes and interfaces for producing rendering-independent images.
java.awt.print.printer	Provides classes and interfaces for a general printing API.
java.beans	Contains classes related to developing beans — components based on the JavaBeans™ architecture.
java.beans.beancontext	Provides classes and interfaces relating to bean context.
java.io	Provides for system input and output through data streams, serialization, and the file system.
java.lang	Provides classes that are fundamental to the design of the Java programming language.
java.lang.annotation	Provides library support for the Java programming language annotation facility.
java.lang.instrument	Provides services that allow Java programming language agents to instrument programs running on the JVM.
java.lang.management	Provides the management interface for monitoring and management of the Java virtual machine as well as the operating system on which the Java virtual machine is running.
java.lang.ref	Provides reference-object classes, which support a limited degree of interaction with the garbage collector.
java.lang.reflect	Provides classes and interfaces for obtaining reflective information about classes and objects.
java.math	Provides classes for performing arbitrary-precision integer arithmetic (BigInteger) and arbitrary-precision decimal arithmetic (BigDecimal).
java.net	Provides the classes for implementing networking applications.
java.nio	Defines buffers, which are containers for data, and provides an overview of the other NIO packages.

Die Java API

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

The screenshot shows the Java API documentation for the `String` class. On the left, a navigation pane lists various classes and packages. The main content area displays the following information:

- Overview Package Class Use Tree Deprecated Index Help**
- OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP**
- SUMMARY METHOD (2) CONSTRUCTOR (1) METHOD (2) FIELD (2) CLASS (1) INTERFACE (1)**
- Package**
- Class String**
- java.lang.String**
- ↳ java.lang.Object**
- All Implemented Interfaces:**
 - [Serializable](#), [CharSequence](#), [Comparable<String>](#)
- public final class String**
- extends [Object](#)**
- implements [Serializable](#), [Comparable<String>](#), [CharSequence](#)**
- The `String` class represents character strings. All string literals in Java programs, such as "abc", are implemented as instances of this class.
- Strings are constant; their values cannot be changed after they are created. String buffers support mutable strings. Because String objects are immutable they can be shared. For example:

```
String s1 = "abc";
```
- It is equivalent to:

```
char detail[] = {'a', 'b', 'c'};
String s2 = new String(detail);
```
- Here are some more examples of how strings can be used:

```
System.out.println("ABC");
String s3 = "ABC";
System.out.println(s3 + 123);
String s4 = "ABC".substring(2,3);
String s5 = s4.substring(1, 2);
```
- The class `String` includes methods for examining individual characters of the sequence, for comparing strings, for searching strings, for extracting substrings, and for creating a copy of a string with all characters translated to uppercase or to lowercase. Case mapping is based on the Unicode Standard version specified by the [Character](#) class.
- The Java language provides special support for the string concatenation operator (+) and for conversion of other objects to strings. String concatenation is implemented through the `StringBuilder` (`StringBuffer`) class and its `append` method. String conversions are implemented through the method `toString`, defined by `Object` and inherited by all classes in Java. For additional information on string concatenation and conversion, see [Gosling, Joy, and Steele, The Java Language Specification](#).
- Unless otherwise noted, passing a null argument to a constructor or method in this class will cause a [NullPointerException](#) to be thrown.
- A `String` represents a string in the UTF-16 format in which supplementary characters are represented by surrogate pairs (see the section [Unicode Character Representations](#) in the `Character` class for more information). Index values refer to `char` code units, so a supplementary character uses two positions in a `String`.
- The `String` class provides methods for dealing with Unicode code points (i.e., characters), in addition to those for dealing with Unicode code units (i.e., `char` values).
- Since:** JDK 1.0

Inhalt

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

1 Was könnt ihr und wo wollen wir hin?



4!

Inhalt

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

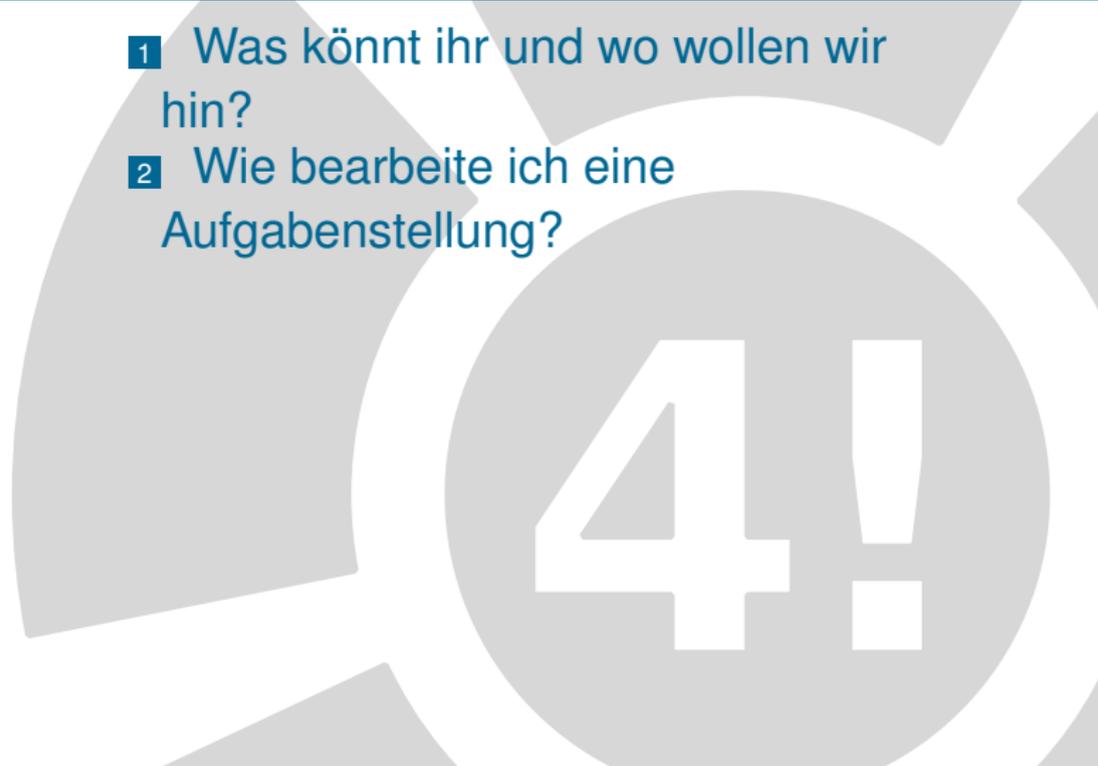
Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

- 1 Was könnt ihr und wo wollen wir hin?
- 2 Wie bearbeite ich eine Aufgabenstellung?

A large, stylized graphic of a gear with a '4!' inside it. The gear is light gray and has several teeth. The number '4' and the exclamation mark '!' are white and are positioned in the center of the gear. The background is a dark blue gradient.

