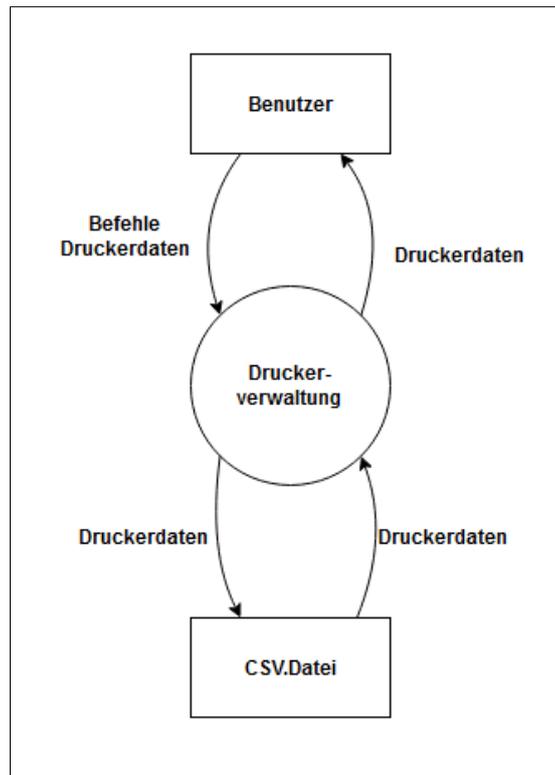
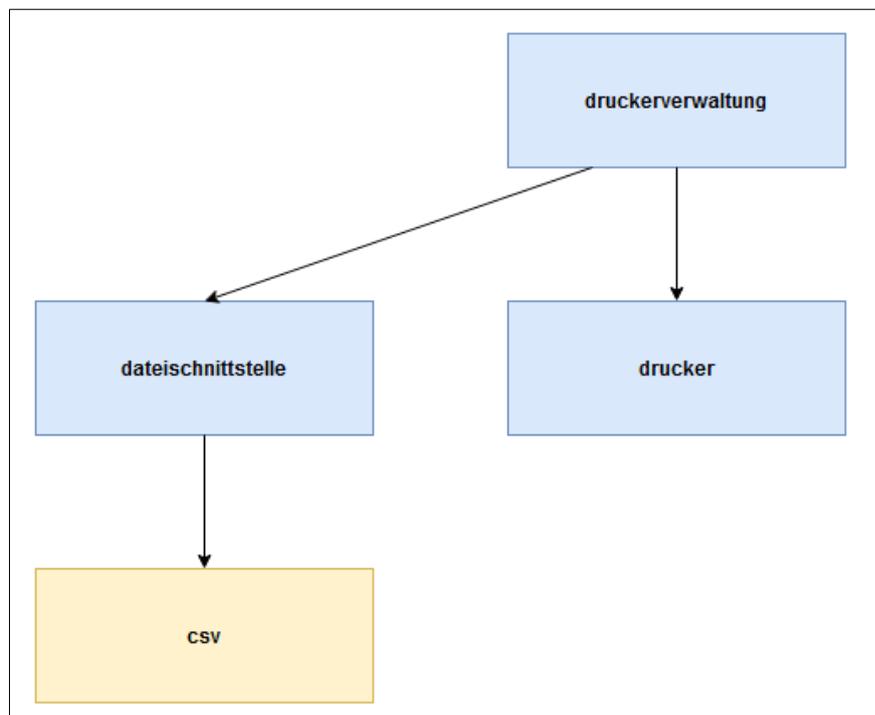


