

# Crashkurs Software-Know-how

Software-Welten kennenlernen,  
Programmierung selbst erleben

# Crashkurs Software-Know-how

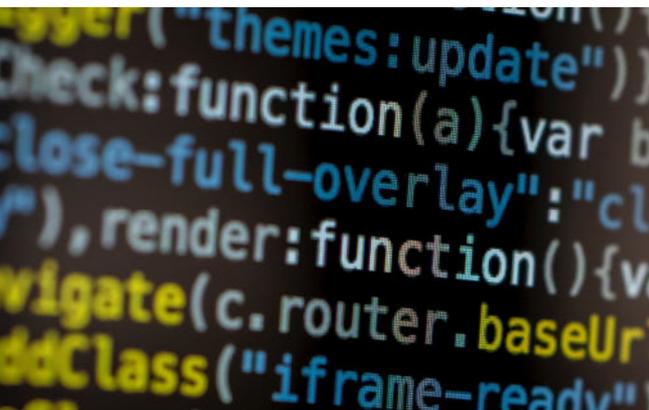
Software-Welten kennenlernen, Programmierung selbst erleben

## Kurzbeschreibung

Heute ist die ganze Welt computergesteuert. Ohne Software jedoch sind Computer lediglich tote, nutzlose Materie. Die Geschichte der Programmierung ist spannend, Software-Erstellung selbst jedoch komplex, anspruchsvoll und oftmals mehr als nur verwirrend.

Dieses Seminar erläutert Ihnen zunächst auf unterhaltsame und zugleich hochinformativ Weise, wie Software funktioniert, welchen Anforderungen sie standhalten muss und wie sich ihre einzelnen Entwicklungsstufen darstellen. Zudem erleben Sie "hautnah", welches Wissen und welche Kompetenzen erforderlich sind, um fremde Software im Projektfall kompetent einschätzen zu können. Dies stärkt ihre Position in alle Richtungen deutlich.

**Das Besondere:** Unter Einsatz sog. Online-Compiler programmieren Sie mit dem Dozenten live, lernen also, wie man unter Einsatz diverser Programmiersprachen und Tools effiziente Algorithmen entwickelt und Code schreibt. Hierzu sind keinerlei Installationen oder Vorbereitungen auf Ihrem Rechner erforderlich.



## Inhalte (Auszug)

- Aufbau, Arbeitsweise und Erstellungsprozess von Software
- SW-Qualitätsanforderungen im Detail, „1000 Hürden“
- SW-Projekte planen, Tools wählen, Code schreiben
- „Werkzeugkasten“: Frameworks, Toolkits, APIs, SDKs
- Hochsprachen: Evolution, Stammbaum, Syntax, Typisierung
- Algorithmus-/Programm-Effizienz, Komplexität, Robustheit
- Plattform-übergreifend Mobile- und Web-Apps entwickeln
- Compiler, Debugger, Testing- und Upgrading-Strategien

## Was lernen Sie in diesem Seminar?

Sie finden sich schnell und sicher in die Welt moderner Software ein und verschaffen sich so ein Fundament für eventuelle weitere Spezialisierungen. Sie erfahren auf den Punkt die jeweilige zündende Idee hinter modernen Schlagworten aus dem Reich der Programmierung. Ob in Bezug auf Windows-, Android- oder Apple-Endgeräte – Software-Entwicklung wird entmystifiziert, Programm-Strukturelemente werden klar und zugehörige Programmier-Tools nachvollziehbar. Nach Besuch unseres Seminars sind Sie in der Lage, Software-Anforderungen zu formulieren, Programm-Pakete einzuschätzen und sogar selbst erste Programme zu erstellen.

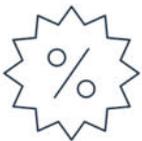
# Crashkurs Software-Know-how

## Software-Welten kennenlernen, Programmierung selbst erleben

### Zielgruppe

Das zweitägige Seminar richtet sich an alle Mitarbeitenden und Führungskräfte mit leichten IT-Grundkenntnissen. Bei Bedarf können Sie sich diese über unseren [Crashkurs IT-Hintergrundwissen](#) desselben Dozenten aneignen. Dies ist jedoch ausdrücklich **keine** Voraussetzung zur Teilnahme. Ideal ist das Seminar zum Beispiel für Quereinsteiger, die sich aufgrund neuer Herausforderungen im Unternehmen mit Software-Auswahl oder gar -Programmierung befassen wollen oder müssen.