Inhalt

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

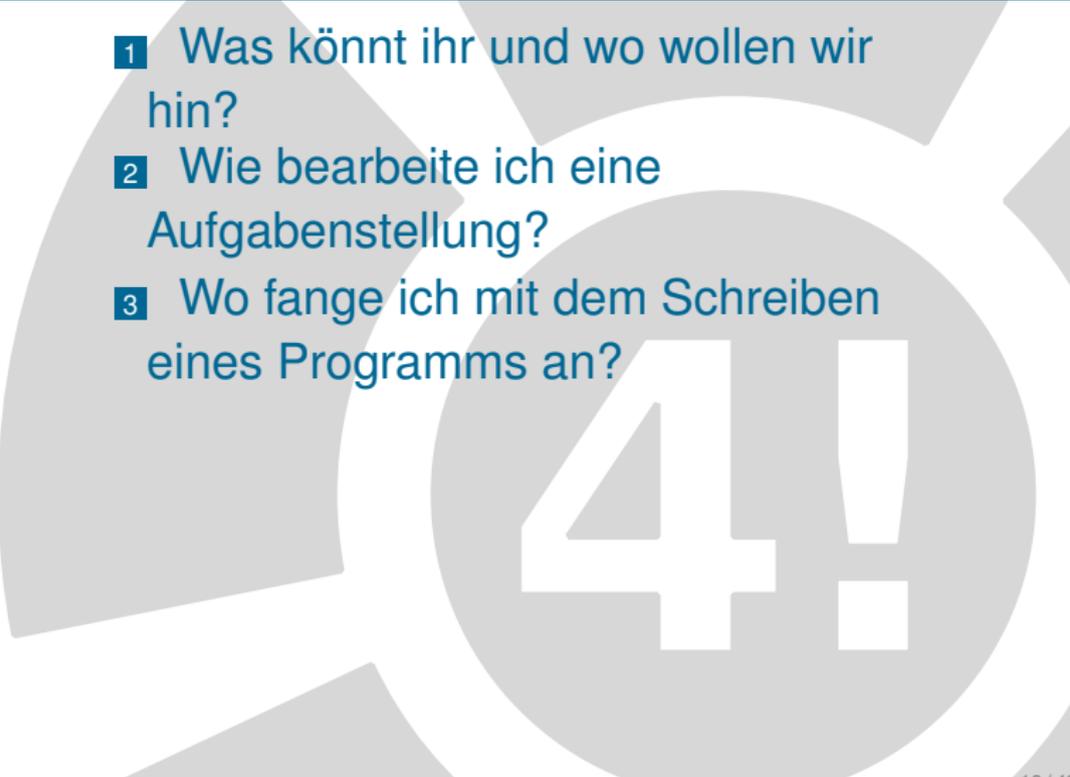
Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

- 1 Was könnt ihr und wo wollen wir hin?
- 2 Wie bearbeite ich eine Aufgabenstellung?
- 3 Wo fange ich mit dem Schreiben eines Programms an?



4!

Inhalt

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

- 1 Was könnt ihr und wo wollen wir hin?
- 2 Wie bearbeite ich eine Aufgabenstellung?
- 3 Wo fange ich mit dem Schreiben eines Programms an?
- 4 Wie behalte ich den Überblick in meinem Code?

Inhalt

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

- 1 Was könnt ihr und wo wollen wir hin?
- 2 Wie bearbeite ich eine Aufgabenstellung?
- 3 Wo fange ich mit dem Schreiben eines Programms an?
- 4 Wie behalte ich den Überblick in meinem Code?
- 5 Warum ist cooler Code schlechter Code?

Inhalt

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

- 1 Was könnt ihr und wo wollen wir hin?
- 2 Wie bearbeite ich eine Aufgabenstellung?
- 3 Wo fange ich mit dem Schreiben eines Programms an?
- 4 Wie behalte ich den Überblick in meinem Code?
- 5 Warum ist cooler Code schlechter Code?
- 6 Abschlussworte

Was ihr bisher könnt

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte



4!

Was ihr bisher könnt

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was ihr bisher könnt

Was ihr bisher könnt

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was ihr bisher könnt

- ▶ Umgang mit Variablen
- ▶ Ausgabe
- ▶ Fallunterscheidungen
- ▶ Schleifen
- ▶ Arrays
- ▶ Methoden
- ▶ API nutzen

Was ihr bisher könnt

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was ihr bisher könnt

- ▶ Umgang mit Variablen
- ▶ Ausgabe
- ▶ Fallunterscheidungen
- ▶ Schleifen
- ▶ Arrays
- ▶ Methoden
- ▶ API nutzen

⇒ alles eher Faktenwissen

Was nun?

Von der Aufgabe zum
Code

Jörg Ferdinand

**Was könnt ihr und wo
wollen wir hin?**

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

A large, stylized graphic of a gear or sun with a central circle containing the number '4!'. The gear is composed of several segments, and the central circle is a solid light gray. The number '4!' is white and centered within the circle.

4!

Was nun?

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

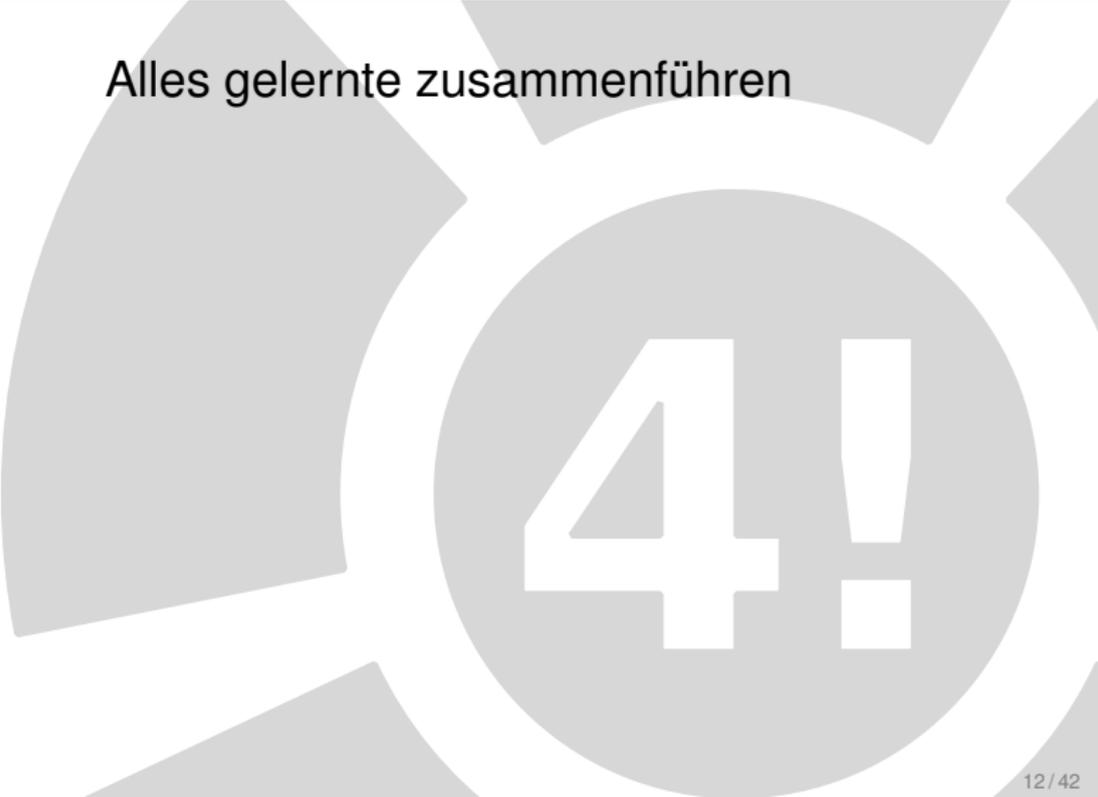
Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Alles gelernte zusammenführen



