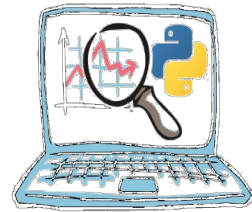


LS 2 – Nutzungsdaten eines IT-Systems auswerten und graphisch zu Verfügung stellen



Ausgangssituation

Nachdem, die Nutzungsdaten eines IT-Systems in einer zentralen Datenbank zusammengefügt wurden, soll deren Nutzung vereinfacht werden. Hierzu sollen die Daten visualisiert und mit Hilfe einer graphischen Benutzeroberfläche angezeigt werden.

Teil A: Einarbeitung in die GUI-Programmierung mit Tkinter



Aufgabe 1 – Einführungsvideos und Beispiele

Sehen Sie sich die beiden Einführungsvideos zur GUI-Programmierung mit Tkinter an und machen Sie sich mit den vorgeführten Beispielen `beispiel_ls2_01a.py` und `beispiel_ls2_01b.py` vertraut.

- a) Bringen Sie die Beispiele bei sich zum Laufen.
- b) Beantworten Sie die folgenden Fragen zu `beispiel_ls2_01a.py`.
 - Wie groß ist das Hauptfenster?
 - Welche unterschiedlichen Fensterelemente werden in dem Hauptfenster angelegt? Wie heißen die dabei benutzten Klassen? In welchen Zeilen findet die jeweilige Initialisierung statt?
 - Wann wird die Funktion `dividieren` aufgerufen? In welcher Zeile wird dies angegeben?
 - Mit Hilfe welcher Klasse wird das Hauptfenster initialisiert? Welche Methoden dieser Klasse werden alles genutzt?
 - Mit Hilfe welcher Methode(n) werden die einzelnen Fensterelemente im Hauptfenster positioniert? Welche Parameter muss man jeweils mitgeben? Wie heißt diese Art der Positionierung?
- c) Beantworten Sie die folgenden Fragen zu `beispiel_ls2_01b.py`.
 - Mit Hilfe welcher Methode(n) werden die einzelnen Fensterelemente im Hauptfenster positioniert? Welche Parameter muss man jeweils mitgeben? Wie heißt diese Art der Positionierung?
 - Viele Fensterelemente werden mit Hilfe von Klassen aus dem Subsystem `tk` gebildet. Was unterscheidet diese Klassen von den ursprünglichen?
 - In welcher Methode wird das Fenster mit seinen Elementen aufgebaut? Wo wird diese Methode aufgerufen?
 - Wenn der Konstruktor der Klasse `Taschenrechner` aufgerufen wird, werden verschiedene weitere Methoden aufgerufen. Visualisieren Sie deren Aufrufhierarchie. Methoden- und Funktionsaufrufe, die in andere Dateien verweisen, brauchen nicht

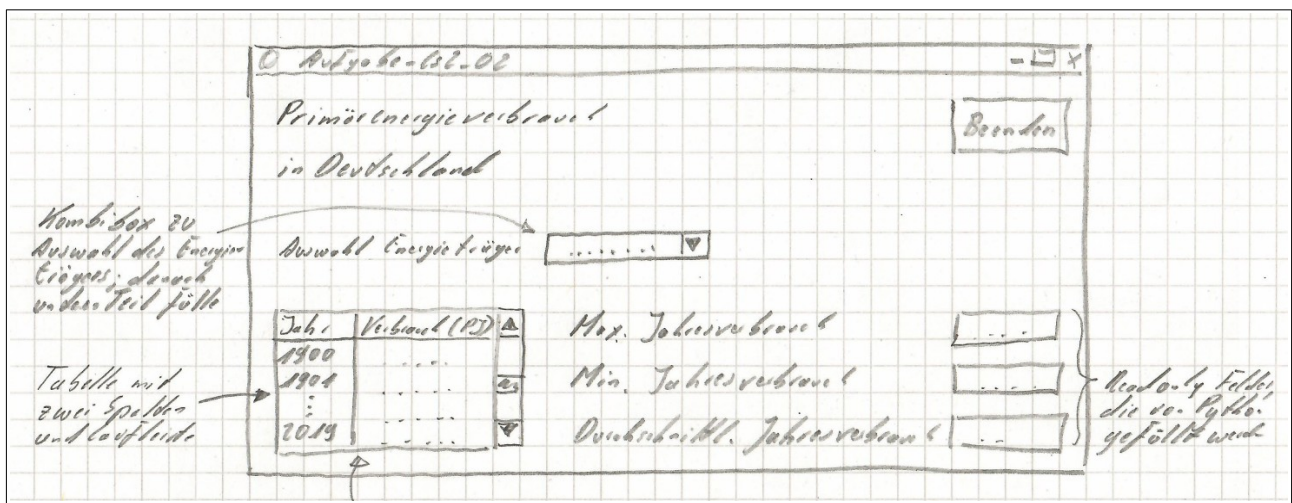
weiter verfolgt werden.