Aufgabe 2 - Lösungen

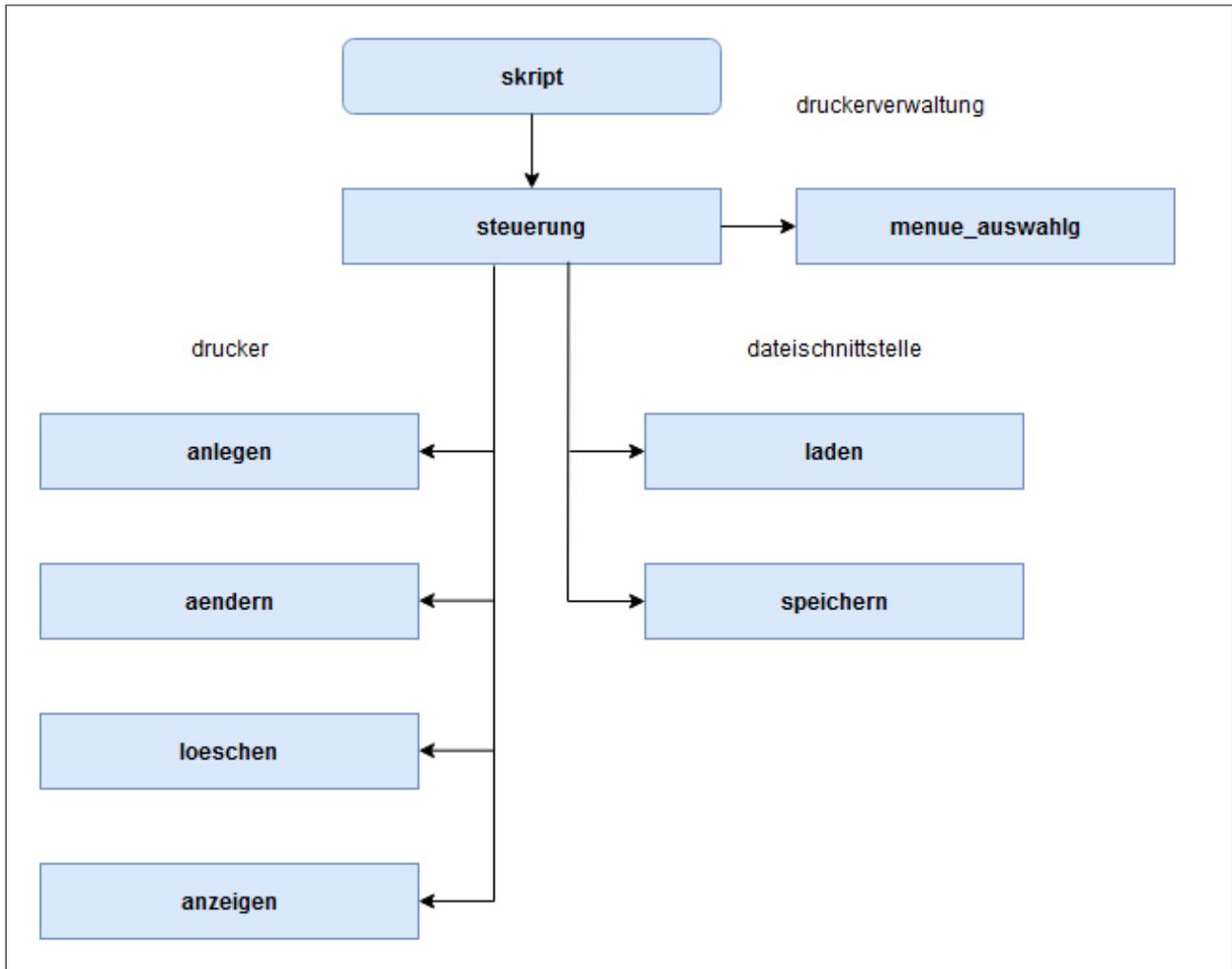
Druckerverwaltung - Kontext Diagramm



Druckerverwaltung - Modulhierarchie



Druckerverwaltung – Funktionshierarchie



Aufgabe 3 - Lösungen

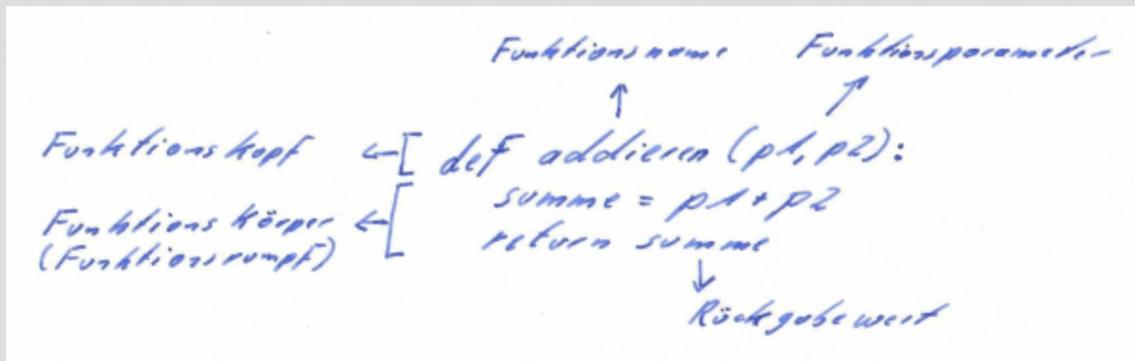
- a) Was versteht man bei der Programmierung unter einer Funktion?

Antwort:

Eine **Funktion** ist eine Zusammenfassung mehrerer Programmanweisungen unter einem Namen / Funktionsnamen.

- b) Schreiben Sie eine Python Funktion welche zwei Zahlen addiert. Die beiden Summanden werden als Parameter mitgegeben, die Summe als Return-Wert zurückgegeben.
c) Kennzeichnen Sie an der Funktion aus (b) die Teile Funktionskopf, Funktionskörper, Funktionsname und Funktionsparameter.

Antwort:



- d) Zeigen Sie mit einem Beispiel, wie man für Parameter Default-Werte festlegen kann.

Antwort:

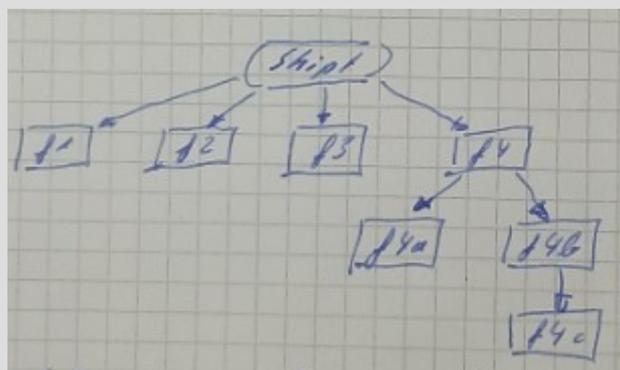
Default-Parameter sind Parameter, welche einen Standardwert besitzen. Dieser wird genutzt, wenn der Parameter beim Aufruf fehlt.

Beispiel:

```
def person_anlegen( vorname, nachname, land = "Deutschland")
```

- e) Visualisieren Sie die Funktionsaufrufhierarchie von `beispiel_ls3_02.py` mit einem geeigneten Diagramm.

Antwort:



f) Mit welchem Schlüsselwort werden Funktionsdefinitionen in Python gekennzeichnet?

