LU10a - Übersicht

Diese Learning Unit befasst sich mit Python-Modulen und Python-Paketen, zwei Mechanismen, die die modulare Programmierung erleichtern.

Bei der modularen Programmierung wird eine große, unhandliche Programmieraufgabe in separate, kleinere, besser zu verwaltende Teilaufgaben oder Module aufgeteilt. Die einzelnen Module können dann wie Bausteine zu einer größeren Anwendung zusammengefügt werden.

Die Modularisierung von Code in einer großen Anwendung hat mehrere Vorteile:

- **Vereinfachung:** Anstatt sich auf das gesamte Problem zu konzentrieren, konzentriert sich ein Modul normalerweise auf einen relativ kleinen Teil des Problems. Wenn Sie an einem einzigen Modul arbeiten, müssen Sie sich mit einem kleineren Problembereich auseinandersetzen. Das macht die Entwicklung einfacher und weniger fehleranfällig.
- ** Wartbarkeit:** Module sind in der Regel so konzipiert, dass sie logische Grenzen zwischen verschiedenen Problemdomänen erzwingen. Wenn Module so geschrieben werden, dass die gegenseitige Abhängigkeit minimiert wird, sinkt die Wahrscheinlichkeit, dass Änderungen an einem einzelnen Modul Auswirkungen auf andere Teile des Programms haben. (Unter Umständen können Sie sogar Änderungen an einem Modul vornehmen, ohne die Anwendung außerhalb dieses Moduls zu kennen.) Dies macht es für ein Team aus vielen Programmierern praktikabler, gemeinsam an einer großen Anwendung zu arbeiten.
- **Wiederverwendbarkeit:** Die in einem einzelnen Modul definierte Funktionalität kann leicht (über eine entsprechend definierte Schnittstelle) von anderen Teilen der Anwendung wiederverwendet werden. Dadurch entfällt die Notwendigkeit, Code zu duplizieren.
- **Skalierung:** Module definieren in der Regel einen separaten Namensraum, wodurch Kollisionen zwischen Bezeichnern in verschiedenen Bereichen eines Programms vermieden werden.

Inhalt von RealPython

From:
https://wiki.bzz.ch/ - **BZZ - Modulwiki**

Permanent link:
https://wiki.bzz.ch/de/modul/m319/learningunits/lu10/uebersicht

Last update: **2025/06/23 07:45**