4!

Was nun?

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Alles gelernte zusammenführen

Im Detail

Was nun?

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Alles gelernte zusammenführen

Im Detail

- ▶ Wie beginnt man mit einer Aufgabe?

Was nun?

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Alles gelernte zusammenführen

Im Detail

- ▶ Wie beginnt man mit einer Aufgabe?
- ▶ Wie teilt man sie geschickt auf?

Was nun?

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Alles gelernte zusammenführen

Im Detail

- ▶ Wie beginnt man mit einer Aufgabe?
- ▶ Wie teilt man sie geschickt auf?
- ▶ Wie löst man die einzelnen Teile

Was nun?

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Alles gelernt zusammenführen

Im Detail

- ▶ Wie beginnt man mit einer Aufgabe?
- ▶ Wie teilt man sie geschickt auf?
- ▶ Wie löst man die einzelnen Teile
- ▶ Wie setze ich das Ganze in Java um?

Beispielaufgabenstellung

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

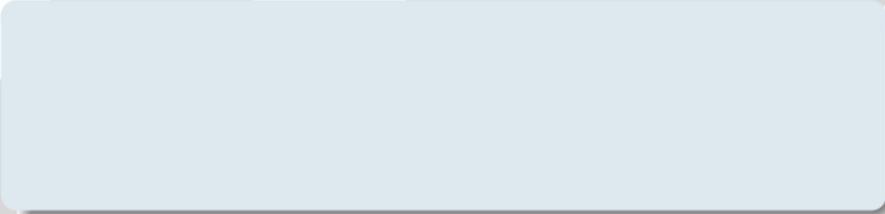
Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte



4!

Beispielaufgabenstellung

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Chef:

4!

Beispielaufgabenstellung

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Chef:

“Ich brauche ein Programm um Hangman zu spielen.”

4!

Eure Vorstellung:

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

A large, stylized graphic of a gear or sun with a central circle containing the number '4!'. The gear is composed of several segments, and the central circle is a solid light gray. The number '4!' is white and centered within the circle.

4!

Eure Vorstellung:

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte



Eure Vorstellung:

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

```
++-+++-----+--
|| //          |
||//          _|_
|+//         /o 0\
++          (| ( |)Menueschleife
+|          \_-\_/
||          _|_
|| )-----O| : |O----- (
||          | : |
||          | _ |
++          || ||
 /++\       _|| ||_
//||\ \    \_/_ \_/_
// || \ \
-----++-+++-----
```


Schritt 1 - Aufgabe verstehen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

A large, stylized graphic of a gear or sun with a central circle containing the text '4!'. The gear is composed of several segments, and the central circle is a solid light gray. The text '4!' is white and bold, centered within the circle.

4!

Schritt 1 - Aufgabe verstehen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was sind die Regeln von Hangman?

Schritt 1 - Aufgabe verstehen

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Was sind die Regeln von Hangman?

- ▶ Programm wählt zufällig ein Wort

Schritt 1 - Aufgabe verstehen

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Was sind die Regeln von Hangman?

- ▶ Programm wählt zufällig ein Wort
- ▶ Spieler gibt Buchstaben ein

Schritt 1 - Aufgabe verstehen

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Was sind die Regeln von Hangman?

- ▶ Programm wählt zufällig ein Wort
- ▶ Spieler gibt Buchstaben ein
 - ▶ Falscher Buchstabe führt zu einem Strich

Schritt 1 - Aufgabe verstehen

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Was sind die Regeln von Hangman?

- ▶ Programm wählt zufällig ein Wort
- ▶ Spieler gibt Buchstaben ein
 - ▶ Falscher Buchstabe führt zu einem Strich
 - ▶ Richtiger Buchstabe wird eingetragen

Schritt 1 - Aufgabe verstehen

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Was sind die Regeln von Hangman?

- ▶ Programm wählt zufällig ein Wort
- ▶ Spieler gibt Buchstaben ein
 - ▶ Falscher Buchstabe führt zu einem Strich
 - ▶ Richtiger Buchstabe wird eingetragen
- ▶ Spiel wird beendet wenn Wort erraten oder Hangman komplett

Schritt 2 - Ablauf nachvollziehen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte



4!

Schritt 2 - Ablauf nachvollziehen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Wie?

- ▶ Am besten grafisch (Stift und Papier)

Schritt 2 - Ablauf nachvollziehen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Wie?

- ▶ Am besten grafisch (Stift und Papier)
- ▶ Sorgt für mehr Durchblick

Schritt 2 - Ablauf nachvollziehen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Wie?

- ▶ Am besten grafisch (Stift und Papier)
- ▶ Sorgt für mehr Durchblick
- ▶ Ordnet den Ablauf des Programms

Schritt 2 - Ablauf nachvollziehen

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Wie?

- ▶ Am besten grafisch (Stift und Papier)
- ▶ Sorgt für mehr Durchblick
- ▶ Ordnet den Ablauf des Programms
- ▶ Definiert Teilbereiche des Spieles

Schritt 2 - Ablauf nachvollziehen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Wie?

- ▶ Am besten grafisch (Stift und Papier)
- ▶ Sorgt für mehr Durchblick
- ▶ Ordnet den Ablauf des Programms
- ▶ Definiert Teilbereiche des Spieles
 - ▶ Nützlich für Gruppenarbeiten

Schritt 2 - Ablauf nachvollziehen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Wie?

- ▶ Am besten grafisch (Stift und Papier)
- ▶ Sorgt für mehr Durchblick
- ▶ Ordnet den Ablauf des Programms
- ▶ Definiert Teilbereiche des Spieles
 - ▶ Nützlich für Gruppenarbeiten
 - ▶ Teile und Hersche Prinzip

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte



²<http://www.flickr.com/photos/lepke/276252512/>

Skizze

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

A large, stylized graphic of a gear or sun with a central circle containing the number '4!'. The gear is composed of several segments, and the central circle is a solid light gray. The number '4!' is white and centered within the circle.