Antwort:

Funktionen werden in Python mit dem Schlüsselwort **def** gekennzeichnet. `def` ist die Abkürzung für definition.

g) Python besitzt einen Satz von fest vordefinierten Funktionen (build in functions). Nennen Sie drei und beschreiben Sie kurz, was diese machen. Wo finden Sie Dokumentationen dieser Funktionen?

Antwort:

Beispiele:

`print()` → Ausgabe in der Konsole
`input()` → Eingabe in der Konsole
`len()` → Länge einer Liste oder eines Dictionaries
`float()` → Umwandlung in eine Kommazahl
...

Dokumentation:

Python Original Dokumentation: <https://docs.python.org/3/library/functions.html>
w3schools: https://www.w3schools.com/python/python_ref_functions.asp

h) Was versteht man unter einem Modul? Wie ist der Zusammenhang zwischen Modul und Datei in Python?

Antwort:

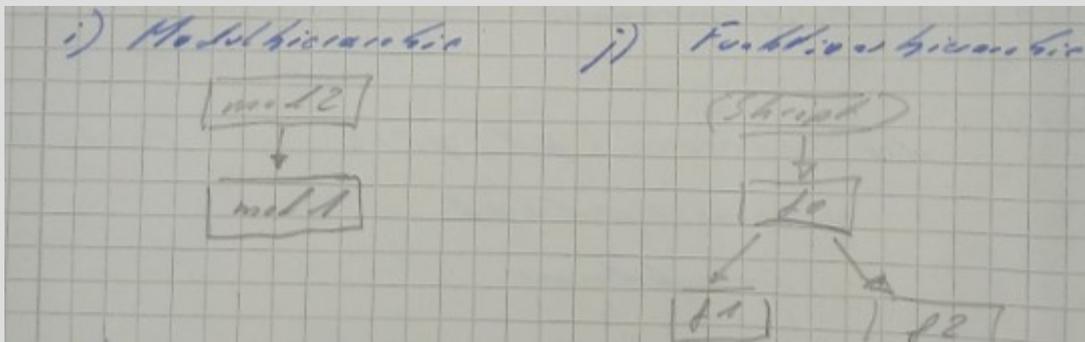
Ein **Modul** ist eine Zusammenfassung mehrerer inhaltlich zusammenhängender Funktionen. Ein Modul kann neben Funktionen auch noch andere Elemente enthalten, wie z.B. Klassen oder Konstanten.

In Python ist jede Datei ein Modul und ein Modul eine Datei. **Modul == Datei** In anderen Programmiersprachen kann dies anders sein.

i) Visualisieren Sie die Modulaufrufhierarchie von `beispiel_ls3_03....py` mit einem geeigneten Diagramm.

j) Visualisieren Sie die Funktionsaufrufhierarchie von `beispiel_ls3_03....py` mit einem geeigneten Diagramm.

Antwort:



k) Mit welcher Anweisung werden andere Module in Python eingebunden?

Antwort:
Import-Anweisung

Aufgabe 4 - Lösungen

- a) Was ist der Unterschied zwischen einem White-Box-Test und einem Black-Box-Test?

Antwort:

Ein **White-Box-Test** ist ein Testverfahren, bei dem der Tester Kenntnisse über die interne Funktionsweise des zu testenden Systems hat. Der Tester hat Zugriff auf den Quellcode und kann gezielt Testfälle entwickeln, um bestimmte Pfade oder Bedingungen im Code zu überprüfen. Ein **Black-Box-Test** hingegen basiert auf der funktionalen Spezifikation des Systems, ohne Kenntnis der internen Implementierung. Der Tester betrachtet das System als eine "Black Box" und testet die Ein- und Ausgabewerte, um die erwarteten Ergebnisse zu überprüfen.

- b) Was versteht man unter einem Komponenten-Test?

Antwort:

Ein **Komponenten-Test** ist ein Testverfahren, bei dem einzelne Komponenten (Module, Funktionen, Klassen) eines Systems isoliert getestet werden. Es zielt darauf ab, die Funktionalität und Korrektheit jeder einzelnen Komponente sicherzustellen. Dieser Test wird normalerweise von den Entwicklern selbst durchgeführt, um sicherzustellen, dass jede Komponente unabhängig von anderen Komponenten ordnungsgemäß funktioniert.

- c) Was unterscheidet Komponenten-Test und Integrations-Test?

Antwort:

Ein **Integrations-Test** ist ein Testverfahren, bei dem verschiedene Komponenten eines Systems zusammengeführt und getestet werden, um sicherzustellen, dass sie ordnungsgemäß miteinander interagieren. Im Gegensatz zum Komponenten-Test wird hier die Integration der Komponenten und die Kommunikation zwischen ihnen überprüft.

- d) Was versteht man unter einem Regressionstest?

Antwort:

Ein **Regressionstest** ist ein automatisiertes Testverfahren, bei dem bereits getestete Komponenten erneut getestet werden, um sicherzustellen, dass nach Änderungen oder Erweiterungen keine neuen Fehler eingeführt wurden. Es dient dazu, sicherzustellen, dass das System weiterhin wie erwartet funktioniert und keine Regression (Rückentwicklung) aufgetreten ist.

e) Nennen Sie Werkzeuge, welche es zur Testautomatisierung bei Python gibt?

Antwort:

Testautomatisierung bezieht sich auf den Einsatz von Softwarewerkzeugen und Technologien, um den Testprozess zu automatisieren. Dabei werden Testfälle und -skripte erstellt, die automatisch ausgeführt werden können, ohne dass menschliche Eingriffe erforderlich sind.

