

Programmierung unter GNU/Linux für Einsteiger

Edgar 'Fast Edi' Hoffmann

Community FreieSoftwareOG

kontakt@freiesoftwareog.org

7. September 2016

Programmierung

Begriffserklärung

Programmierung

Begriffserklärung

Programmierung (von griechisch próγραμμα „Vorschrift“) bezeichnet die Tätigkeit, Computerprogramme zu erstellen.

Programmierung

Begriffserklärung

Programmierung (von griechisch próγραμμα „Vorschrift“) bezeichnet die Tätigkeit, Computerprogramme zu erstellen.

Dies umfasst vor Allem die Umsetzung (Implementierung) des Softwareentwurfs in Quellcode sowie – je nach Programmiersprache – das Übersetzen des Quellcodes in die Maschinensprache, meist unter Verwendung eines Compilers.

Programmierung

Konzepte der Softwareentwicklung

Programmierung

Konzepte der Softwareentwicklung

Programme werden unter Verwendung von Programmiersprachen formuliert („kodiert“). In eine solche Sprache „übersetzt“ der Programmierer die (z. B. im Pflichtenheft) vorgegebenen Anforderungen und Algorithmen.

Programmierung

Konzepte der Softwareentwicklung

Programme werden unter Verwendung von Programmiersprachen formuliert („kodiert“). In eine solche Sprache „übersetzt“ der Programmierer die (z. B. im Pflichtenheft) vorgegebenen Anforderungen und Algorithmen.

Zunehmend wird er dabei durch Codegeneratoren unterstützt, die zumindest Teile des Programmcodes auf Basis von Modellen (die im Entwurf entstanden sind) automatisch erzeugen.

Programmierung

Konzepte der Softwareentwicklung

Programme werden unter Verwendung von Programmiersprachen formuliert („kodiert“). In eine solche Sprache „übersetzt“ der Programmierer die (z. B. im Pflichtenheft) vorgegebenen Anforderungen und Algorithmen.

Zunehmend wird er dabei durch Codegeneratoren unterstützt, die zumindest Teile des Programmcodes auf Basis von Modellen (die im Entwurf entstanden sind) automatisch erzeugen.

Zu den weiteren Aufgaben von Programmierern zählen zum Beispiel das Testen (Entwicklertest) seines Programms, das Erstellen der Softwaredokumentation usw.

Programmierung

Entwicklungsmodelle

Programmierung

Entwicklungsmodelle

- Wasserfallmodell

Programmierung

Entwicklungsmodelle

- Wasserfallmodell
- Extreme Programming

Programmierung Entwicklungsmodelle

- Wasserfallmodell
- Extreme Programming
- Scrum

Programmierung

Entwicklungsmodelle

- Wasserfallmodell
- Extreme Programming
- Scrum
- Prototyping

Programmierung Entwicklungsmodelle

- Wasserfallmodell
- Extreme Programming
- Scrum
- Prototyping
- Testgetriebene Softwareentwicklung

Programmierung

Zusätzliche Werkzeuge und Helfer

Programmierung

Zusätzliche Werkzeuge und Helfer

- Struktogramm bzw. Nassi-Shneiderman

Programmierung

Zusätzliche Werkzeuge und Helfer

- Struktogramm bzw. Nassi-Shneiderman
- PAP

Programmierung

Zusätzliche Werkzeuge und Helfer

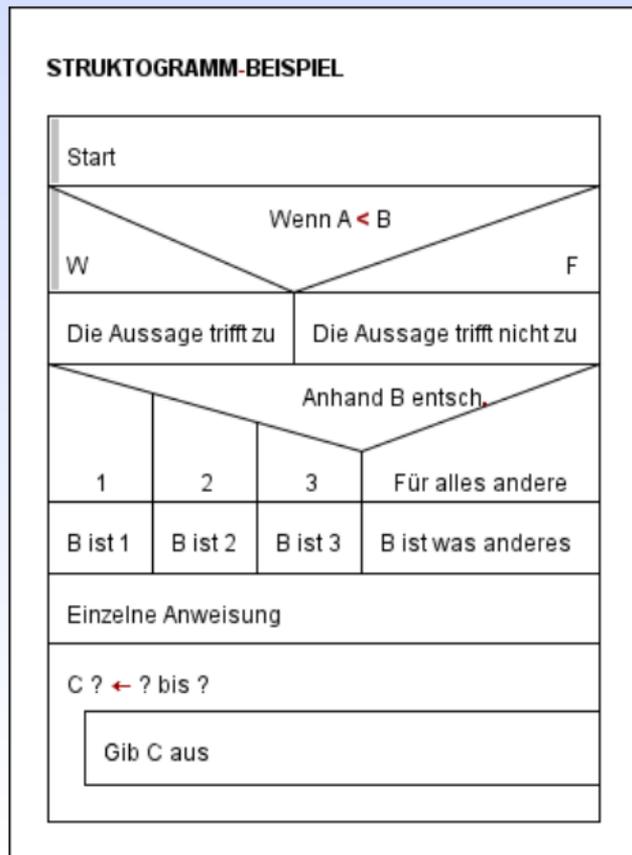
- Struktogramm bzw. Nassi-Shneiderman
- PAP
- GUI-Prototyping bzw. Mockups

