# Klassendiagramm

# Sequenzdiagramm

# Speicherdiagramm

# Datentypen

|  |  |  |
| --- | --- | --- |
| **Datentyp** | **Grösse** | **Beschreibung** |
| boolean | - | Boolescher Wahrheitswert |
| char | 16bit | Unicode Zeichen |
| byte | 8bit | Zweierkomplement-Wert |
| short | 16bit | Zweierkomplement-Wert |
| int | 32bit | Zweierkomplement-Wert |
| long | 64bit | Zweierkomplement-Wert |
| float | 32bit | Gleitkommazahl |
| double | 64bit | Gleitkommazahl |

# KeyListener

addKeyListener(new MyKeyListener());

public MyKeyListener {

public void keyPressed(KeyCode key) {

return key.getKeyChar();

}

}

# Strings vergleichen

String example = „Hello“;

String test = „Hello“;

example == test; // false;

example.equals(test) ; // true ;

# Klassen

public class Klasse {

public static final double PI = 3.141592; // Konstante, wegen final (Klassenvariable)  
public static int ID = 42; // Klassenvariable  
private int age; // Instanzvariable  
private String name = « Klasse » ; // Instanzvariable

public Klasse(int alter) { // Konstruktor

this.name = name ; // Assignment

}

public int getDoubleAge() { // Methode

int factor = 2 ; // Lokale variable

return factor \* age ;

}

}

# Vector / ArrayList

List<String> data = new ArrayList<String>();

# Testing

# Projekt