Testautomatisierung wird aus mehreren Gründen durchgeführt:

Effizienz: Durch die Automatisierung von Tests können sich wiederholende Aufgaben schneller und effizienter erledigt werden, was Zeit und Ressourcen spart.

Wiederholbarkeit: Automatisierte Tests können beliebig oft wiederholt werden, um sicherzustellen, dass die Funktionalität des Systems konsistent bleibt.

Skalierbarkeit: Bei zunehmender Komplexität des Systems oder der Anwendung können automatisierte Tests leichter angepasst und erweitert werden, um neue Funktionen oder Szenarien abzudecken.

Zuverlässigkeit: Durch die Automatisierung von Tests wird menschliches Versagen minimiert und die Zuverlässigkeit der Testergebnisse verbessert.

Es gibt mehrere Werkzeuge zur Testautomatisierung bei Python, darunter **Pytest** und **unittest**.

f) Welche Fehler werden typischerweise von einem Compiler erkannt?

Antwort:

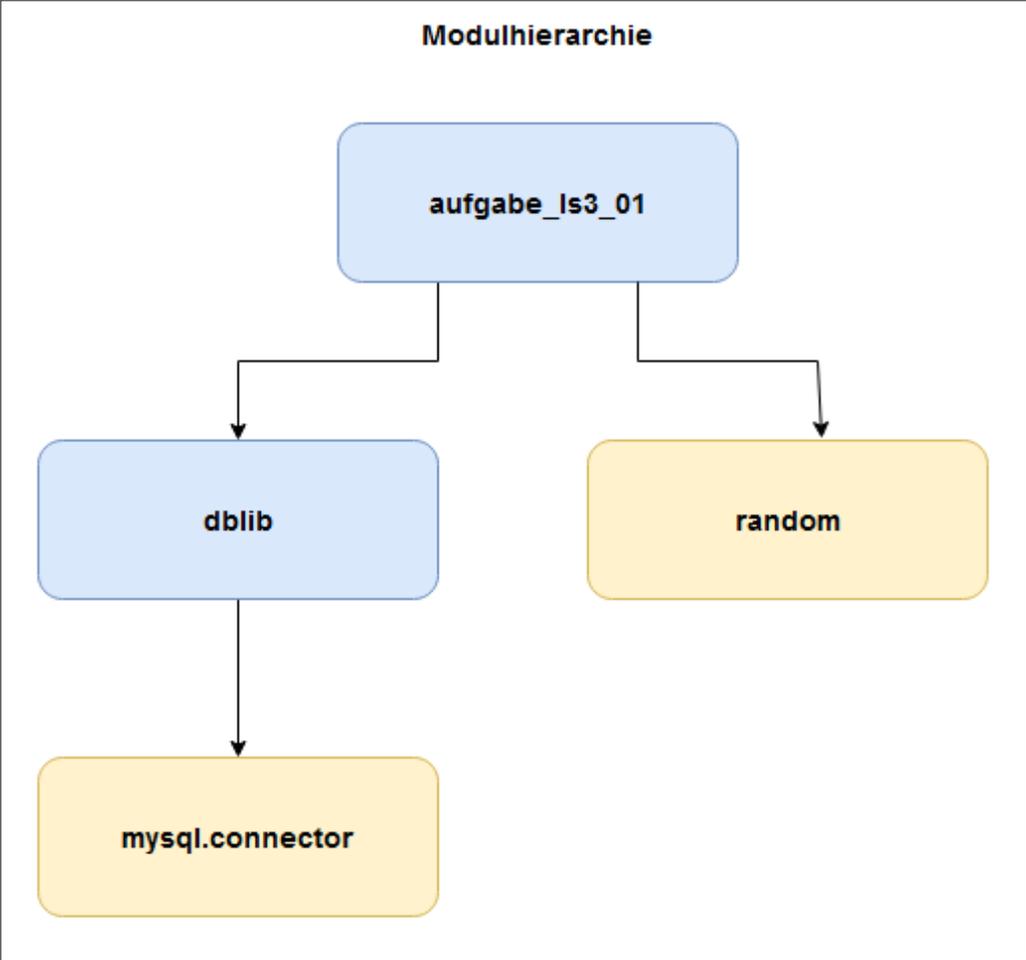
Typische Fehler, die von einem Compiler erkannt werden, sind **Syntaxfehler**, wie zum Beispiel fehlende oder falsch platzierte Klammern, die Verwendung einer Variable, die nicht deklariert wurde, und Typfehler, wie zum Beispiel die Zuweisung eines Wertes eines falschen Typs zu einer Variablen.

g) Was versteht man unter Test-Driven-Development?