Programmierung

Beispiel eines Nassi-Shneiderman-Diagramms

Programmierung

Beispiel eines Nassi-Shneiderman-Diagramms



Programmierung

Entwicklungsumgebungen (IDEs)

Programmierung Entwicklungsumgebungen (IDEs)

- Früher machte man's auf der Konsole

Programmierung Entwicklungsumgebungen (IDEs)

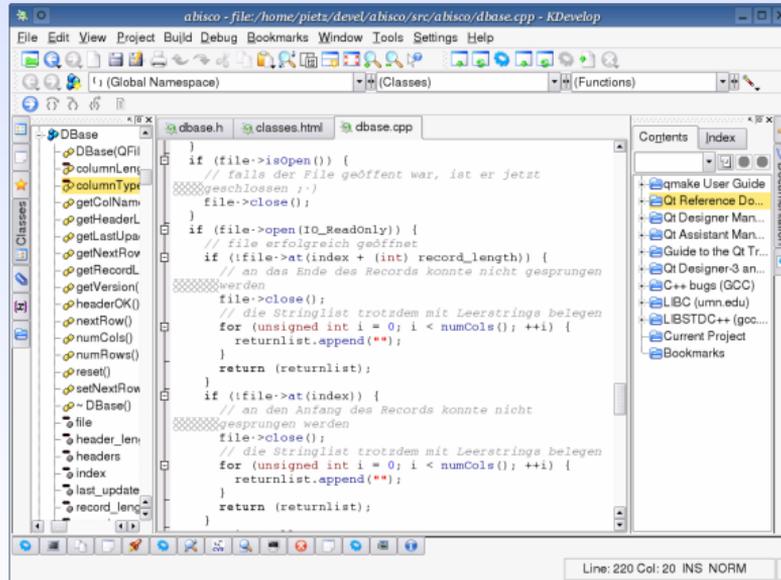
- Früher machte man's auf der Konsole
- Heute gibt es komfortable IDEs

Programmierung Entwicklungsumgebungen (IDEs)

- Früher machte man's auf der Konsole
- Heute gibt es komfortable IDEs

Programmierung Entwicklungsumgebungen (IDEs)

- Früher machte man's auf der Konsole
- Heute gibt es komfortable IDEs



The screenshot shows the KDevelop IDE interface. The main window displays a C++ source file named `dbase.cpp`. The code defines a class `DBase` with methods for opening, reading, and closing a database file. The code is as follows:

```
if (file->isOpen()) {
    // falls der File geöffnet war, ist er jetzt
    // geschlossen ;)
    file->close();
}

if (file->open(IO_ReadOnly)) {
    // file erfolgreich geöffnet
    if (!(file->at(index + (int) record_length)) {
        // an das Ende des Records konnte nicht gesprungen
        // werden
        file->close();
        // die Stringlist trotzdem mit Leerstrings belegen
        for (unsigned int i = 0; i < numCols(); ++i) {
            returnlist.append("");
        }
        return (returnlist);
    }
    if (!(file->at(index)) {
        // an den Anfang des Records konnte nicht
        // gesprungen werden
        file->close();
        // die Stringlist trotzdem mit Leerstrings belegen
        for (unsigned int i = 0; i < numCols(); ++i) {
            returnlist.append("");
        }
        return (returnlist);
    }
}
```

The left sidebar shows a class hierarchy for `DBase`, including methods like `columnLen()`, `columnType()`, `getColName()`, `getHeaderL()`, `getLastUpa()`, `getNextRow()`, `getRecordL()`, `getVersion()`, `headerOK()`, `nextRow()`, `numCols()`, `numRows()`, `reset()`, `setNextRow()`, `DBase()`, `file`, `header_len`, `headers`, `index`, `last_update`, and `record_length`. The right sidebar shows a documentation pane with a table of contents and an index.