- Die Tkinter Elemente der Benutzeroberfläche bilden ausgehend vom Hauptfenster eine Hierarchie. Stellen Sie diese graphisch dar.
- d) Führen Sie `beispiel_ls2_01b.py` folgende Änderungen und Erweiterungen durch
 - Ergänzen Sie unten rechts einen „Beenden-Knopf“. Wenn auf diesen Knopf gedrückt wird, soll sich die Anwendung beenden.

Aufgabe 2 - GUI-Anzeige des Primärenergieverbrauchs



In der Datei `energiea.csv` sind Daten zum Primärenergieverbrauchs in Deutschland enthalten. Die Angaben sind jeweils in Peta-Joule (PJ). Programmieren Sie mit Python und Tkinter eine GUI zur Anzeige der Daten. Eine Skizze der Anzeige ist beigelegt.



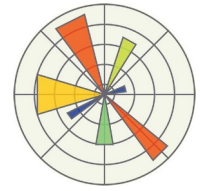
Hinweise:

- Strukturieren Sie Ihre Lösung objekt-orientiert. Orientieren Sie sich dabei an dem Beispiel `beispiel_ls2_01b.py`.
- Um Ihnen das Leben zu erleichtern, ist bereits ein Anfangsrahmen für das Programm vorhanden (`aufgabe_ls2_02_Ausgangsbasis.py`).
- Nutzen Sie wo vorhanden die neueren themed-Klassen von Tkinter.
- Nutzen Sie die Grid-Technologie zur Gestaltung.
- Da Tkinter kein Tabellen-Widget besitzt, können Sie eine Liste zur Simulation nutzen.
- Speichern Sie das Programm unter dem Namen **`aufgabe_ls2_02.py`** ab.

Zusatzaufgaben:

- Stellen Sie das Energiebild neben der Aufgabennummer (`energy.jpg`) auf dem GUI dar.
- Die Tkinter Elemente der Benutzeroberfläche bilden ausgehend vom Hauptfenster eine Hierarchie. Stellen Sie diese graphisch dar.

Teil B: Einarbeitung in Matplotlib



Aufgabe 3 – Visualisierung des Primärenergieverbrauchs

In der Datei `energiea.csv` sind Daten zum Primärenergieverbrauchs in Deutschland enthalten. Die Angaben sind jeweils in Peta-Joule (PJ). Programmieren Sie mit Python unter Nutzung der Module Pandas und Matplotlib folgende Visualisierungen:

- Visualisieren Sie den Verbrauch von erneuerbaren Energien von 1990 bis 2019 als Säulendiagramm.
- Visualisieren Sie den Verbrauch von Steinkohle, Braunkohle und Mineralölen von 1990 bis 2019 in einem Liniendiagramm.
- Visualisieren Sie für 2019 die Energieverteilung in einem Kuchendiagramm.

Hinweise:

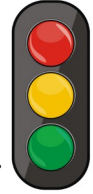
- Orientieren Sie sich bei der Lösung an den zur Verfügung gestellten Beispielen.
- Versehen Sie alle Diagramme wo sinnvoll mit Titeln, Achsenbeschriftungen und Legenden.
- Speichern Sie das Programm unter dem Namen **`aufgabe_ls2_03.py`** ab.