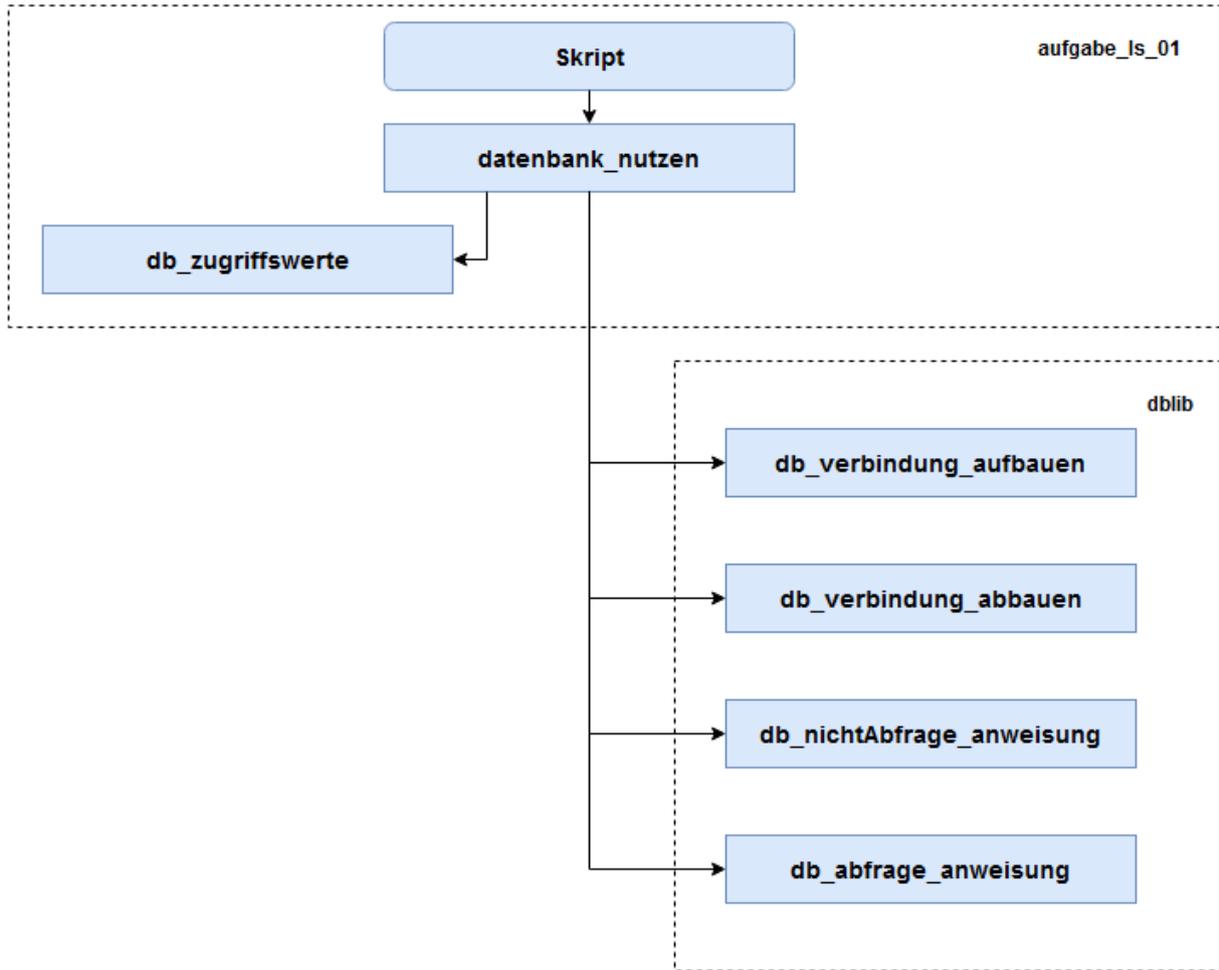
Antwort:

Test-Driven-Development (TDD) ist eine Entwicklungspraktik, bei der der Entwickler zuerst einen Testfall schreibt, der die gewünschte Funktionalität beschreibt, und dann den Code schreibt, um den Testfall zu bestehen. Der Entwickler iteriert diesen Prozess, indem er neue Testfälle schreibt und den Code verbessert, um alle Testfälle zu bestehen. TDD fördert eine bessere Testabdeckung und hilft, Fehler frühzeitig im Entwicklungsprozess zu erkennen.