**Wichtig:** Es wird nicht davon ausgegangen, dass die Teilnehmenden zukünftig selbst programmieren sollen. Vielmehr vermittelt Ihnen der Dozent lebendig sein fundiertes Wissen und seine umfassende Erfahrung, damit Sie die moderne Software und Software-Entwicklung besser verstehen und ein Gefühl für diese Materie entwickeln.



### Zusatzinformationen

- Das Seminar hat eine begrenzte **Teilnehmerzahl von 15 Personen**. Unser Dozent kann dadurch gezielt auf individuelle Fragestellungen eingehen. Die Mindestteilnehmerzahl beträgt 4.
- Die Bitkom Akademie ist [anerkannter Bildungsträger in Baden-Württemberg](#) und [Nordrhein-Westfalen](#). Teilnehmende haben im Rahmen des Bildungszeitgesetzes die Möglichkeit, Bildungsurlaub bzw. eine Bildungsfreistellung zu beantragen. Auf Anfrage erstellen wir Anträge auf Anerkennung unserer Seminar-Veranstaltungen auch in anderen Bundesländern.
- Dieses Online-Seminar wird mit Zoom durchgeführt. Systemvoraussetzungen und unterstützte Betriebssysteme können Sie [hier](#) einsehen. Für die Einwahl in Zoom über die verschiedenen Anwendungen (Desktop Client, App oder Web-Client) bietet Ihnen [diese Tabelle](#) einen zusätzlichen Vergleich zu den jeweiligen Eigenschaften.
- Wir erklären ausdrücklich, dass beim Bitkom – Unterzeichner der Charta der Vielfalt – jede Person, unabhängig von Geschlecht, Nationalität, ethnischer Herkunft, Religion oder Weltanschauung, Behinderung, Alter, sexueller Orientierung und Identität willkommen ist.

# Seminarprogramm

## Crashkurs Software-Know-how

### Tag 1

#### **Einführung in die Software-Entwicklung**

- Evolution, Klassifizierung, Philosophien, Verwandtschaften und Anwendungsgebiete von Programmiersprachen
- Grundlegende SW-Funktionsweise, Aufbau, Strukturelemente, Abstraktionsprinzip, Syntax und Semantik
- Die Schritte und Architektur-Entscheidungen vom Algorithmen-Set zum lauffähigen Gesamtprogramm
- Wie schreibt man robuste, performante, reaktive, flexible, fehlertolerante und anpassungsfähige Programme?
- Wie SW die HW steuert, wie IT-Komplexität beherrschbar wird, das wichtige 3-Schichten-Modell, Illusion Grafik

#### **Prozesse und Werkzeuge der Programmierung**

- Wahl von Entwicklungsumgebung (IDE), Frameworks, APIs, SDKs, OOP-Klassenbibliotheken, Drittkomponenten
- Sukzessive Quellcode-Eingabe, alternierend mit mehrstufigen Testvorgängen und Fehlersuche /-behebung
- Systematisches Debugging im Detail: Einblick in Tausende von Variablen und Objekte zur Laufzeit erhalten
- Compiler, Linker, Monitor, Profiler: die wichtigsten Tools bei der Erstellung von Desktop bzw. Mobile Apps
- Lösungskonzepte zur Überwindung spezifischer Endgeräte-Hardware und Betriebssystem-Abhängigkeiten

#### **Philosophien und Methoden der Software-Erstellung**

- U.a. prozedurale, funktionale, objektorientierte, rekursive Programmierung: Merkmale und Einsatztechniken
- Programmier-Methoden und -Paradigmen im Vergleich: u.a. Scrum, agile SW-Entwicklung, klassische Modelle
- SW-Leistungsparameter schon bei der Algorithmus-Wahl berücksichtigen: KPIs der Performanz, Latenz,  $O(n)$  etc.
- Asynchron und parallel laufende Programmstränge realisieren, resiliente SW: was ist alles zu berücksichtigen?
- Bsp. Such- und Sortier-Algorithmen: warum sich die Skalierungsperformanz oft in Zehnerpotenzen unterscheidet

#### **Didaktischer Aufbau**

**Tag 1** vermittelt Ihnen spannend, unterhaltsam und allgemeinverständlich Funktionsweise, Unterschiede und Anwendung aktueller Software-Technologien sowie alle relevanten Zusammenhänge. Spontane Fragen sind ausdrücklich erwünscht. Eingeflochtene Quizfragen lockern das Seminar auf. Per Online-Compiler werden von Beginn an Algorithmen zu kleineren Aufgabenstellungen in handfeste Programme gegossen. Diese werden sofort gestartet, getestet, analysiert und sukzessive gemeinsam optimiert.

