

# LS 3 – CPU-Monitoring-System

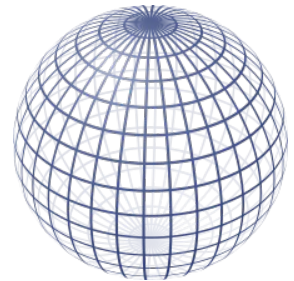
Die CPU-Auslastung soll über einen längeren Zeitraum aufgezeichnet werden. Die Auslastung lässt sich mit Python herausbekommen. Für die dauerhafte Aufzeichnung kann man eine Datenbank nutzen. Können Python und Datenbank zusammenarbeiten?



## Teil A: Modulare Zerlegung

### Aufgabe 1a - Kugelberechnung (I)

Erstellen Sie ein Programm, welches nach Eingabe des Radius die Oberfläche und das Volumen einer Kugel ausrechnet. Die Berechnungen soll jeweils in einer separaten Unterfunktion erfolgen. Das Einlesen des Radius und die Ausgaben sollen im Hauptprogramm erfolgen. Speichere das Programm unter dem Namen **Aufgabe\_ls3\_04a.py** ab.



### Aufgabe 1b - Kugelberechnung (II)

Modifizieren Sie das Programm aus (1a) so, dass das Einlesen des Radius und die Ausgaben ebenso in eigenen Unterprogrammen erfolgen. Speichere das Programm unter dem Namen **Aufgabe\_ls3\_04b.py** ab.

### Aufgabe 1c - Kugelberechnung (III)

Modifizieren Sie das Programm aus (1b) so, dass es auf 3 Module für Steuerung, Funktionalität und Benutzerinteraktion aufgeteilt wird. Speichere das Programm unter dem Namen **Aufgabe\_ls3\_04c...py** ab.

## Aufgabe 2 – Modulare Druckerverwaltung

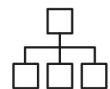


Erstellen Sie eine modulare Druckerverwaltung. Zerlegen Sie hierzu die Druckerverwaltung aus LS2 in Module und Funktionen, so dass sie den folgenden strukturellen Anforderungen entspricht:

- Die Druckerverwaltung soll aus drei Modulen bestehen.
- Das Modul `dateischnittstelle` soll die beiden Funktionen `laden` und `speichern` enthalten. Die Funktion `laden` lädt die in der CSV-Datei enthaltenen Daten in die `drucker_liste`. Die Funktion `speichern` speichert den Inhalt der `drucker_liste` in der CSV-Datei.
- Das Modul `drucker` beinhaltet die Funktionen `anlegen`, `aendern`, `loeschen` und `anzeigen`.
- Das Modul `druckerverwaltung` beinhaltet die Funktion `steuerung` (Hauptprogramm) und die Funktion `menue_auswahl`.

Speichern Sie das Programm unter dem Namen **`aufgabe_ls3_03...py`** ab. Dokumentieren Sie die Modul- und Funktionsaufrufhierarchie in geeigneten Diagrammen. Dokumentieren Sie die Steuerfunktion mit einem Struktogramm.

## Aufgabe 3 – Reflexion Modularisierung



- Was versteht man bei der Programmierung unter einer Funktion?
- Schreiben Sie eine Python Funktion welche zwei Zahlen addiert. Die beiden Summanden werden als Parameter mitgegeben, die Summe als Return-Wert zurückgegeben.
- Kennzeichnen Sie an der Funktion aus (b) die Teile Funktionskopf, Funktionskörper, Funktionsname und Funktionsparameter.
- Zeigen Sie mit einem Beispiel, wie man für Parameter Default-Werte festlegen kann.
- Visualisieren Sie die Funktionsaufrufhierarchie von `beispiel_ls3_02.py` mit einem geeigneten Diagramm.
- Mit welchem Schlüsselwort werden Funktionsdefinitionen in Python gekennzeichnet?
- 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?
- Was versteht man unter einem Modul? Wie ist der Zusammenhang zwischen Modul und Datei in Python?
- Visualisieren Sie die Modulaufrufhierarchie von `beispiel_ls3_03...py` mit einem geeigneten Diagramm.
- Visualisieren Sie die Funktionsaufrufhierarchie von `beispiel_ls3_03...py` mit einem geeigneten Diagramm.
- Mit welcher Anweisung werden andere Module in Python eingebunden?

**Aufgabe 4 – Reflexion Testen**

- Was ist der Unterschied zwischen einem White-Box-Test und einem Black-Box-Test?
- Was versteht man unter einem Komponenten-Test?
- Was unterscheidet Komponenten-Test und Integrations-Test?
- Was versteht man unter einem Regressionstest?
- Nennen Sie Werkzeuge, welche es zur Testautomatisierung bei Python gibt?
- Welche Fehler werden typischerweise von einem Compiler erkannt?
- Was versteht man unter Test-Driven-Development?