Programmierung

Entwicklungsumgebungen (IDEs)

Programmierung

Entwicklungsumgebungen (IDEs)

Integrierte Entwicklungsumgebungen verfügen in der Regel über folgende Komponenten:

Programmierung

Entwicklungsumgebungen (IDEs)

Integrierte Entwicklungsumgebungen verfügen in der Regel über folgende Komponenten:

- Texteditor

Programmierung

Entwicklungsumgebungen (IDEs)

Integrierte Entwicklungsumgebungen verfügen in der Regel über folgende Komponenten:

- Texteditor
- Compiler bzw. Interpreter

Programmierung

Entwicklungsumgebungen (IDEs)

Integrierte Entwicklungsumgebungen verfügen in der Regel über folgende Komponenten:

- Texteditor
- Compiler bzw. Interpreter
- Linker

Programmierung

Entwicklungsumgebungen (IDEs)

Integrierte Entwicklungsumgebungen verfügen in der Regel über folgende Komponenten:

- Texteditor
- Compiler bzw. Interpreter
- Linker
- Debugger

Programmierung

Entwicklungsumgebungen (IDEs)

Integrierte Entwicklungsumgebungen verfügen in der Regel über folgende Komponenten:

- Texteditor
- Compiler bzw. Interpreter
- Linker
- Debugger
- Quelltextformatierungsfunktion

Programmierung

Entwicklungsumgebungen (IDEs)

Integrierte Entwicklungsumgebungen verfügen in der Regel über folgende Komponenten:

- Texteditor
- Compiler bzw. Interpreter
- Linker
- Debugger
- Quelltextformatierungsfunktion
- weitere hilfreiche Komponenten wie Versionsverwaltung, Projektmanagement, UML-Modellierung, ...

Programmierung

Entwicklungsumgebungen (IDEs)

Programmierung

Entwicklungsumgebungen (IDEs)

In erster Linie sind integrierte Entwicklungsumgebungen hilfreiche Werkzeuge, die dem Softwareentwickler häufig wiederkehrende Aufgaben abnehmen,

Programmierung

Entwicklungsumgebungen (IDEs)

In erster Linie sind integrierte Entwicklungsumgebungen hilfreiche Werkzeuge, die dem Softwareentwickler häufig wiederkehrende Aufgaben abnehmen, einen schnellen Zugriff auf wichtige Funktionen bieten, mit denen die Arbeits(zwischen)ergebnisse verwaltet und in spätere Bearbeitungsfunktionen direkt überführt werden können.

Programmierung Entwicklungsumgebungen (IDEs)

In erster Linie sind integrierte Entwicklungsumgebungen hilfreiche Werkzeuge, die dem Softwareentwickler häufig wiederkehrende Aufgaben abnehmen, einen schnellen Zugriff auf wichtige Funktionen bieten, mit denen die Arbeits(zwischen)ergebnisse verwaltet und in spätere Bearbeitungsfunktionen direkt überführt werden können.

Der Entwickler wird dadurch von formalen Arbeiten entlastet und kann sich ganz auf seine eigentliche Aufgabe, die Softwareentwicklung/Programmierung konzentrieren.

Programmierung

Klassische Programmiersprachen

Programmierung

Klassische Programmiersprachen

- C, C++, C#

Programmierung

Klassische Programmiersprachen

- C, C++, C#
- Fortran

Programmierung

Klassische Programmiersprachen

- C, C++, C#
- Fortran
- Basic

Programmierung

Klassische Programmiersprachen

- C, C++, C#
- Fortran
- Basic
- Cobol

Programmierung

Klassische Programmiersprachen

- C, C++, C#
- Fortran
- Basic
- Cobol
- Pascal

Programmierung

Klassische Programmiersprachen

- C, C++, C#
- Fortran
- Basic
- Cobol
- Pascal
- Assembler

Programmierung

Moderne Programmiersprachen

Programmierung

Moderne Programmiersprachen

- Java

Programmierung

Moderne Programmiersprachen

- Java
- Mono

Programmierung

Moderne Programmiersprachen

- Java
- Mono
- Perl, Python, Ruby, PHP

Programmierung

Moderne Programmiersprachen

- Java
- Mono
- Perl, Python, Ruby, PHP
- ...

Programmierung

Beispielcode

Programmierung

Beispielcode

Listing 2: Hallo Welt in C

```
#include <stdio.h>
int main() {
    (void) printf(" Hallo , _Welt!\n" );
    return 0;      /* Sauber beenden */
}
```

Programmierung

Beispielcode

Programmierung

Beispielcode

```
File Edit View Selection Find Packages Help
C Python
/* C */
#include <stdio.h>
int quadrat(int i) {
    return i*i;
}
int main(void) {
    int i;
    for (i=1; i<=10; i++) {
        printf("%d %d\n", i, quadrat(i));
    }
    return 0;
}

Java
// Java
public class Main {
    public static int quadrat(int i) {
        return i*i;
    }
    public static void main(String[] args) {
        for(int i=1; i<=10; i++) {
            System.out.println(i + " " + quadrat(i));
        }
    }
}

Haskell
-- Haskell
quadrat x = x*x
main = putStrLn $ concatMap (\i -> show i ++ " " ++ show (quadrat i) ++ "\n") [1..10]
C 1:1 UTF-8 C

# Python
def quadrat(i):
    return i*i
for i in range(1, 11):
    print(i, quadrat(i))

PHP
<?php
// PHP
function quadrat($i)
{
    return $i*$i;
}
for ($i = 1; $i <= 10; $i++)
{
    echo $i, " ", quadrat($i), "\n";
}
?>

JavaScript
// JavaScript
function quadrat(i) {
    return i*i;
}
for (var i=1; i<=10; i++) {
    console.log(i, quadrat(i));
}
```