Aufgabe 4 – Reflexion Matplotlib

- Wozu dient das Paket /die Module Matplotlib?
- Welche Programmierschnittstellen bietet Matplotlib und welche ist zur Programmierung empfohlen?
- Wie müssen die Daten bei der Nutzung von Matplotlib zur Verfügung gestellt werden?
- Beim Arbeiten mit Matplotlib haben Sie es mit Objekten der Klassen `Figure`, `Axes` und `Axis` zu tun. Beschreiben Sie kurz, was hinter den einzelnen Klassen steckt und wie der Zusammenhang zwischen ihnen ist.
- Welche Methode müssen Sie nutzen um ein Liniendiagramm zu zeichnen?
- Welche Methoden müssen Sie nutzen, um Balken- und Säulendiagramme zu zeichnen?
- Die Variable `y`-werte soll die Daten 2, 0, 2, 2 als Numpy Feld enthalten. Wie sieht der Programm Code aus?
- Mit welchem Befehl können Sie sich einen Rahmen und eine Zeichenfläche erstellen?
- Mit welchem Befehl wird letztendlich das Bild sichtbar gemacht?
- Die Klasse für Zeichenflächen bietet Methoden zur Achsenbeschriftung und zur setzen eines Titels an. Wie heißen diese? Geben Sie Beispielcode an, in dem alle drei Bestandteile gesetzt werden.
- Kann ein Rahmen auch mehrere Zeichenflächen haben? Wenn ja, geben Sie Beispiel hierfür an.
- Wie können Sie Bild speichern? In welchem Format wird es standardmäßig gespeichert?

Teil C: Vertiefung

Aufgabe 5 - Verfügbarkeit von Internet-Servern überwachen



Der Zustand der für den Schulbetrieb wichtigen Server soll durch eine kleine GUI angezeigt werden. Es soll jeweils der Name des Servers und ampelartig der Zustand angezeigt werden. Überwacht werden soll:

- Schulhomepage – www.bkgut.de
- WebUntis – mese.webuntis.com
- Google – www.google.de
- Wikipedia – www.wikipedia.de

Hinweise:

- Strukturieren Sie Ihre Lösung objekt-orientiert. Erstellen Sie eine Klasse für das GUI und eine Klasse für die Serverüberwachung.
- Nutzen Sie wo vorhanden die neueren themed-Klassen von Tkinter.
- Nutzen Sie die Grid-Technologie zur Gestaltung.
- Speichern Sie das Programm unter dem Namen **aufgabe_ls2_05.py** ab.

Zusatzaufgaben:

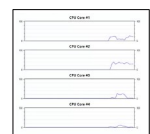
- Überwachen Sie zusätzlich noch Logineo und den öffentlichen Netmanzugang
- Dokumentieren Sie Ihre Anwendung mit einem UML-Klassendiagramm.

Aufgabe 6 - Vertiefung Matplotlib



- Visualisieren Sie die Einwohnerzahlen der deutschen Bundesländer mit jeweils einem
 - Balkendiagramm,
 - Säulendiagramm und
 - KuchendiagrammBeschriften Sie jedes Diagramm sinnvoll.
Speichern Sie das Programm unter dem Namen **aufgabe_ls2_06.py** ab.
- In der Datei `alpa_20220910.txt` befinden sich unter anderem die Alpha- und Beta-Werte eines Produktionsprozesses. Visualisieren Sie den Zusammenhang als Streudiagramm (scatter diagram). Beschriften Sie jedes Diagramm sinnvoll. Entfernen Sie unvollständige und doppelte Einträge.
Speichern Sie das Programm unter dem Namen **aufgabe_ls2_07.py** ab.

Aufgabe 7 - CPU-Auslastung anzeigen



Erstellung Sie mit Python eine GUI-Anwendung, welche die aktuelle CPU-Auslastung anzeigt. Die CPU-Auslastung soll alle 5 Sekunden neu abgefragt werden. Zusätzlich soll die CPU-Auslastung der letzten 10 Min graphisch als Liniendiagramm angezeigt werden.