## Teil B: Datenbankanbindung

**Aufgabe 5 – Datenbankfunktionen extrahieren**

Bauen Sie `aufgabe_ls3_00.py` so um, dass die 4 allgemeinen Zugriffsfunktionen in ein eigenes Modul ausgelagert sind, welches importiert wird. Die Funktion `dbZugriffsWerte` ist spezifisch und bleibt. Verlagern Sie den restlichen Skript-Code in eine Funktion. Speichern Sie das Datenbankmodul unter dem Namen **`dblib.py`** ab, das umgebaute Beispiel soll unter dem Namen **`aufgabe_ls3_01.py`** dokumentieren Sie die Modul- und Funktionsaufrufhierarchie.

**Aufgabe 6 – CPU-Monitoring-System**

Die CPU-Auslastung soll über einen längeren Zeitraum aufgezeichnet werden. Hierzu ist ein Aufzeichnungssystem zu erstellen, welches aus mehreren Komponenten besteht:

- Einer zentralen Datenbank zur Speicherung der Auslastungswerte
- Ein Python Programm, welches in regelmäßigen Intervallen die CPU-Auslastung ermittelt und diese in der Datenbank abspeichert.
- Ein Python Programm, welches die in der Datenbank enthaltenen Werte anzeigt. Wenn möglich sollen sinnvolle Auswahlkriterien unterstützt werden.
- Ein Python Programm, welches die in der Datenbank enthaltenen Werte als CSV-Datei exportiert, so dass eine Weiterverarbeitung und Visualisierung mit Hilfe eines Tabellenkalkulationsprogramms möglich ist.

Erstellen Sie eine aktuelle Übersichtsdokumentation bestehend aus Kontextdiagramm, Blockdiagramm mit Modulen sowie der Funktionsaufrufhierarchie.



Mögliche Erweiterungen:

- Erweitern Sie Komponente 3, so dass neben der Anzeige auch eine einfache statistische Auswertung möglich ist.
- Erweitern Sie Komponente 3, so dass auch ein Löschen der Daten möglich ist.
- Erweitern Sie das System so, dass neben der CPU-Auslastung noch andere Kenngrößen Ihrer Wahl abgespeichert und angezeigt werden

Hinweise:

- Benutzen XAMP Portable, zum Aufbau einer MySQL- bzw. MariaDB-Datenbank.
- Die Komponenten 3 und 4 können zu einer zusammengefasst werden.
- Speichern Sie die Programme unter dem Namen **aufgabe\_ls3\_02a.py**, **aufgabe\_ls3\_02b.py**, ... ab. Das Skript zur Erstellung der Datenbank soll **aufgabe\_ls3\_02.sql** heißen.

### Aufgabe 7 – Reflexion Datenbanken



- Was versteht man unter einer Datenbank? Was versteht man unter einem Datenbankmanagementsystem? Was versteht man unter einem Datenbanksystem?
- In der Praxis werden häufig sogenannte relationale Datenbanken eingesetzt. Was versteht man hierunter?
- Welche relationalen Datenbanken kennen Sie (Hersteller/Produkt)?
- Was ist die Beziehung zwischen MySQL und MariaDB?
- Was ist XAMPP?
- Mit welchem in XAMPP enthaltenen Werkzeug können Sie die Datenbank administrieren?
- Gibt es auch nicht-rationale Datenbanken? Wenn ja, nennen Sie eine und worin unterscheidet sich diese von einer relationalen Datenbank?
- Skizzieren Sie den Aufbau eines Python Programms, welches auf eine MariaDB-Datenbank zugreift.
- Mit welcher Sprache programmiert man Zugriffe auf relationale Datenbanken? Welche Abkürzung wird hierfür verwendet und wofür steht die Abkürzung?
- Mit welcher Anweisung kann man Daten in einer Datenbank abspeichern?
- Mit welcher Anweisung kann man Daten aus einer Datenbank auslesen?
- Welches Modul müssen Sie in Python importieren, um auf eine MySQL-/MariaDB-Datenbank zuzugreifen?

# Teil C: Zusatzaufgaben



## Aufgabe 8 - Lambda-Funktionen

In Python gibt es Lambda-Funktionen. Erarbeiten Sie ein Beispiel, aus dem hervorgeht, was Lambda-Funktionen sind und wie man sie nutzen kann.

 $\lambda$ 

## Aufgabe 9 - Druckerverwaltung mit graphischer Benutzeroberfläche



Bauen Sie den Druckerverwaltung aus Aufgabe 2 so um, dass sie anstellen der Konsolen-Benutzerschnittstelle eine graphische Benutzeroberfläche besitzt.