

Interne Datenstruktur

Als primäre Datenstruktur zur Speicherung von Daten haben wir uns für JSON-Dateien entschieden, da wir uns durch vorherige Nutzung mit denen auskannten und sie sich besonders in Kombination mit Javascript anbieten. Da die ständige Beschreibung und Auslesung dieser Dateien zu Ressourcenaufwändig wäre und sich auch sehr negativ auf die Laufzeit des Programmes auswirken würde, werden Dateien, solange sie verwendet werden in ein Objekt eingelesen und dann erst gegen Ende der Nutzung wieder in eine JSON-Datei eingespeichert. Der Nutzer erhält Ergebnisse in Form einer JSON-Datei. Demnach werden Personendaten, die aktuell verwendet werden, aus existierenden JSON-Dateien ausgelesen und in einem Objekt der Klasse Benutzer gespeichert. Wenn keine vorherigen Daten existieren werden sie, sobald sie verfügbar sind, in einem Benutzer-Objekt eingespeichert. Wenn die Verwendung vorbei ist, werden diese in eine JSON-Datei gespeichert. Außerdem werden Rezeptdaten aus JSON-Dateien aus der Datenbank in ein Rezept-Objekt eingelesen und dies dann in ein Array gespeichert. Die Antworten der API werden mit dem zum Benutzer gehörigen Bedarfs-Objekt verrechnet und in dem zum Benutzer gehörigen erreichtBedarf-Objekt gespeichert und ggfs. zu Ende des Nutzungszyklus in einer JSON Datei gespeichert.

Konkret sehen diese Klassen so aus:

Rezepte mit den folgenden Eigenschaften:

title: Ein *String*, welcher den Namen des Rezepts beinhaltet

zutaten: Ein *Array*, welches *Strings* mit den konkreten Mengenangaben und dem Namen der Zutat beinhaltet

zubereitung: Ein *String* mit der Zubereitung

dauer: *Number* mit der Zeitspanne die die Zubereitung benötigt

Benutzer mit den folgenden Eigenschaften:

Benutzerid: Eine *Number*, die fortlaufend generiert wird und den konkreten Benutzer repräsentiert

groesse: Eine *Number*, in welcher die Benutzereingabe zur Körpergröße gespeichert wird

gewicht: Eine *Number*, in welcher die Benutzereingabe zum Körpergewicht gespeichert wird

geschlecht: Ein *String*, in welchem die Benutzereingabe zum Geschlecht gespeichert wird. Zulässig hier sind Eingaben die entweder „m“ oder „w“ betragen.

activity: Ein *String*, in dem die Benutzereingabe zum Aktivitätslevel gespeichert wird. Zulässig hier sind Eingaben die entweder „gering“, „maessig“, „aktiv“ oder „sehr aktiv“ betragen.

alter: Eine *Number*, in welcher die Benutzereingabe zum Alter gespeichert wird

Legende:

Fettdruck: Klassenname

Unterstrichen: Name einer Klasseneigenschaft

Kursiv: Datentyp

bedarf: Ein *Objekt der Klasse Bedarfsdaten* die den täglichen Bedarf an Nährstoffen repräsentiert

erreichtTag: Ein *Objekt der Klasse erreichtBedarf*, welche die Menge des an dem Tag erreichten Bedarf an Nährstoffen repräsentiert

Bedarf mit folgenden Attributen:

kcal: Eine aus Benutzer.groesse, Benutzer.gewicht, Benutzer.alter, Benutzer.geschlecht, und Benutzer.activity errechnete *Number*, welche den täglichen Kalorienbedarf repräsentiert

fett: Eine aus kcal errechnete *Number*, welche den täglichen Fettbedarf repräsentiert

gesFett: Eine aus fett errechnete *Number*, welche die maximale Menge an gesättigten Fettsäuren repräsentiert

ungesFett: Eine aus fett und gesFett errechnete *Number*, welche den Tagesbedarf an ungesättigten Fettsäuren repräsentiert

protein: Eine aus gewicht und alter errechnete *Number*, welche die täglich empfohlene Menge an Proteinen repräsentiert

carbs: Eine aus kcal, fett und protein errechnete *Number*, welche die täglich empfohlene Menge an Kohlenhydraten repräsentiert

zucker: Eine aus carbs errechnete *Number*, die die täglich empfohlene Maximalzufuhr an Zucker repräsentiert

ErreichtBedarf mit folgenden Eigenschaften:

kcal: Eine aus dem gegessenen errechnete *Number*. Diese repräsentiert wie viel Prozent von Bedarf.kcal an dem Tag erreicht wurden

fett: Eine aus dem gegessenen errechnete *Number*. Diese repräsentiert wie viel Prozent von Bedarf.fett an dem Tag erreicht wurden

gesFett: Eine aus dem gegessenen errechnete *Number*. Diese repräsentiert wie viel Prozent von Bedarf.gesFett an dem Tag erreicht wurden

ungesFett: Eine aus dem gegessenen errechnete *Number*. Diese repräsentiert wie viel Prozent von Bedarf.ungesFett an dem Tag erreicht wurden

protein: Eine aus dem gegessenen errechnete *Number*. Diese repräsentiert wie viel Prozent von Bedarf.protein an dem Tag erreicht wurden

Legende:

Fettdruck: Klassenname

Unterstrichen: Name einer Klasseneigenschaft

Kursiv: Datentyp

carbs: Eine aus dem gegessenen errechnete *Number*. Diese repräsentiert wie viel Prozent von Bedarf.carbs an dem Tag erreicht wurden

zucker: Eine aus dem gegessenen errechnete *Number*. Diese repräsentiert wie viel Prozent von Bedarf.zucker an dem Tag erreicht wurden

Legende:

Fettdruck: Klassenname

Unterstrichen: Name einer Klasseeigenschaft

Kursiv: Datentyp