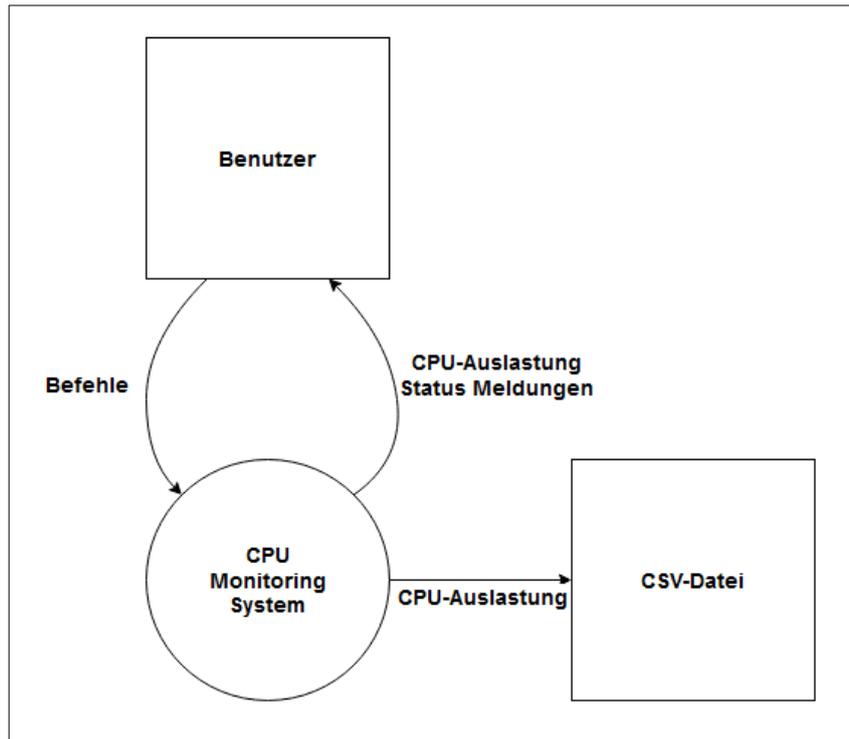
Aufgabe 5 – Lösungen



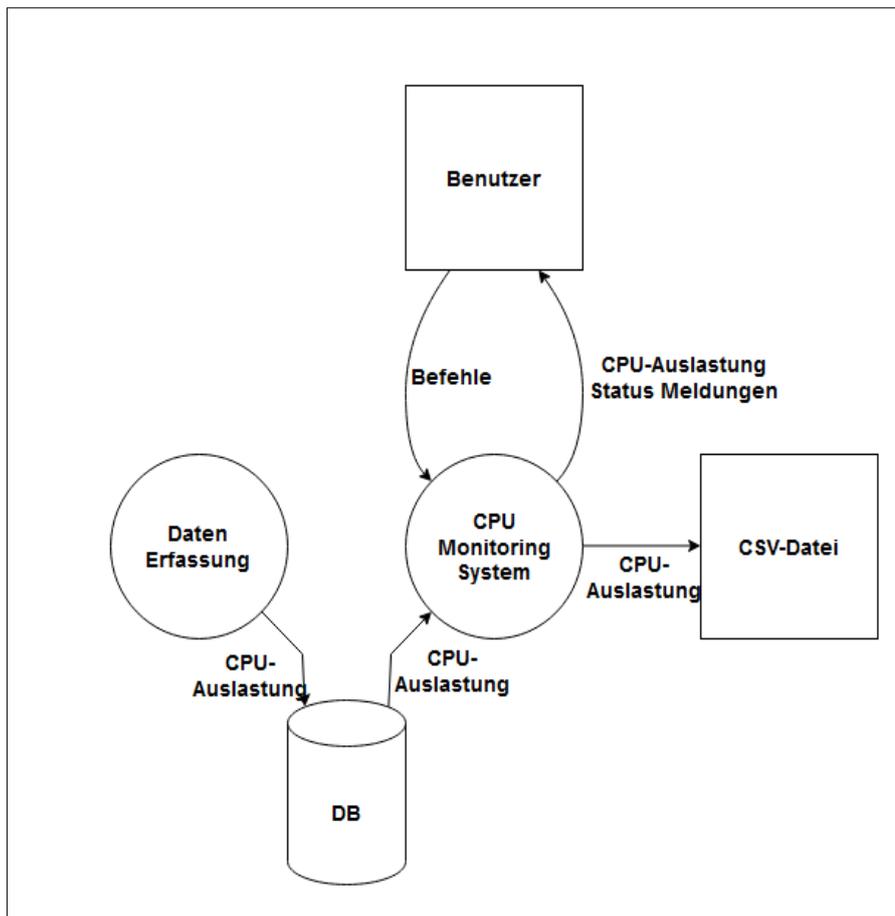
Funktionshierarchie



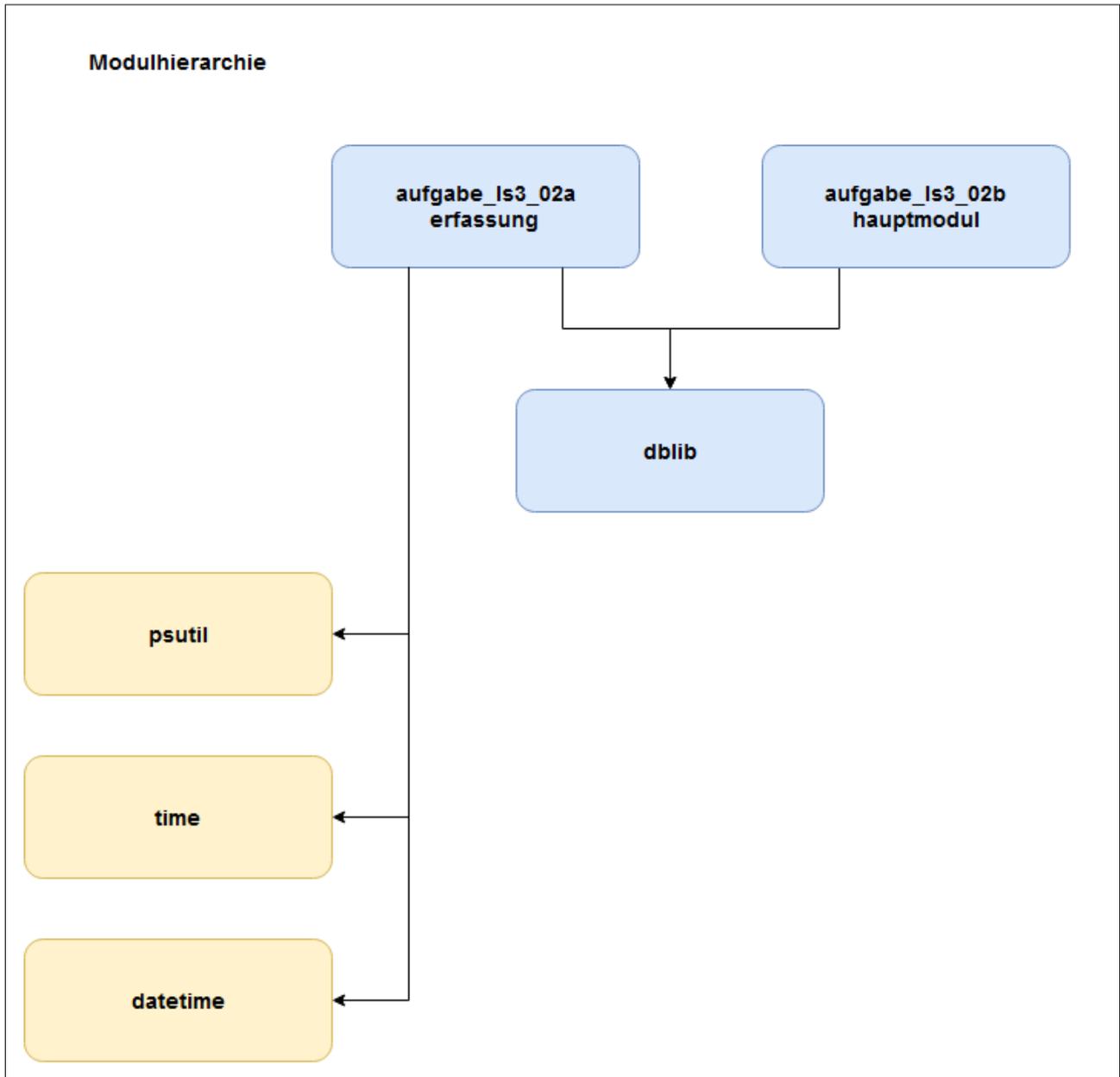
Abgabe 6 – Kontextdiagramm (Grob)



