

16. Dokumentationskommentare

Dokumentationskommentare



Spezielle Art von Kommentaren

```
/** ... */
```

können vor die Deklarationen von Klassen, Methoden, Feldern gesetzt werden. Werkzeug *javadoc* erzeugt daraus Dokumentation in HTML

```
/** A stack of integers.

This is a FIFO data structure storing integers in a stack-like way. */
public class Stack {

    /** The elements in the stack. */
    private int[] data;
    ...

    /** Push an integer on the stack. 

    If the stack is full an error is reported and the program stops. */
    public void push (int x) {

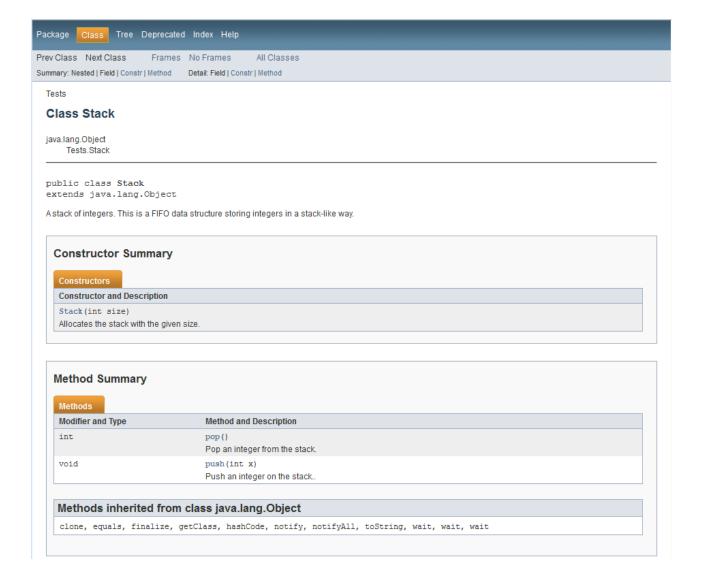
        ...
        Rest wird in Langdoku
        übernommen
}
```

Vorteil

Dokumentation und Code stehen an derselben Stelle Höhere Wahrscheinlichkeit, dass die Dokumentation aktuell gehalten wird

Erzeugte HTML-Datei (Stack.html)





Tags



Zusatzinformationen in Dokumentationskommentaren

- Müssen am Zeilenanfang stehen
- Generelle Syntax: @tagname parameters
- Gehen bis zum nächsten Tag oder bis zum Kommentarende

Klassen-Tags

@author text
Name des Autors der Klasse

@version text
Beschreibt Version der Klasse (in beliebigem Format)

Methoden-Tags

@param name text
Beschreibt Name und Bedeutung des Parameters

@return text Beschreibt Bedeutung des Rückgabewerts einer Funktion

@exception class text Gibt eventuell ausgelöste Exception(klasse) samt Bedeutung an

Beispiel für Tags



```
/** A stack of integers.
  This is a FIFO data structure storing integers in a stack-like way.
  @author Hanspeter Moessenboeck
  @version 2.0 */
public class Stack {
  /** Push an integer on the stack.
     If the stack is full an exception is thrown.
     @param x The value to be pushed.
     @exception Exception Thrown if the stack is full. */
  public void push (int x) throws Exception {
  /** Pops an integer from the stack.
     If the stack is empty an exception is thrown.
     @return The value popped from the stack.
     @exception Exception Thrown if the stack is empty. */
  public int pop() throws Exception {
```

Method Detail

push

```
public void push (int x)
            throws java.lang.Exception
Push an integer on the stack. If the stack is full an exception is thrown.
```

Parameters:

```
x - The value to be pushed .
```

java.lang.Exception - Thrown if the stack is full.

pop

```
public int pop()
        throws java.lang.Exception
```

Pops an integer from the stack. If the stack is empty an exception is thrown.

Returns:

The value popped from the stack.

java.lang.Exception - Thrown if the stack is empty.

javadoc



Aufrufsyntax

javadoc [options] {filename | packagename}

Optionen

• -public zeigt nur public-Deklarationen

• -private zeigt public- und private-Deklarationen

• -d path Zielverzeichnis, in das HTML-Dateien gespeichert werden

• ...

Beispiel

javadoc -public Stack.java

erzeugt:

Stack.html diverse andere Übersichtsdateien

Vollständige Dokumentation von javadoc

https://docs.oracle.com/javase/9/