Inhalt

Grundlagen zu IT-Themen 3
1.1 Netzwerke 3
1.1.1 Was ist ein Netzwerk? 3
1.1.2 IP-Adresse 3
1.1.3 Subnetzmaske 3
1.1.4 Router / Gateway 3
1.1.5 DNS 3
1.1.6 Typische Begriffe 3
1.1.7 Wichtige Ports 3
1.2 Objektorientierte Programmierung (OOP) 4
1.2.1 Was ist OOP? 4
1.2.2 Objekte – Dinge aus der realen Welt 4
1.2.3 Klasse – die Beschreibung eines Objekts im Quellcode 4
1.2.4 Instanz – das Objekt im Arbeitsspeicher 5
1.2.5 Warum ist diese Unterscheidung wichtig? 5
1.2.6 Attribute sind **Eigenschaften** 5
1.2.7 Methoden sind Aktionen 6
1.2.8 Die vier Grundideen der OOP (in einfachem Bild) 6
1.2.9 Warum ist OOP praktisch? 6
1.3 Linux – Wichtige Befehle für Einsteiger 7
1.3.1 Was ist Linux? 7
1.3.2 Navigation im System 7
1.3.3 Dateien und Ordner bearbeiten 7
1.3.4 Systeminformationen 7
1.3.5 Rechte & Benutzer 7
1.3.6 Netzwerkbefehle 7
1.4 SQL Basics 8
1.4.1 Was ist SQL? 8
1.4.2 Wichtige Begriffe 8
1.4.3 Die vier Grundbefehle 8
1.4.4 Beispiel 8
1.4.5 Joins – Tabellen verbinden 8
1.5 Cloud Basics 9
Was ist die Cloud? 9

1.5.1 Die drei Arten von Cloud-Diensten ... 9
1.5.2 Warum ist die Cloud praktisch? ... 9
1.5.3 Grundbausteine in der Cloud ... 9
Typische Anbieter ... 9
1.6 IT-Security Essentials ... 10
1.6.1 Was bedeutet IT-Sicherheit? ... 10
1.6.2 Die drei wichtigsten Ziele ... 10
1.6.3 Typische Gefahren ... 10
1.6.4 Wie können Sie sich schützen? ... 10
1.6.5 Wichtige Konzepte ... 10

Grundlagen zu IT-Themen

1.1 Netzwerke

1.1.1 Was ist ein Netzwerk?

Ein Netzwerk ist wie ein Straßensystem für Computer. Es verbindet Geräte miteinander, damit sie Informationen austauschen können. So wie Sie einem Freund eine Nachricht schicken – nur viel schneller und automatisch.

1.1.2 IP-Adresse

Stellen Sie sich vor, jeder Computer hat eine eigene Hausnummer.
Die IP-Adresse sagt: „Hier wohne ich."
Beispiel: 192.168.0.10

1.1.3 Subnetzmaske

Das ist wie eine Straßenkarte. Sie sagt, welche Häuser zur gleichen Straße gehören und wer „Nachbar" ist. IP-Adresse und Subnetzmaske gehören untrennbar zusammen.

1.1.4 Router / Gateway

Der Router ist wie ein Postbote:

- Er weiß, wohin Daten müssen.
- Wenn etwas außerhalb des eigenen Netzwerks soll, sagt der Router: „Kein Problem, ich bringe das hin!"

1.1.5 DNS

DNS ist das Telefonbuch des Internets. Wenn Sie „google.de" eingeben, fragt Ihr Computer das DNS: „Kannst du mir bitte die Nummer (IP-Adresse) dazu sagen?"

1.1.6 Typische Begriffe

LAN – Netzwerk in einem Gebäude

WAN – große Netzwerke, z. B. das Internet

DHCP – ein System, das automatisch IP-Adressen verteilt („Hier, nimm diese Adresse!" sagt es zum Computer)

MAC-Adresse – die Seriennummer der Netzwerkkarte

1.1.7 Wichtige Ports

Ports sind wie Türen an einem Haus.

- 80 = HTTP (Webseiten)
- 443 = HTTPS (sichere Webseiten)
- 22 = SSH (Fernzugriff)

1.2 Objektorientierte Programmierung (OOP)

1.2.1 Was ist OOP?

Objektorientierte Programmierung bedeutet, dass Programme so gebaut werden, wie wir über Dinge in der echten Welt nachdenken. Das macht Software verständlicher, ordentlicher und einfacher zu erweitern.

1.2.2 Objekte – Dinge aus der realen Welt

Stellen Sie sich ein Auto, einen Hund oder eine Lampe vor. Das sind Dinge mit Eigenschaften und Fähigkeiten.

Ein Objekt ist so ein Ding – nur im Kopf oder als Idee des Anwendungsentwicklers.