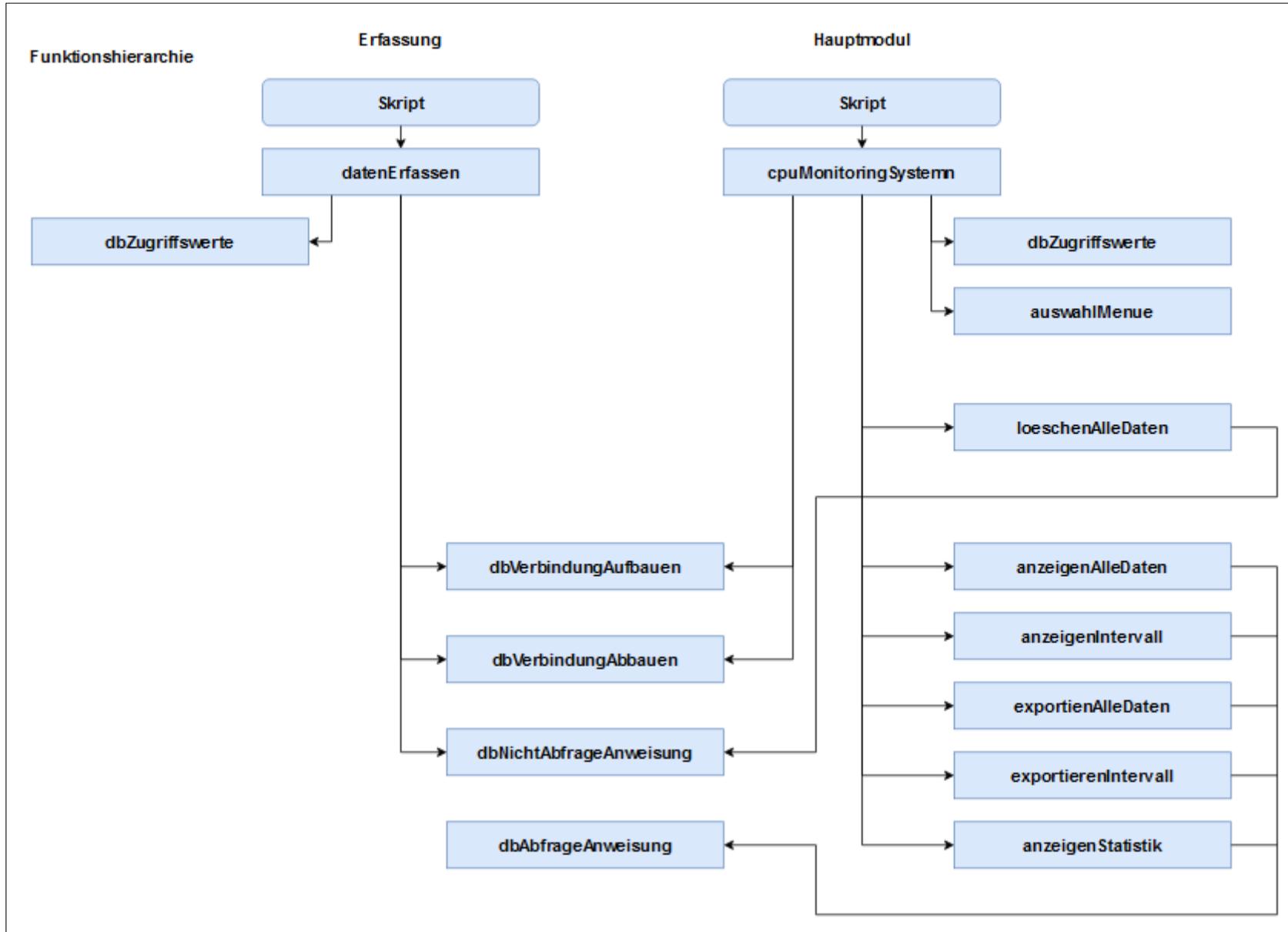
Abgabe 6 – Kontextdiagramm (Verfeinert)



Abgabe 6 – Modulhierarchie



Abgabe 6 – Funktionshierarchie



Aufgabe 7 – Lösungen

- a) Was versteht man unter einer Datenbank? Was versteht man unter einem Datenbankmanagementsystem? Was versteht man unter einem Datenbanksystem?