Programmierung

Interpreter und Kompiler

Programmierung

Interpreter und Compiler

Bei der Erstellung bzw. beim laufenlassen eines Programmes unterscheidet man zwei Möglichkeiten:

Programmierung

Interpreter und Compiler

Bei der Erstellung bzw. beim laufenlassen eines Programmes unterscheidet man zwei Möglichkeiten:

- Interpreter

Programmierung

Interpreter und Compiler

Bei der Erstellung bzw. beim laufenlassen eines Programmes unterscheidet man zwei Möglichkeiten:

- Interpreter

Programmierung

Interpreter und Compiler

Bei der Erstellung bzw. beim laufenlassen eines Programmes unterscheidet man zwei Möglichkeiten:

- Interpreter
übersetzt den Quellcode im Gegensatz zu Assemblern oder Compilern nicht in eine auf dem System direkt ausführbare Datei, sondern liest diesen ein, analysiert ihn und führt ihn aus.

Programmierung

Interpreter und Compiler

Bei der Erstellung bzw. beim laufenlassen eines Programmes unterscheidet man zwei Möglichkeiten:

- Interpreter
 - übersetzt den Quellcode im Gegensatz zu Assemblern oder Compilern nicht in eine auf dem System direkt ausführbare Datei, sondern liest diesen ein, analysiert ihn und führt ihn aus. Die Analyse des Quellcodes erfolgt also zur Laufzeit des Programmes

Programmierung

Interpreter und Compiler

Bei der Erstellung bzw. beim laufenlassen eines Programmes unterscheidet man zwei Möglichkeiten:

- Interpreter
 - übersetzt den Quellcode im Gegensatz zu Assemblern oder Compilern nicht in eine auf dem System direkt ausführbare Datei, sondern liest diesen ein, analysiert ihn und führt ihn aus. Die Analyse des Quellcodes erfolgt also zur Laufzeit des Programmes
- Compiler

Programmierung

Interpreter und Compiler

Bei der Erstellung bzw. beim laufenlassen eines Programmes unterscheidet man zwei Möglichkeiten:

- Interpreter
 - übersetzt den Quellcode im Gegensatz zu Assemblern oder Compilern nicht in eine auf dem System direkt ausführbare Datei, sondern liest diesen ein, analysiert ihn und führt ihn aus. Die Analyse des Quellcodes erfolgt also zur Laufzeit des Programmes
- Compiler

Programmierung

Interpreter und Compiler

Bei der Erstellung bzw. beim laufenlassen eines Programmes unterscheidet man zwei Möglichkeiten:

- **Interpreter**
übersetzt den Quellcode im Gegensatz zu Assemblern oder Compilern nicht in eine auf dem System direkt ausführbare Datei, sondern liest diesen ein, analysiert ihn und führt ihn aus. Die Analyse des Quellcodes erfolgt also zur Laufzeit des Programmes
- **Kompiler**
übersetzt den Quellcode einer bestimmten Programmiersprache in eine Form, die von einem Computer (direkter) ausgeführt werden kann

Programmierung

Freie Entwicklungsumgebungen

Programmierung

Freie Entwicklungsumgebungen

- FreeBASIC

Programmierung

Freie Entwicklungsumgebungen

- FreeBASIC
- Dev-CPP

Programmierung

Freie Entwicklungsumgebungen

- FreeBASIC
- Dev-CPP
- Gambas

Programmierung

Freie Entwicklungsumgebungen

- FreeBASIC
- Dev-CPP
- Gambas
- Lazarus (Free Pascal)

Programmierung

Freie Entwicklungsumgebungen

- FreeBASIC
- Dev-CPP
- Gambas
- Lazarus (Free Pascal)
- Code::Blocks

Programmierung

Freie Entwicklungsumgebungen

- FreeBASIC
- Dev-CPP
- Gambas
- Lazarus (Free Pascal)
- Code::Blocks
- ...

Programmierung

Exoten und Cooles - Potaka

Programmierung

Exoten und Cooles - Potaka

Potaka, erste Programmiersprache in bengalischer Schrift.