Es hat:

- **Eigenschaften** (z. B. Farbe, Gewicht, Status „an“ oder „aus“)
- **Fähigkeiten** (z. B. „fahre los“, „bell“, „mach Licht an“)

Ein Objekt ist also die Vorstellung eines realen Dings.

1.2.3 Klasse – die Beschreibung eines Objekts im Quellcode

Bevor der Computer etwas benutzen kann, braucht er einen **Bauplan**.

Eine Klasse ist genau dieser Bauplan.
Sie sagt dem Computer:

- Welche Eigenschaften so ein Ding haben soll
- Welche Fähigkeiten es hat
- Wie es sich grundsätzlich verhält

Man kann sagen:
Die Klasse ist nur der Bauplan – aber sie ist selbst noch kein Objekt.

Beispiel:

- Klasse: „Auto“ → beschreibt allgemein, wie Autos aufgebaut sind
- Klasse enthält Eigenschaften wie Farbe, PS, Hersteller
- Klasse enthält Methoden wie „fahreLos()“ und „bremse()“

Klasse = Code, kein echtes Auto

1.2.4 Instanz – das Objekt im Arbeitsspeicher

Erst wenn der Computer den Bauplan benutzt, entsteht eine **Instanz**.

Eine Instanz ist:

- das **konkret erzeugte Objekt**,
- mit **echten Daten**,
- das **im Arbeitsspeicher existiert**.

Vergleich:

- Klasse = Bauplan „Auto“
- Instanz = das **einzelne Auto**, z. B. Ihres, mit
    - roter Farbe
    - 120 PS
    - aktuellem Kilometerstand

Instanz = das fertige Objekt, das wirklich existiert – zumindest im Speicher des Computers.

1.2.5 Warum ist diese Unterscheidung wichtig?

Weil sie hilft, OOP leichter zu verstehen:

Begriff	Erklärung
Objekt	Ein Ding, das Sie aus der echten Welt kennen
Klasse	Der Bauplan des Dings im Quellcode
Instanz	Das „echte“ Ding im Arbeitsspeicher, mit seinen Werten

Wenn der Computer etwas damit tun soll, sagen Sie zu ihm:
„Du, ruf bitte die Methode fahreLos() auf die Instanz meines Autos auf!“

1.2.6 Attribute sind Eigenschaften

Beispiele:

- **Auto**: Farbe, Anzahl Türen, PS
- **Hund**: Name, Alter, Größe

1.2.7 Methoden sind Aktionen

Das sind **Fähigkeiten oder Aktionen**, die ein Objekt ausführen kann.
Wenn der Computer das macht, sprechen Sie ihn an:

- „Du, *fahreLos()* bitte!“
- „Du, *machLichtAn()*!“

Methoden = Verhalten des Objekts.

1.2.8 Die vier Grundideen der OOP (in einfachem Bild)

Kapselung: Dinge, die zusammengehören, bleiben zusammen (wie Werkzeug in einer Kiste).

Abstraktion: nur das Wichtige wird gezeigt. Ein Auto hat einen Motor – aber Sie müssen nicht wissen, wie jede Schraube funktioniert, um zu fahren.

Vererbung: Ein Bauplan kann von einem anderen erben. Beispiel:

- Klasse Auto
- Klasse Elektroauto erbt Eigenschaften von Auto, hat aber zusätzliche Fähigkeiten

Polymorphie: gleiche Aktion, unterschiedliche Umsetzung. Eine Methode kann je nach Objekt unterschiedlich funktionieren. Alle Autos können „fahreLos()“ – aber wie schnell oder wie genau, ist unterschiedlich.

1.2.9 Warum ist OOP praktisch?

Code wird verständlicher

Sie können Teile wiederverwenden

Große Programme werden überschaubarer

1.3 Linux – Wichtige Befehle für Einsteiger

1.3.1 Was ist Linux?

Linux ist ein Betriebssystem – so wie Windows oder macOS.
Es hilft Ihrem Computer zu entscheiden, **was passieren soll**, wenn Sie etwas tun möchten.

Aber: Bei Linux geben Sie oft Anweisungen in Form von Text ein.
Dann sprechen **Sie** mit Ihrem Computer – und der Computer antwortet.