# Seminarprogramm

## Crashkurs Software-Know-how

### Tag 2

#### Moderne Software-Problemstellungen

- Programme Plattform-übergreifend konzipieren u.a. für Windows, Android, iOS und diverse Device-Typen
- mobile Endgeräte: flexible Bedienoberflächen, Hardware-Abhängigkeit, Leistungsparameter
- Ergonomie-Ansprüche, Internet-Konnektivität, Cloud-basierte Datenbank-Anbindung, Bedienelemente
- Fatale Fehler und Missverständnisse bei der SW-Planung/-Erstellung, u.a. Unterschätzung von Komplexitäten
- Lösung des Dokumentationsdilemmas, SW-Wartung /-Internationalisierung, "Parallelwelt" Web-App-Erstellung

#### Weitere Software-Themen und -Begriffe (AUSZUG)

- .NET/C#: ein starkes Team, neuartige Hochsprachen ('D', F#, Carbon), Unterschiede populärer OS-Frameworks
- IT-Systemarchitekturen: auf welche bewährten Standards sich Ihr Programm stützen kann (OSI, JSON, XML usw.)
- Datentypen-Vergleich zwischen diversen HLLs, OOP "unter der Haube", Algorithmen zur String-Verarbeitung
- Lokale und Cloud-basierte DB-Anbindung, SQL- bzw. LINQ-Integration, UML-Klassendefinition /-Notation
- Analogien: u.a. Programmier- ↔ natürliche Sprachen, OOP ↔ Biologie, Software- ↔ Unternehmensprozesse

#### Didaktischer Aufbau

**Tag 2** erläutert weiterhin anschaulich zentrale Prinzipien und Fachbegriffe im Software- und Programmierbezug, dabei wird flexibel auf alle individuellen Fragen eingegangen. Großen Wert legt der Dozent auf wichtige Hinweise zu typischen Fehlern, Mythen und Missverständnissen im Software-Bereich. Im Verlauf des Seminars erstellen Sie gemeinsam Programme steigenden Anspruchs auf Basis unterschiedlichster Programmiersprachen, deren Unterschiede, Philosophien und Vor- bzw. Nachteile Ihnen so plastisch und praxisnah deutlich werden.

# Ihr Referent



## Diplom-Inform. Univ. Stefan Hable

Geschäftsführer HMC Management Consulting

- seit **2002**: Geschäftsführer und Inhaber der international tätigen Berliner Unternehmensberatung HMC Management Consulting
- ab **2000** Niederlassungs- und mehrfacher Business-Unit-Leiter in der Telekommunikation
- ab **1998** Bereichsleiter und stv. Geschäftsleitung in der ERP-Software-Entwicklung (Unternehmenslogistik)
- ab **1993** Entwicklungsleiter und Projektmanager in einem japanischen Medizintechnik-Konzern
- seit **1993**: Diplom-Informatiker, Friedrich-Alexander-Universität Erlangen-Nürnberg
- seit jeher starke IT-Affinität, **40 Jahre** aktive Erfahrung in der IT-System- und Software-Entwicklung
- AEVO-Ausbilderqualifikation, ausgeprägte internationale Erfahrung, interdisziplinäre Ausrichtung

Als kompetenter und erfahrener Informatiker, der selbst anspruchsvollste, komplexe Software konzipiert und erstellt hat (z.B. zur medizinischen Bilddiagnostik), als IT-Manager die Realisierung hochvolumiger Software-Pakete verantwortete (z.B. im ERP- und Mobilfunk-Bereich) und heute noch aktiv Apps programmiert, erläutert der Referent alle zentralen Aspekte zum Thema Software und Programmierung.

## Shortfacts



### Termine, Preise und Veranstaltungsorte

Alle aktuellen Informationen finden Sie auf der Website der [Bitkom Akademie](https://www.bitkom-akademie.de).

**Kontaktieren Sie uns – wir beraten Sie gern.**

Bitkom Akademie | Albrechtstraße 10 | 10117 Berlin  
T 030 27576-540 | [info@bitkom-akademie.de](mailto:info@bitkom-akademie.de)  
Weitere Seminare finden Sie unter [www.bitkom-akademie.de](https://www.bitkom-akademie.de)

**bitkom**  
akademie