Programmierung

Exoten und Cooles - Potaka

Potaka, erste Programmiersprache in bengalischer Schrift.

```
1 /* পতাকা();
2  * মজার মজার কোড লিখুন! রান করুন! সাবমিট করুন!
3  */
4
5 //*** প্রোগ্রাম: ৩ বড় না ৪ বড় ? ***//
6 যদি(৩ থেকে ৪ বড় হয়){
7     দেখাও("বলু: আমি আগেই জানতাম ৪ বড়।");
8 }নাহলে{
9     দেখাও("পলু: ঘোড়ার ডিম");
10 }
11
12
13 //*** প্রোগ্রাম: নন্দলাল ***//
14 ধরি জগত = "সুন্দর";
15 যদি (জগত দেখতে "সুন্দর" হয়){
16     দেখাও("থাকবো নাকো বন্ধ ঘরে দেখবো এবার জগতটাকে");
17 }নাহলে{
18     দেখাও("আমি নন্দলাল হব, এ জগতকে ভয় পাব");
19 }
```

Programmierung

Exoten und Cooles - Chuck

Programmierung

Exoten und Cooles - Chuck

Chuck ist eine Programmiersprache für die Erzeugung und Verarbeitung von Audiosignalen, wobei ein besonderer Wert auf parallele Ausführung und exakte zeitliche Bezüge gelegt ist.

Programmierung

Exoten und Cooles - Chuck

Chuck ist eine Programmiersprache für die Erzeugung und Verarbeitung von Audiosignalen, wobei ein besonderer Wert auf parallele Ausführung und exakte zeitliche Bezüge gelegt ist.

Die Sprache wird interpretiert, dabei ist es möglich, im laufenden Betrieb Programmteile hinzuzufügen, zu verändern oder zu entfernen.

Programmierung

Exoten und Cooles - Chuck

Chuck ist eine Programmiersprache für die Erzeugung und Verarbeitung von Audiosignalen, wobei ein besonderer Wert auf parallele Ausführung und exakte zeitliche Bezüge gelegt ist.

Die Sprache wird interpretiert, dabei ist es möglich, im laufenden Betrieb Programmteile hinzuzufügen, zu verändern oder zu entfernen.

```
1 // demo0.ck
2 // basic demo showing time and duration
3
4 5::second + now => time later;
5
6 while( now < later )
7 {
8     <<<now>>>;
9     1::second => now;
10 }
11
12 <<<now>>>;
```

Programmierung

Exoten und Cooles - Brainfuck

Programmierung

Exoten und Cooles - Brainfuck

Brainfuck ist für den ernsthaften Einsatz umständlich und ineffizient, aber gut geeignet, um Grundlagen der Computertechnik zu erlernen..

Programmierung

Exoten und Cooles - Brainfuck

Brainfuck ist für den ernsthaften Einsatz umständlich und ineffizient, aber gut geeignet, um Grundlagen der Computertechnik zu erlernen..

```
1  ++++++++
2  [
3  >+++++++>+++++++>++++><<<<-
4  ]           Schleife zur Vorbereitung der Textausgabe
5  >++.       Ausgabe von 'H'
6  >+.       Ausgabe von 'e'
7  ++++++.   'l'
8  .         'l'
9  +++.     'o'
10 >++.     Leerzeichen
11 <<<<+++++. 'W'
12 >.      'o'
13 +++.   'r'
14 -----, 'l'
15 -----, 'd'
16 >+.    '!'
17 >.     Zeilenvorschub
18 +++.  Wagenrücklauf
```

Programmierung

Exoten und Cooles - Lolcode

Programmierung

Exoten und Cooles - Lolcode

Lolcode, deren Syntax aus Netzjargon besteht.

Programmierung

Exoten und Cooles - Lolcode

Lolcode, deren Syntax aus Netzjargon besteht.

```
1 HAI
2 CAN HAS STDIO?
3 VISIBLE "HAI WORLD!"
4 KTHXBYE
```

Programmierung

Exoten und Cooles - Honorable Mentions

Programmierung

Exoten und Cooles - Honorable Mentions

- Cow

Programmierung

Exoten und Cooles - Honorable Mentions

- Cow
- Whitespace

Programmierung

Wo sitzen die besten Hacker?

Programmierung

Wo sitzen die besten Hacker?

Das Programmier-Portal HackerRank hat die Programmierfähigkeiten von Codern rund um den Globus getestet.

Programmierung

Wo sitzen die besten Hacker?

Das Programmier-Portal HackerRank hat die Programmierfähigkeiten von Codern rund um den Globus getestet.

Auf HackerRank sind nach eigenen Angaben 1,5 Millionen Programmierer aus aller Welt registriert.

Programmierung

Wo sitzen die besten Hacker?

Programmierung

Wo sitzen die besten Hacker?

Programmierung

Wo sitzen die besten Hacker?