4!

Skizze

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Menü

A large, stylized graphic of a gear with the number '4!' inside it. The gear is light gray and occupies the right half of the slide. The number '4!' is white and centered within a circular cutout in the gear. The background is a solid dark blue color.

4!

Skizze

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte



4!

Skizze

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

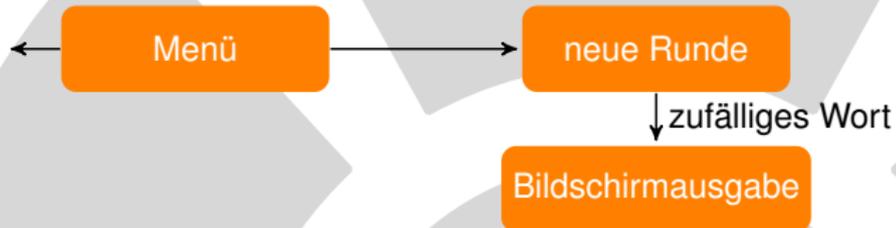
Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte



4!

Skizze

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

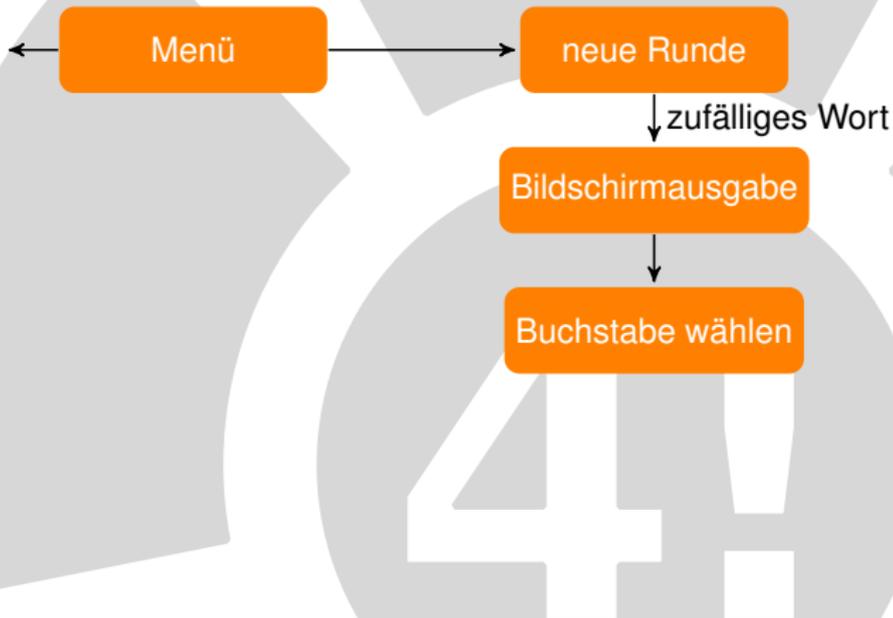
Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte



Skizze

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

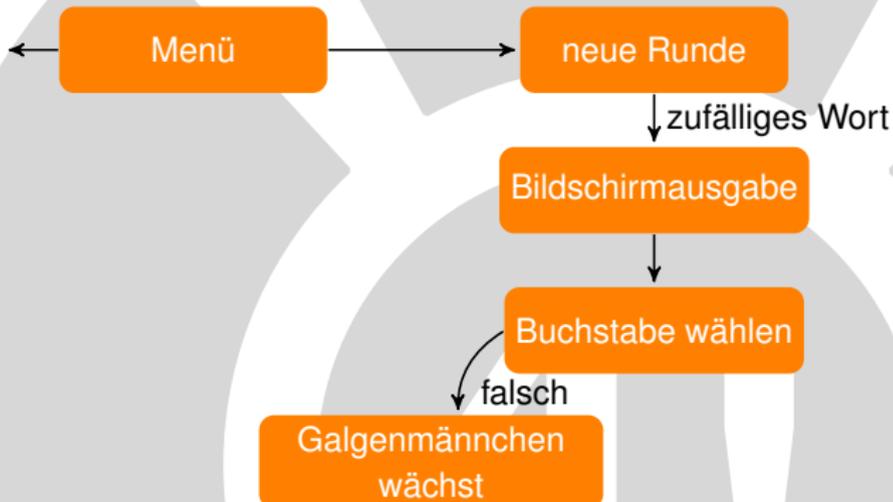
Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte



Skizze

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte



Skizze

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

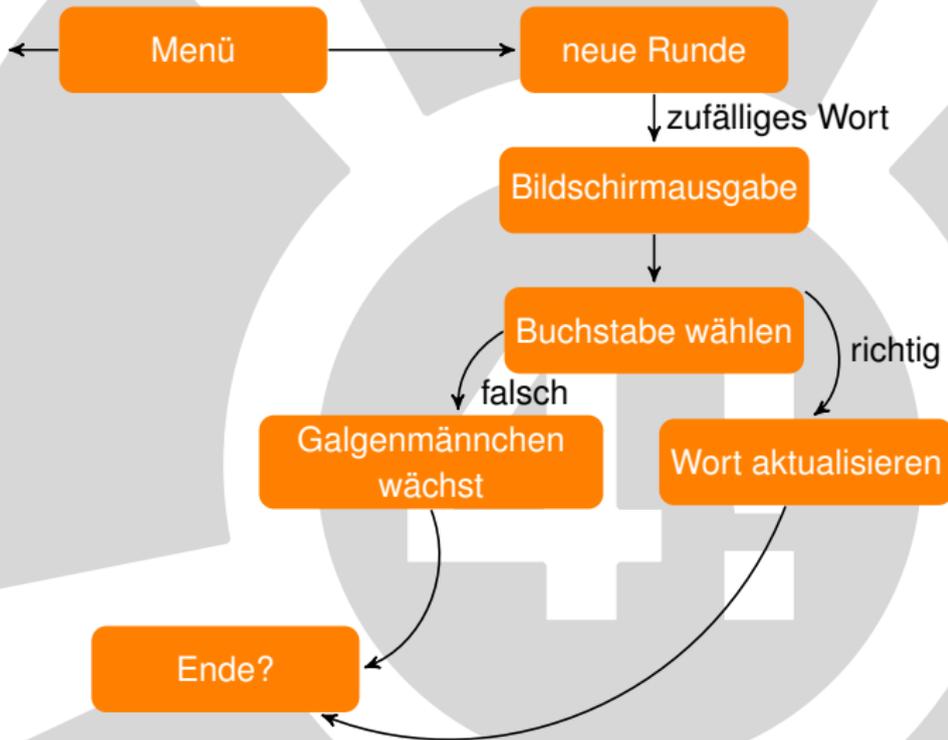
Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte



Skizze

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

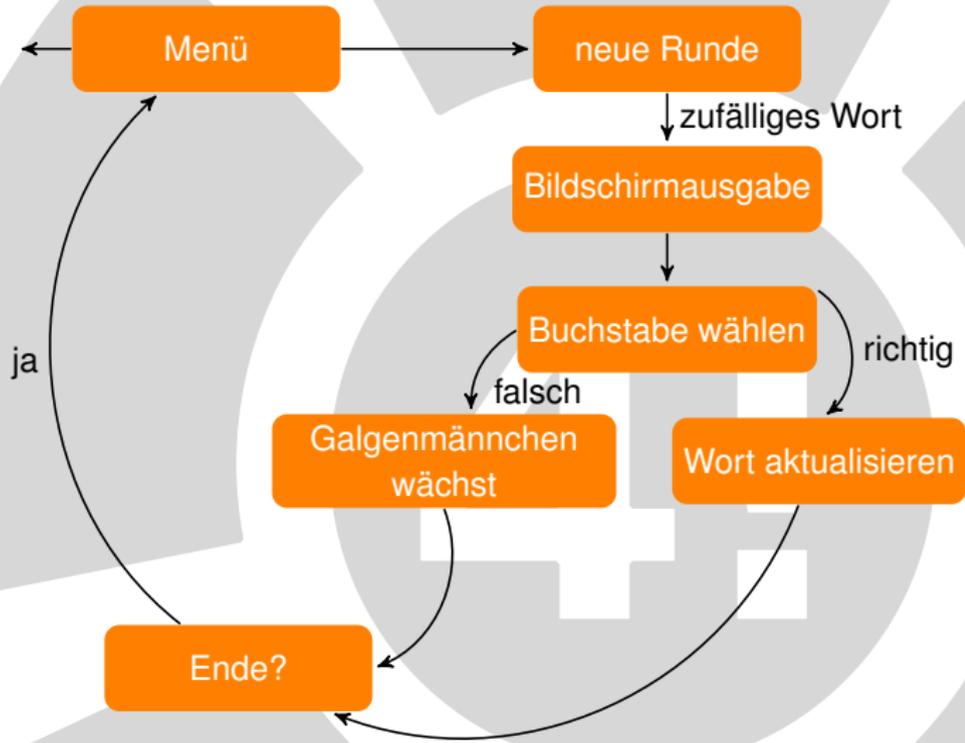
Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte



Skizze

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

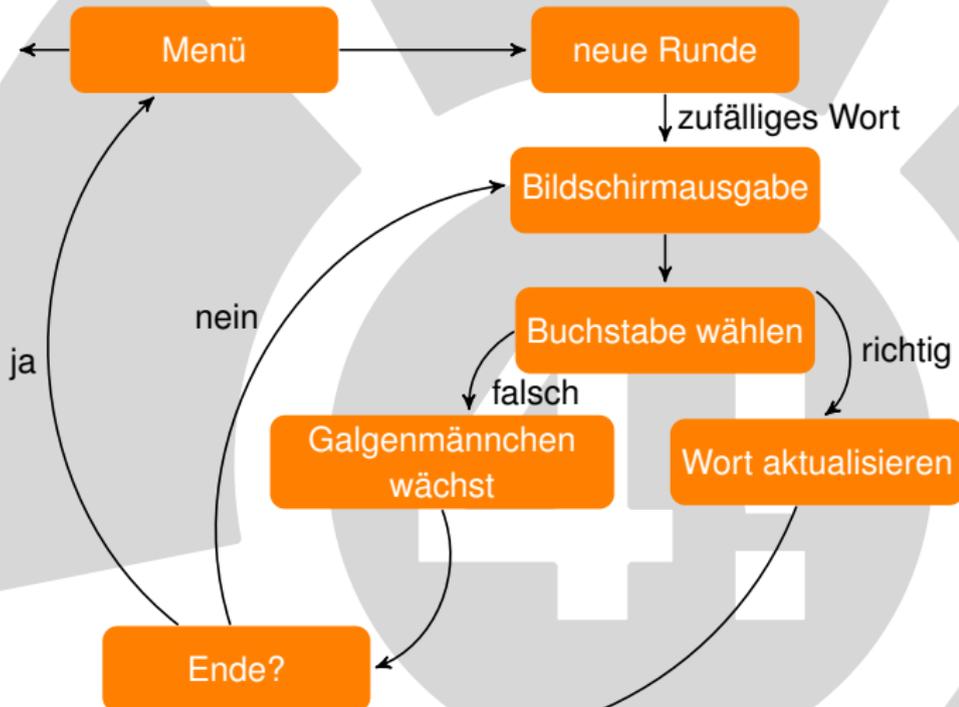
Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte



Schritt 3 - Skizze verfeinern

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte



4!

Schritt 3 - Skizze verfeinern

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Skizzen mit Ablauf erzeugen

Schritt 3 - Skizze verfeinern

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Skizzen mit Ablauf erzeugen

- ▶ bisher schlecht in Java
umzusetzen

Schritt 3 - Skizze verfeinern

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Skizzen mit Ablauf erzeugen

- ▶ bisher schlecht in Java umzusetzen
- ▶ linearerer Ablauf kann erzeugt werden

Schritt 3 - Skizze verfeinern

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Skizzen mit Ablauf erzeugen

- ▶ bisher schlecht in Java umzusetzen
- ▶ linearerer Ablauf kann erzeugt werden
- ▶ Teilprobleme können definiert werden

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

A large, stylized graphic of a gear or sun with a central circle containing the number '4!'. The gear is composed of several segments, and the central circle is a solid light gray. The number '4!' is white and centered within the circle.

4!

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

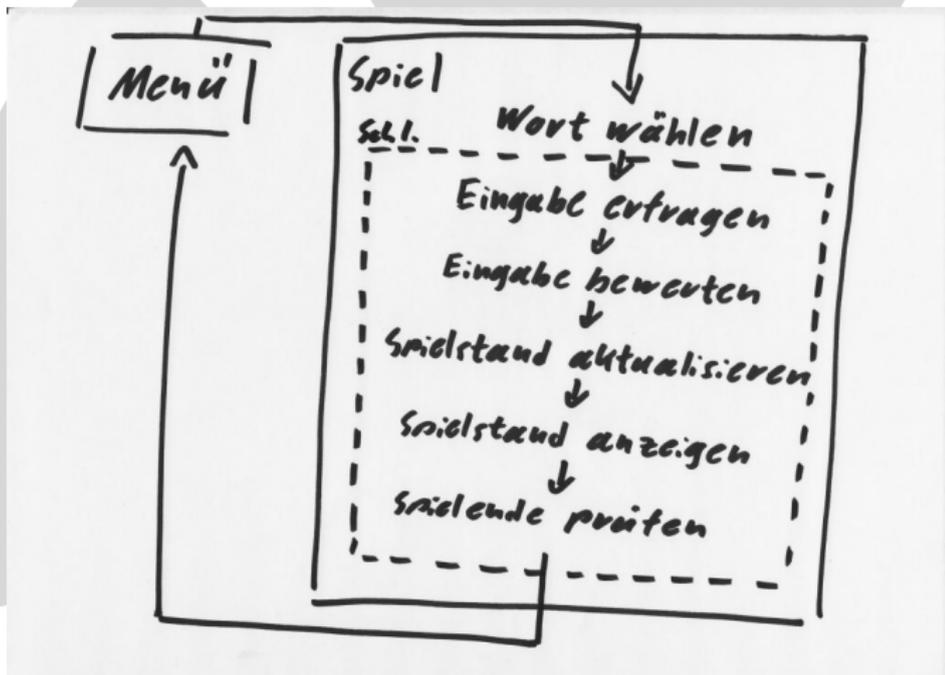
Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte