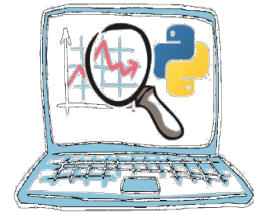
Hinweise:

- Strukturieren Sie Ihre Lösung objekt-orientiert. Erstellen Sie eine Klasse für das GUI und eine Klasse für die Bereitstellung der CPU-Daten.
- Nutzen Sie wo vorhanden die neueren themed-Klassen von Tkinter.
- Nutzen Sie die Grid-Technologie zur Gestaltung.
- Nutzen Sie `Matplotlib` zur Anzeige des Liniendiagramms.
- Nutzen Sie das Modul `psutil` zur Bestimmung der CPU-Auslastung.
- Speichern Sie das Programm unter dem Namen **`aufgabe_ls2_08.py`** ab.

Zusatzaufgaben:

- Zeigen Sie zusätzlich die Anzahl der laufenden Prozesse an.

Teil D: Nutzungsdaten graphisch zur Verfügung stellen (optional)



Aufgabe 8 - Einarbeitung in die bestehende Nutzungsdatensvisualisierung

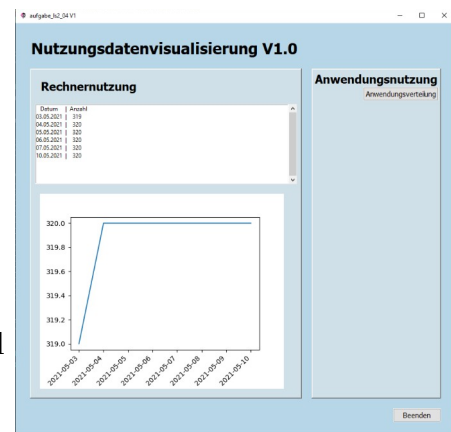
Machen Sie sich mit der vorhandenen Nutzungsdatensvisualisierung V1.0 vertraut.

- Erstellen Sie ein Kontextdiagramm und ein UML-Anwendungsfalldiagramm für die Nutzungsdatensvisualisierung.
- Beschreiben Sie die Struktur des Python-Programms mit einem UML-Klassendiagramm.
- Die Tkinter Elemente der Benutzeroberfläche bilden ausgehend vom Hauptfenster eine Hierarchie. Stellen Sie diese graphisch dar.
- Wenn der Konstruktor der Klasse NdsGui aufgerufen wird werden verschiedene Hilfsmethoden aufgerufen. Visualisieren Sie deren Aufrufhierarchie. Methoden- und Funktionsaufrufe, die in andere Dateien verweisen brauchen nicht weiter verfolgt werden.
- Welcher Schrifttyp wird für den Haupttitel genutzt? Wie groß ist die Schrift?
- Aus wie vielen Spalten und Zeilen ist das Raster aufgebaut, welches im Hauptrahmen aufgespannt wird? Gibt es Zellen in dem Raster, die nicht benutzt werden?
- Das Datum wird im europäischen Format tt.mm.jjjj angezeigt. Es soll in das amerikanische Format mm/tt/jjjj geändert werden. Welche Zeile ist zu ändern. Wie müsste die Zeile nun lauten (vollständig angeben)?
- Der Titel in der Kopfzeile muss geändert werden. In welcher Zeile würde Sie dies machen?

Aufgabe 9 - Nutzungsdatensvisualisierung V2.0

Führen Sie die folgenden Erweiterungsarbeiten an der Nutzungsdatensvisualisierung aus:

- Bisher wurden in dem System nur die Rechnernutzungsdaten ausgewertet und visualisiert. Erweitern Sie das Programm, so dass auch die **Anwendungsnutzungsdaten** ausgewertet und visualisiert werden.
- Nach Drücken des Knopfs **Anwendungsverteilung** soll ein zweites Fenster mit einer Tabellen erscheinen. In dieser Tabelle sind alle Anwendungen aufgelistet, die absolute Anzahl ihrer Aufrufe und ihr prozentualer Anteil an allen Anwendungsaufrufen.



Speichern Sie das Programm unter dem Namen **aufgabe_ls2_04** ab.