Which Country Has the Best Developers?

Ranked by Average Score Across All HackerRank Challenges

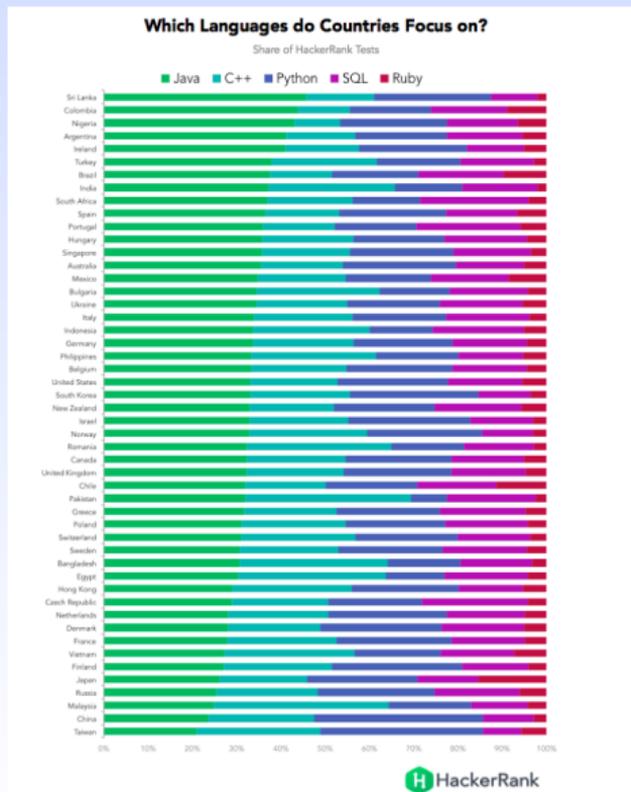
| Rank | Country | Score Index | Rank | Country | Score Index |
|------|----------------|-------------|------|----------------|-------------|
| 1 | China | 100.0 | 26 | Netherlands | 78.9 |
| 2 | Russia | 99.9 | 27 | Chile | 78.4 |
| 3 | Poland | 98.0 | 28 | United States | 78.0 |
| 4 | Switzerland | 97.9 | 29 | United Kingdom | 77.7 |
| 5 | Hungary | 93.9 | 30 | Turkey | 77.5 |
| 6 | Japan | 92.1 | 31 | India | 76.0 |
| 7 | Taiwan | 91.2 | 32 | Ireland | 75.9 |
| 8 | France | 91.2 | 33 | Mexico | 75.7 |
| 9 | Czech Republic | 90.7 | 34 | Denmark | 75.6 |
| 10 | Italy | 90.2 | 35 | Israel | 74.8 |
| 11 | Ukraine | 88.7 | 36 | Norway | 74.6 |
| 12 | Bulgaria | 87.2 | 37 | Portugal | 74.2 |
| 13 | Singapore | 87.1 | 38 | Brazil | 73.4 |
| 14 | Germany | 84.3 | 39 | Argentina | 72.1 |
| 15 | Finland | 84.3 | 40 | Indonesia | 71.8 |
| 16 | Belgium | 84.1 | 41 | New Zealand | 71.6 |
| 17 | Hong Kong | 83.6 | 42 | Egypt | 69.3 |
| 18 | Spain | 83.4 | 43 | South Africa | 68.3 |
| 19 | Australia | 83.2 | 44 | Bangladesh | 67.8 |
| 20 | Romania | 81.9 | 45 | Colombia | 66.0 |
| 21 | Canada | 81.7 | 46 | Philippines | 63.8 |
| 22 | South Korea | 81.7 | 47 | Malaysia | 61.8 |
| 23 | Vietnam | 81.1 | 48 | Nigeria | 61.3 |
| 24 | Greece | 80.8 | 49 | Sri Lanka | 60.4 |
| 25 | Sweden | 79.9 | 50 | Pakistan | 57.4 |

Programmierung

Wo sitzen die besten Hacker?

Programmierung

Wo sitzen die besten Hacker?



Programmierung

Verständnisfrage

Why do Java developers wear glasses?

Programmierung

Verständnisfrage

Programmierung

Verständnisfrage

Because they can't C#

Programmierung

Muahahahaaa

Programmierung Muahahahaaa



Programmierung

Einstieg mit Gambas

Programmierung

Einstieg mit Gambas

Gambas ist eine voll ausgestattete Objektsprache und Entwicklungsumgebung, die auf einem BASIC-Interpreter basiert.

Die Architektur ist stark durch Java inspiriert, und deshalb besteht Gambas aus:

Programmierung

Einstieg mit Gambas

Gambas ist eine voll ausgestattete Objektsprache und Entwicklungsumgebung, die auf einem BASIC-Interpreter basiert.