Schritt 4 - Teilen in Methoden

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Skizze in Methode umwandeln

Schritt 4 - Teilen in Methoden

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Skizze in Methode umwandeln

- ▶ Jedes Teilproblem ist eine Methode im Programm

Schritt 4 - Teilen in Methoden

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Skizze in Methode umwandeln

- ▶ Jedes Teilproblem ist eine Methode im Programm
- ▶ sinnvolle Namen geben!

Schritt 4 - Teilen in Methoden

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Skizze in Methode umwandeln

- ▶ Jedes Teilproblem ist eine Methode im Programm
- ▶ sinnvolle Namen geben!
- ▶ Was braucht die Methode?

Schritt 4 - Teilen in Methoden

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Skizze in Methode umwandeln

- ▶ Jedes Teilproblem ist eine Methode im Programm
- ▶ sinnvolle Namen geben!
- ▶ Was braucht die Methode?
 - ▶ Parameter?

Schritt 4 - Teilen in Methoden

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Skizze in Methode umwandeln

- ▶ Jedes Teilproblem ist eine Methode im Programm
- ▶ sinnvolle Namen geben!
- ▶ Was braucht die Methode?
 - ▶ Parameter?
 - ▶ Rückgabewert?

Schritt 4 - Teilen in Methoden

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte



4!

Schritt 4 - Teilen in Methoden

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Methoden

Schritt 4 - Teilen in Methoden

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Methoden

- ▶ `void menue ()`
- ▶ `boolean isGameWanted()`
- ▶ `String chooseWord ()`
- ▶ `char askForChar ()`
- ▶ `boolean isCharInWord (char character, String word)`
- ▶ `void updateGame (char guessedCharacter)`
- ▶ `void displayGame ()`
- ▶ `boolean isGameOver()`

Schritt 5 - Wiederholen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Kontrollieren

Schritt 5 - Wiederholen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Kontrollieren

- ▶ Inhalt der Methoden sofort ersichtlich?

Schritt 5 - Wiederholen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Kontrollieren

- ▶ Inhalt der Methoden sofort ersichtlich?
 - ▶ Ja, weiter mit nächster Methode

Schritt 5 - Wiederholen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Kontrollieren

- ▶ Inhalt der Methoden sofort ersichtlich?
 - ▶ Ja, weiter mit nächster Methode
 - ▶ Nein, neue Skizze

Schritt 5 - Wiederholen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Kontrollieren

- ▶ Inhalt der Methoden sofort ersichtlich?
 - ▶ Ja, weiter mit nächster Methode
 - ▶ Nein, neue Skizze
- ▶ Bei offensichtlichen Methoden
Aufschreiben der Eingabe- und
Rückgabeparameter

Code Einblick

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

**Wo fange ich mit dem
Schreiben eines
Programms an?**

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

**Viele Methoden, aber wie in Java
umsetzen?**

Code Einblick

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Viele Methoden, aber wie in Java umsetzen?

- ▶ Empfehlung 1:
TopDown–Verfahren

Code Einblick

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Viele Methoden, aber wie in Java umsetzen?

- ▶ Empfehlung 1:
TopDown–Verfahren
 - ▶ Beginn bei main hode

Code Einblick

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Viele Methoden, aber wie in Java umsetzen?

- ▶ Empfehlung 1:
TopDown–Verfahren
 - ▶ Beginn bei main hode
- ▶ Empfehlung 2: mit den Methoden
die man am besten testen kann

Code Einblick

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Viele Methoden, aber wie in Java umsetzen?

- ▶ Empfehlung 1:
TopDown–Verfahren
 - ▶ Beginn bei main hode
- ▶ Empfehlung 2: mit den Methoden
die man am besten testen kann
 - ▶ Beginn mit Ausgabefunktionen

Code Einblick

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Viele Methoden, aber wie in Java umsetzen?

- ▶ Empfehlung 1:
TopDown–Verfahren
 - ▶ Beginn bei main hode
- ▶ Empfehlung 2: mit den Methoden
die man am besten testen kann
 - ▶ Beginn mit Ausgabefunktionen
- ▶ Kommentare für noch nicht
existierende Methoden einfügen

Beispielcode

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

**Wo fange ich mit dem
Schreiben eines
Programms an?**

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

A large, stylized graphic of a gear or sun with a central circle containing the number '4!'. The gear is composed of several segments, and the central circle is a solid light gray. The number '4!' is white and prominently displayed in the center of the circle.

4!

Beispielcode

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

```
1 public static void main(String[] args) {  
2     // Menueschleife  
3         // neues Spiel ?  
4             // wenn nein, dann Ende  
5         // Wort waehlen  
6         // Spielschleife  
7             // Spielstand ausgeben  
8             // Buchstaben erfragen  
9             // Eingabe bewerten  
10            // Spielstand aktualisieren  
11            // Spielende pruefen  
12            // Ende Spielschleife  
13        // Ende Menueschleife  
14    }
```

Wie behalte ich den Überblick in meinem Code?

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte



4!

Wie behalte ich den Überblick in meinem Code?

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Überblick durch:

- ▶ Nun pro Kommentar eine neue Methode

Wie behalte ich den Überblick in meinem Code?

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Überblick durch:

- ▶ Nun pro Kommentar eine neue Methode
- ▶ Variablen, Kommentare sowie Ausgaben einfügen

Wie behalte ich den Überblick in meinem Code?

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Überblick durch:

- ▶ Nun pro Kommentar eine neue Methode
- ▶ Variablen, Kommentare sowie Ausgaben einfügen
- ▶ Sobald neue Methode fertig: Testdurchlauf

Testen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte



4!

Testen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Testen

- ▶ So früh wie möglich
- ▶ So oft wie möglich
- ▶ So gründlich wie möglich

Testen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Testen

- ▶ So früh wie möglich
- ▶ So oft wie möglich
- ▶ So gründlich wie möglich

Testen – Dummymethoden

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte



4!

Testen – Dummymethoden

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was sind Dummymethoden?

Testen – Dummymethoden

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Was sind Dummymethoden?