1.3.2 Navigation im System

Sie können Ihrem Computer sagen:

- cd – „Geh bitte in dieses Verzeichnis!“
- ls – „Zeig mir, was hier drin ist!“
- pwd – „Sag mir, wo wir gerade sind!“

1.3.3 Dateien und Ordner bearbeiten

- mkdir ordner – „Erstelle diesen Ordner!“
- rm datei – „Lösch diese Datei!“
- cp quelle ziel – „Kopiere das hierhin!“
- mv quelle ziel – „Verschiebe oder benenne es um!“

1.3.4 Systeminformationen

- top – zeigt an, was Ihr Computer gerade macht
- df -h – zeigt freien Speicherplatz
- uname -a – zeigt Infos über das System

1.3.5 Rechte & Benutzer

- Linux ist sehr streng mit Berechtigungen.
- chmod – „Gib dieser Datei bestimmte Rechte!“
- sudo – „Mach das bitte mit Administratorrechten!“

1.3.6 Netzwerkbefehle

- ip a – zeigt Netzwerkadressen
- ping adresse – „Schick Test-Pakete hin und prüfe, ob jemand antwortet!“
- apt update – „Aktualisiere bitte die Paketliste!“

1.4 SQL Basics

1.4.1 Was ist SQL?

SQL ist eine Sprache, mit der Sie einem Computer sagen können: „Bitte hol mir diese Daten“, „Bitte speichere das“ oder „Bitte ändere etwas“.
Stellen Sie sich eine große Kiste voller Karteikarten vor – SQL hilft Ihnen, genau die Karte zu finden, die Sie suchen, oder neue Karten einzusortieren.

1.4.2 Wichtige Begriffe

Tabelle = ein Blatt Papier mit Reihen und Spalten

Zeile = ein Eintrag

Spalte = eine Kategorie, z. B. „Name“

Primary Key = eine eindeutige Nummer, um einen Eintrag wiederzufinden

1.4.3 Die vier Grundbefehle

SELECT – „Zeig mir diese Daten!“

INSERT – „Füge diesen Eintrag hinzu!“

UPDATE – „Ändere diesen Wert!“

DELETE – „Lösch diesen Eintrag!“

1.4.4 Beispiel

SELECT * FROM users;
= „Zeig mir alle Nutzerinnen und Nutzer.“

1.4.5 Joins – Tabellen verbinden

Wenn zwei Ordner Informationen enthalten (z. B. Personen & Bestellungen) und Sie diese gemeinsam sehen möchten, werden die Ordner „verbunden“.

1.5 Cloud Basics

Was ist die Cloud?

Die Cloud ist wie ein großer Computer an einem anderen Ort: Sie besitzen ihn nicht, aber Sie dürfen ihn nutzen.
Es ist so, als würden Sie eine Werkstatt verwenden, statt eigenes Werkzeug kaufen zu müssen.

1.5.1 Die drei Arten von Cloud-Diensten

IaaS – Sie erhalten Werkzeuge (z. B. virtuelle Server) und bauen alles selbst.

PaaS – Sie erhalten eine vorbereitete Werkbank (Entwicklungsumgebung).

SaaS – Sie nutzen ein fertiges Produkt, z. B. E-Mail-Dienste.

1.5.2 Warum ist die Cloud praktisch?

Sie können sofort mehr „Platz“ bekommen.

Sie bezahlen nur, was Sie tatsächlich nutzen.

Sie benötigen keinen eigenen Serverraum.

1.5.3 Grundbausteine in der Cloud

Compute = Rechenkraft („Du führst diese Aufgabe aus!“ sagt man hier zum System)

Storage = Speicherplatz (ein virtuelles Regal)

Netzwerk = Wege, auf denen Daten unterwegs sind

Sicherheit = Regeln, Türen und Schlösser

Typische Anbieter

Amazon AWS, Microsoft Azure, Google Cloud – große Anbieter, bei denen Sie IT „mieten“ können.

1.6 IT-Security Essentials

1.6.1 Was bedeutet IT-Sicherheit?

IT-Sicherheit schützt Computer und Daten vor Angriffen oder Fehlern – so wie Sie Ihr Haus abschließen, damit niemand Unbefugtes hineinkommt.

1.6.2 Die drei wichtigsten Ziele

Vertraulichkeit – nur berechtigte Personen dürfen Daten sehen

Integrität – niemand darf Daten heimlich verändern

Verfügbarkeit – Systeme sollen funktionieren, wenn Sie sie benötigen

1.6.3 Typische Gefahren

Phishing – jemand versucht, Sie zu täuschen

Malware – Schadsoftware wie ein Virus

Brute-Force-Angriffe – viele Passwortversuche hintereinander

Man-in-the-Middle – jemand hört eine Verbindung ab

1.6.4 Wie können Sie sich schützen?

Verwenden Sie starke Passwörter

Nutzen Sie Zwei-Faktor-Authentifizierung

Halten Sie Programme aktuell

Verschlüsseln Sie wichtige Daten

Vergeben Sie nur notwendige Rechte

1.6.5 Wichtige Konzepte

Hashing = ein unveränderlicher „Fingerabdruck“ von Daten

Zero Trust = niemandem automatisch vertrauen

Backups = Sicherheitskopien, falls etwas schiefgeht