Die Architektur ist stark durch Java inspiriert, und deshalb besteht Gambas aus:

- einem Compiler

Programmierung

Einstieg mit Gambas

Gambas ist eine voll ausgestattete Objektsprache und Entwicklungsumgebung, die auf einem BASIC-Interpreter basiert.

Die Architektur ist stark durch Java inspiriert, und deshalb besteht Gambas aus:

- einem Compiler
- einem Interpreter

Programmierung

Einstieg mit Gambas

Gambas ist eine voll ausgestattete Objektsprache und Entwicklungsumgebung, die auf einem BASIC-Interpreter basiert.

Die Architektur ist stark durch Java inspiriert, und deshalb besteht Gambas aus:

- einem Compiler
- einem Interpreter
- einem Archivar

Programmierung

Einstieg mit Gambas

Gambas ist eine voll ausgestattete Objektsprache und Entwicklungsumgebung, die auf einem BASIC-Interpreter basiert.

Die Architektur ist stark durch Java inspiriert, und deshalb besteht Gambas aus:

- einem Compiler
- einem Interpreter
- einem Archivar
- einem Scripter

Programmierung

Einstieg mit Gambas

Gambas ist eine voll ausgestattete Objektsprache und Entwicklungsumgebung, die auf einem BASIC-Interpreter basiert.

Die Architektur ist stark durch Java inspiriert, und deshalb besteht Gambas aus:

- einem Compiler
- einem Interpreter
- einem Archivar
- einem Scripter
- einer Entwicklungsumgebung (IDE), welche selbst mit Gambas geschrieben wurde

Programmierung

Einstieg mit Gambas

Gambas ist eine voll ausgestattete Objektsprache und Entwicklungsumgebung, die auf einem BASIC-Interpreter basiert.

Die Architektur ist stark durch Java inspiriert, und deshalb besteht Gambas aus:

- einem Compiler
- einem Interpreter
- einem Archivar
- einem Scripter
- einer Entwicklungsumgebung (IDE), welche selbst mit Gambas geschrieben wurde
- und vielen Erweiterungskomponenten

Programmierung

Einstieg mit Gambas

- Gambas ist eine freie Basic-Implementierung, jedoch kein Klon von MS Visual Basic

Programmierung

Einstieg mit Gambas

- Gambas ist eine freie Basic-Implementierung, jedoch kein Klon von MS Visual Basic
- Gambas bezieht auch Ideen von Java ein und existiert schon sehr lange

Programmierung

Einstieg mit Gambas

- Gambas ist eine freie Basic-Implementierung, jedoch kein Klon von MS Visual Basic
- Gambas bezieht auch Ideen von Java ein und existiert schon sehr lange
- Es können grafische Programme entwickelt werden (QT und GTK)

Programmierung

Einstieg mit Gambas

- Gambas ist eine freie Basic-Implementierung, jedoch kein Klon von MS Visual Basic
- Gambas bezieht auch Ideen von Java ein und existiert schon sehr lange
- Es können grafische Programme entwickelt werden (QT und GTK)
- Die eingebaute SDL-Bindung lässt das Schreiben von Multimedia-Applikationen und Spielen zu

Programmierung

Einstieg mit Gambas

- Gambas ist eine freie Basic-Implementierung, jedoch kein Klon von MS Visual Basic
- Gambas bezieht auch Ideen von Java ein und existiert schon sehr lange
- Es können grafische Programme entwickelt werden (QT und GTK)
- Die eingebaute SDL-Bindung lässt das Schreiben von Multimedia-Applikationen und Spielen zu
- Zugriff auf Datenbanken wird mit einer einheitlichen Schnittstelle unterstützt

Programmierung

Einstieg mit Gambas

- Gambas ist eine freie Basic-Implementierung, jedoch kein Klon von MS Visual Basic
- Gambas bezieht auch Ideen von Java ein und existiert schon sehr lange
- Es können grafische Programme entwickelt werden (QT und GTK)
- Die eingebaute SDL-Bindung lässt das Schreiben von Multimedia-Applikationen und Spielen zu
- Zugriff auf Datenbanken wird mit einer einheitlichen Schnittstelle unterstützt
- Beispielsweise wurden der Webbrowser Areida (Webkit) und einige HAM-Radio-Anwendungen in Gambas realisiert

Programmierung

Einstieg mit Gambas

- Die Sprache ist klein und schnell, und kann auch als Skriptsprache eingesetzt werden

Programmierung

Einstieg mit Gambas

- Die Sprache ist klein und schnell, und kann auch als Skriptsprache eingesetzt werden
- Sie enthält

Programmierung

Einstieg mit Gambas

- Die Sprache ist klein und schnell, und kann auch als Skriptsprache eingesetzt werden
- Sie enthält
 - eine vollständige Fehlerbearbeitung