- ▶ Wir schreiben die Methoden mit Parametern und Rückgabety auf

Testen – Dummymethoden

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Was sind Dummymethoden?

- ▶ Wir schreiben die Methoden mit Parametern und Rückgabebetyp auf
- ▶ Jedoch erstmal ohne Funktion

Testen – Dummymethoden

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Was sind Dummymethoden?

- ▶ Wir schreiben die Methoden mit Parametern und Rückgabebetyp auf
- ▶ Jedoch erstmal ohne Funktion
- ▶ “Platzhalter”

Testen – Dummymethoden

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

```
1 // TODO: Befuell mich!  
2 public char askForChar(){  
3     // liefere festen Wert  
4     return 'a';  
5 }
```

Testen – Dummymethoden

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

```
1 // TODO: Befuell mich!  
2 public char askForChar(){  
3     // liefere festen Wert  
4     return 'a';  
5 }
```

```
1 // TODO: Befuell mich!  
2 public String chooseWord() {  
3     // liefere festes Wort  
4     return "Javakurs";  
5 }
```

Testen – println debug

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

A large, stylized graphic of a gear or sun with a central circle containing the text '4!'. The gear is composed of several segments, and the central circle is a solid light gray. The text '4!' is white and centered within the circle.

4!

Testen – println debug

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was ist println debug?

Testen – println debug

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was ist println debug?

- ▶ Im Normalfall wird man beim Testen Fehler finden

Testen – println debug

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was ist println debug?

- ▶ Im Normalfall wird man beim Testen Fehler finden
- ▶ “Woran liegt es?”

Testen – println debug

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was ist println debug?

- ▶ Im Normalfall wird man beim Testen Fehler finden
- ▶ “Woran liegt es?”
- ▶ Einfach “Zwischenergebnisse” anzeigen lassen

Beispiel – println debug

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte



4!

Beispiel – println debug

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

```
1 // Eingabe bewerten
2 boolean isCharInWord (char character,
3     String word){
4
5     boolean isInWord = false;
6     for(int i = 0 ; i < word.length() ; i++){
7         if (word.charAt(i)==character){
8             isInWord = true;
9         }
10    }
11
12    return isInWord;
13 }
```

Beispiel – println debug

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

**Wie behalte ich den
Überblick in meinem
Code?**

Warum ist cooler Code
schlechter Code?

Abschlussworte



4!

Beispiel – println debug

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

```
1 // Eingabe bewerten
2 boolean isCharInWord (char character,
3     String word){
4     System.out.println(character + word)
5     boolean isInWord = false;
6     for(int i = 0 ; i < word.length() ; i++){
7         if (word.charAt(i)==character){
8             isInWord = true;
9         }
10    }
11    System.out.println(isInWord);
12    return isInWord;
13 }
```

Beispiel schlechter Code

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

A large, stylized graphic of a gear or sun with a central circle containing the text '4!'. The gear is composed of several segments, and the central circle is a solid light gray. The text '4!' is white and bold, centered within the circle.

4!

Beispiel schlechter Code

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

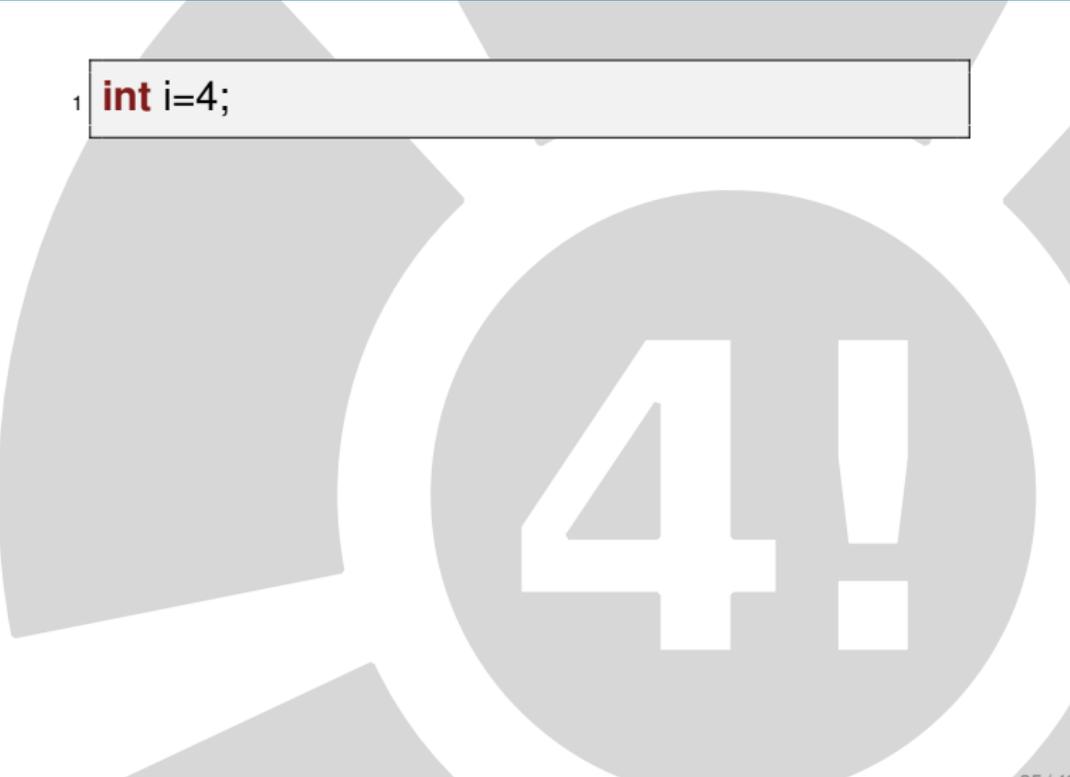
Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

```
1 int i=4;
```



4!

Beispiel schlechter Code

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

```
1 int i=4;
```

```
1 System.out.println( (i==4)? "i ist 4" : "i ist nicht  
4");
```

A large, light gray graphic of the number '4' followed by an exclamation mark, enclosed within a circle. The graphic is positioned in the lower right quadrant of the slide, partially overlapping the code blocks.

Beispiel schlechter Code

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

```
1 int i=4;
```

```
1 System.out.println( (i==4)? "i ist 4" : "i ist nicht  
4");
```

```
1 if (i==4){  
2     System.out.println("i ist 4");  
3 } else {  
4     System.out.println("i ist nicht 4");  
5 }
```

Wie sollte der Code aussehen?

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Folgende Sachen solltet Ihr beachten:

Wie sollte der Code aussehen?

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Folgende Sachen solltet Ihr beachten:

- ▶ Formatierungen

Wie sollte der Code aussehen?

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Folgende Sachen solltet Ihr beachten:

- ▶ Formatierungen
 - ▶ Einrückungen

Wie sollte der Code aussehen?