Antwort:

Eine **Datenbank (DB)** ist eine organisierte Sammlung von Daten, die in einer strukturierten Form gespeichert, verwaltet und abgerufen werden können.

Ein **Datenbankmanagementsystem (DBMS)** ist eine Software, die verwendet wird, um Datenbanken zu erstellen, zu verwalten und darauf zuzugreifen.

Ein **Datenbanksystem (DBS)** bezieht sich auf die Kombination aus einer oder mehrerer Datenbanken und einem Datenbankmanagementsystem.

- b) In der Praxis werden häufig sogenannte relationale Datenbanken eingesetzt. Was versteht man hierunter?

Antwort:

Eine **relationale Datenbank** ist eine Art von Datenbank, die auf dem relationalen Modell basiert. In einer relationalen Datenbank werden Daten in Tabellen organisiert, die miteinander in Beziehung stehen.

- c) Welche relationalen Datenbanken kennen Sie (Hersteller/Produkt)?

Antwort:

Einige bekannte relationale Datenbanken sind:

Oracle Database (Oracle)

Microsoft SQL Server (Microsoft)

MySQL (Oracle Open-Source-Projekt)

MariaDB ((Community-basiertes Open-Source-Projekt)

PostgreSQL (Community-basiertes Open-Source-Projekt)

IBM DB2 (IBM)

- d) Was ist die Beziehung zwischen MySQL und MariaDB?

Antwort:

MySQL und **MariaDB** sind beides relationale Datenbankmanagementsysteme. MariaDB ist eine Abspaltung (Fork) von MySQL, der von den ursprünglichen Entwicklern von MySQL erstellt wurde. Beide DBMS sind ähnlich und noch teilweise kompatibel, aber entwickeln sich langsam etwas unterschiedlich.

- e) Was ist XAMPP?

Antwort:

XAMPP ist ein Kombipaket (**Bundle**), die entwickelt wurde, um die Einrichtung und Verwaltung einer **lokalen Webentwicklungsumgebung** zu erleichtern. Es enthält Apache (Webserver), MySQL oder MariaDB (Datenbank), PHP (Programmiersprache) und Perl (Skriptsprache).

- f) Mit welchem in XAMPP enthaltenen Werkzeug können Sie die Datenbank administrieren?

Antwort:

Mit dem in XAMPP enthaltenen Werkzeug **phpMyAdmin** kann man die Datenbank administrieren. PhpMyAdmin ist eine webbasierte Anwendung, die eine grafische Benutzeroberfläche zum Verwalten von MySQL-Datenbanken bietet.

- g) Gibt es auch nicht-rationale Datenbanken? Wenn ja, nennen Sie eine und worin unterscheidet sich diese von einer relationalen Datenbank?

Antwort:

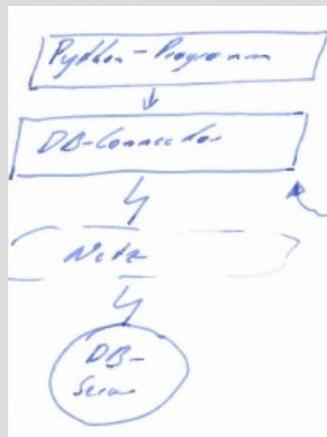
Ja, es gibt auch nicht-relationale Datenbanken.

Ein Beispiel dafür ist eine **dokumentenorientierte Datenbank** wie **MongoDB**. Im Gegensatz zu relationalen Datenbanken speichern dokumentenorientierte Datenbanken Daten in Dokumenten statt in Tabellen. Die Daten werden in JSON-ähnlicher Form gespeichert und es gibt keine festen Tabellenschemata.

Ein anderes Beispiel sind **Zeitreihendatenbanken** wie **InfluxDB**. Eine Zeitreihendatenbank ist darauf spezialisiert, große Mengen von Daten zu speichern und zu verwalten, die über einen bestimmten Zeitraum gesammelt wurden und eine zeitliche Komponente haben. Sie ermöglicht effiziente Speicherung, Abruf und Analyse von Daten, die in der Reihenfolge ihres Auftretens organisiert sind.

- h) Skizzieren Sie den Aufbau eines Python Programms, welches auf eine MariaDB-Datenbank zugreift.

Antwort:



- i) Mit welcher Sprache programmiert man Zugriffe auf relationale Datenbanken? Welche Abkürzung wird hierfür verwendet und wofür steht die Abkürzung?

Antwort:

Man programmiert Zugriffe auf relationale Datenbanken mit der Sprache **SQL (Structured Query Language)**. SQL ist eine standardisierte Sprache, die verwendet wird, um Datenbanken zu erstellen, zu verwalten und abzufragen.

j) Mit welcher Anweisung kann man Daten in einer Datenbank abspeichern?

Antwort:
INSERT-Anweisung

k) Mit welcher Anweisung kann man Daten aus einer Datenbank auslesen?

Antwort:
SELECT-Anweisung

l) Welches Modul müssen Sie in Python importieren, um auf eine MySQL-/MariaDB-Datenbank zuzugreifen?

Antwort:
Um auf eine MySQL-/MariaDB-Datenbank zuzugreifen, muss ein sogenannter Connector in Python importiert werden. Dieser dient zum Verbindungsaufbau mit der Datenbank und zum Absetzen von SQL-Anweisungen.
`import mysql.connector`
Das entsprechende Paket muss vorher mit pip installiert worden sein.