Programmierung

Einstieg mit Gambas

- Die Sprache ist klein und schnell, und kann auch als Skriptsprache eingesetzt werden
- Sie enthält
 - eine vollständige Fehlerbearbeitung
 - Datei- und Prozess-Operationen

Programmierung

Einstieg mit Gambas

- Die Sprache ist klein und schnell, und kann auch als Skriptsprache eingesetzt werden
- Sie enthält
 - eine vollständige Fehlerbearbeitung
 - Datei- und Prozess-Operationen
 - Ereignissteuerung

Programmierung

Einstieg mit Gambas

- Die Sprache ist klein und schnell, und kann auch als Skriptsprache eingesetzt werden
- Sie enthält
 - eine vollständige Fehlerbearbeitung
 - Datei- und Prozess-Operationen
 - Ereignissteuerung
 - Unterstützung für UTF-8

Programmierung

Einstieg mit Gambas

- Die Sprache ist klein und schnell, und kann auch als Skriptsprache eingesetzt werden
- Sie enthält
 - eine vollständige Fehlerbearbeitung
 - Datei- und Prozess-Operationen
 - Ereignissteuerung
 - Unterstützung für UTF-8
 - Internationalisierung und Übersetzungen

Programmierung

Einstieg mit Gambas

- Die Sprache ist klein und schnell, und kann auch als Skriptsprache eingesetzt werden
- Sie enthält
 - eine vollständige Fehlerbearbeitung
 - Datei- und Prozess-Operationen
 - Ereignissteuerung
 - Unterstützung für UTF-8
 - Internationalisierung und Übersetzungen
 - die Möglichkeit, Funktionen in dynamischen Bibliotheken aufzurufen

Programmierung

Einstieg mit Gambas

- Die Sprache ist klein und schnell, und kann auch als Skriptsprache eingesetzt werden
- Sie enthält
 - eine vollständige Fehlerbearbeitung
 - Datei- und Prozess-Operationen
 - Ereignissteuerung
 - Unterstützung für UTF-8
 - Internationalisierung und Übersetzungen
 - die Möglichkeit, Funktionen in dynamischen Bibliotheken aufzurufen
- Gambas ist vollständig objektorientiert mit Klassen, öffentlichen und privaten Symbolen, Polymorphismus, Vererbung und einigem mehr...

Programmierung

Einstieg mit Gambas

Programmierung

Einstieg mit Gambas

Aktuelle Version ist 3.9.0 vom 28.08.16.

Sie enthält unter Anderem folgende neuen Features:

Programmierung

Einstieg mit Gambas

Aktuelle Version ist 3.9.0 vom 28.08.16.

Sie enthält unter Anderem folgende neuen Features:

- gb.web.form (erlaubt die Erstellung von GUIs für Webanwendungen)

Programmierung

Einstieg mit Gambas

Aktuelle Version ist 3.9.0 vom 28.08.16.

Sie enthält unter Anderem folgende neuen Features:

- gb.web.form (erlaubt die Erstellung von GUIs für Webanwendungen)
- gb.form.terminal (stellt ein vollständige Terminal-Emulation zur Verfügung)

Programmierung

Einstieg mit Gambas

Aktuelle Version ist 3.9.0 vom 28.08.16.

Sie enthält unter Anderem folgende neuen Features:

- gb.web.form (erlaubt die Erstellung von GUIs für Webanwendungen)
- gb.form.terminal (stellt ein vollständige Terminal-Emulation zur Verfügung)
- gb.net.smtp (Komponente für PLAIN und CRAM-MD5 authentifizierung)

Programmierung Hands-On

Programmierung Hands-On



Links zur Präsentation

<http://gambas.sourceforge.net/>

<http://www.gambas-buch.de/dw/doku.php>

<http://gambas-club.de>

<http://beginnersguidetogambas.com>

<http://howtogambas.org>

<http://gambos.org>

Links zur Präsentation

<http://chuck.cs.princeton.edu/>

<http://www.freepascal.org/>

<http://developer.android.com>

<http://www.codeblocks.org>

https://de.wikipedia.org/wiki/Liste_von_Integrierten_

[Entwicklungsumgebungen#Freie_integrierte_Entwicklungsumgebungen](https://de.wikipedia.org/wiki/Liste_von_Integrierten_Entwicklungsumgebungen#Freie_integrierte_Entwicklungsumgebungen)

Weitere Informationen bekommen Sie hier:

`http://www.FreieSoftwareOG.org`
und
`Kontakt@FreieSoftwareOG.org`

oder kommen Sie doch einfach zu unserem regelmäßigen Treffen,
jeden 1. Mittwoch im Monat ab 20:00 Uhr.
(Treffpunkt und Thema laut Webseite)