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Folgende Sachen solltet Ihr beachten:

- ▶ **Formatierungen**
 - ▶ Einrückungen
- ▶ **Kommentierung**

Wie sollte der Code aussehen?

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Folgende Sachen solltet Ihr beachten:

- ▶ Formatierungen
 - ▶ Einrückungen
- ▶ Kommentierung
- ▶ Sprechende Namen

Einrückung

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

A large, stylized graphic of a gear or sun with a central circle containing the number '4!'. The gear is composed of several segments, and the central circle is a solid light gray. The number '4!' is white and centered within the circle.

4!

Einrückung

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Erkennen von Teilbereichen wichtig bei:

4!

Einrückung

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Erkennen von Teilbereichen wichtig bei:

- ▶ Bedingungen

Einrückung

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Erkennen von Teilbereichen wichtig bei:

- ▶ Bedingungen
- ▶ Schleifen

Einrückung

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Erkennen von Teilbereichen wichtig bei:

- ▶ Bedingungen
- ▶ Schleifen
- ▶ Klassen

Einrückung

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Erkennen von Teilbereichen wichtig bei:

- ▶ Bedingungen
- ▶ Schleifen
- ▶ Klassen
- ▶ Methoden

Einrückung

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Erkennen von Teilbereichen wichtig bei:

- ▶ Bedingungen
- ▶ Schleifen
- ▶ Klassen
- ▶ Methoden

Einrückung

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Erkennen von Teilbereichen wichtig bei:

- ▶ Bedingungen
- ▶ Schleifen
- ▶ Klassen
- ▶ Methoden

```
1 if( Bedingung ) {  
2     // dann Zweig  
3 } else {  
4     // sonst Zweig  
5 }
```

Einrückung

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Erkennen von Teilbereichen wichtig bei:

- ▶ Bedingungen
- ▶ Schleifen
- ▶ Klassen
- ▶ Methoden

```
1 if( Bedingung ) {  
2     // dann Zweig  
3 } else {  
4     // sonst Zweig  
5 }
```

```
1 while( Bedingung ) {  
2     //  
3     // wiederhole...  
4     //  
5 }
```

Sprechende Namen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

A large, stylized graphic of a gear or sun with a central circle containing the number '4!'. The gear is composed of several segments, and the central circle is a solid light gray. The number '4!' is white and prominently displayed in the center of the circle.

4!

Sprechende Namen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was sind sprechende Namen?

Sprechende Namen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was sind sprechende Namen?

- ▶ Was steht in der Variablen?

Sprechende Namen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was sind sprechende Namen?

- ▶ Was steht in der Variablen?
- ▶ Was tut die Methode?

Sprechende Namen

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Was sind sprechende Namen?

- ▶ Was steht in der Variablen?
- ▶ Was tut die Methode?
- ▶ Kann ich aus dem Namen rauslesen, was sie tut?

Schönheit kostet nichts

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

A large, stylized graphic of a gear or sun with a central circle containing the number '4!'. The gear is composed of several segments, and the central circle is a solid light gray. The number '4!' is white and centered within the circle.

4!

Schönheit kostet nichts

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Schönheit von Code

Schönheit kostet nichts

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Schönheit von Code

- ▶ Wichtig: Code muss laufen

Schönheit kostet nichts

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Schönheit von Code

- ▶ Wichtig: Code muss laufen
- ▶ Optimierung macht der Compiler meist besser als ihr

Schönheit kostet nichts

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Schönheit von Code

- ▶ Wichtig: Code muss laufen
- ▶ Optimierung macht der Compiler meist besser als ihr
- ▶ Erst wenn Code viel zu langsam ist müsst ihr gucken

Schönheit kostet nichts

Von der Aufgabe zum Code

Jörg Ferdinand

Was könnt ihr und wo wollen wir hin?

Wie bearbeite ich eine Aufgabenstellung?

Wo fange ich mit dem Schreiben eines Programms an?

Wie behalte ich den Überblick in meinem Code?

Warum ist cooler Code schlechter Code?

Abschlussworte

Schönheit von Code

- ▶ Wichtig: Code muss laufen
- ▶ Optimierung macht der Compiler meist besser als ihr
- ▶ Erst wenn Code viel zu langsam ist müsst ihr gucken
 - ▶ Aufgabenstellung richtig gelesen?

Schönheit kostet nichts

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Schönheit von Code

- ▶ Wichtig: Code muss laufen
- ▶ Optimierung macht der Compiler meist besser als ihr
- ▶ Erst wenn Code viel zu langsam ist müsst ihr gucken
 - ▶ Aufgabenstellung richtig gelesen?
 - ▶ zu viele "Nice to haves"?(3D-Galgen?)

Ausblick

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was habt ihr gelernt?

Ausblick

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was habt ihr gelernt?

- ▶ API

Ausblick

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was habt ihr gelernt?

- ▶ API
- ▶ Wie zerlege ich eine Aufgabe in Teilprobleme

Ausblick

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was habt ihr gelernt?

- ▶ API
- ▶ Wie zerlege ich eine Aufgabe in Teilprobleme
- ▶ Wie teile ich die Teilprobleme

Ausblick

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was habt ihr gelernt?

- ▶ API
- ▶ Wie zerlege ich eine Aufgabe in Teilprobleme
- ▶ Wie teile ich die Teilprobleme
- ▶ Wie formuliere ich ein Teilproblem in Java

Ausblick

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Was habt ihr gelernt?

- ▶ API
- ▶ Wie zerlege ich eine Aufgabe in Teilprobleme
- ▶ Wie teile ich die Teilprobleme
- ▶ Wie formuliere ich ein Teilproblem in Java
- ▶ Was sollte ich beim Schreiben von Code beachten?

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Organisatorisches

- ▶ **jetzt: Feedback Abgabe**

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Organisatorisches

- ▶ jetzt: Feedback Abgabe
- ▶ dann: Übung im TEL 106/206

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Organisatorisches

- ▶ jetzt: Feedback Abgabe
- ▶ dann: Übung im TEL 106/206
- ▶ man muss nicht alle Aufgaben schaffen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Organisatorisches

- ▶ jetzt: Feedback Abgabe
- ▶ dann: Übung im TEL 106/206
- ▶ man muss nicht alle Aufgaben schaffen
- ▶ heute wieder ein zweiter Vortrag!

Fragen

Von der Aufgabe zum
Code

Jörg Ferdinand

Was könnt ihr und wo
wollen wir hin?

Wie bearbeite ich eine
Aufgabenstellung?

Wo fange ich mit dem
Schreiben eines
Programms an?

Wie behalte ich den
Überblick in meinem
Code?

Warum ist cooler Code
schlechter Code?

Abschlussworte

Fragen?

A large, stylized graphic of a gear or sun with a central circle containing the text '4!'. The gear is composed of several segments, and the central circle is a solid light gray. The text '4!' is white and bold, centered within the circle. The background of the slide is a dark blue gradient.