

Einkaufsplaner

Prüfung 1


Fach	CS1
Klasse	20I/21NI
Lehrperson	Sagi Nedunkanal
Semester	Herbst 2021
Prüfungsbeginn	25.11.2021
Abgabetermin	04.12.2020, 23:59 Uhr

1. Ausgangslage

Es soll ein konsolenbasierter Einkaufsplaner entwickelt werden, womit man basierend auf mehreren Gerichten eine Einkaufsliste erstellen lassen kann.

Die Aufgabenstellung ist nicht in allen Details ausspezifiziert. Bei Unklarheiten wird erwartet, dass beim Dozenten nachgefragt wird.

Symbolbild



Bolognese-Sauce
Was schmeckt besser zu Spaghetti als eine Bolognese? Die würzige Tomatensauce mit gehacktem Rindfleisch und Gemüse ist der Saucen-Klassiker schlechthin.
★★★★★


⌚ Vor- und zubereiten: ca. 55 Min.
[Jetzt kochen](#)
[Auf die Favoriten-Liste](#)
[Rezept herunterladen](#)

Zutaten
Anzahl Personen: 4

- 1 Zwiebel
- 2 Knoblauchzehen
- 50 g Rübli
- 50 g Sellerie
- 2 EL Olivenöl
- 200 g Hackfleisch (Rind)
- 3 EL Tomatenpüree
- 1 dl Rotwein
- 600 g Tomaten
- 1 Zweiglein Rosmarin
- 1 TL Zucker
- 1 TL italienische Kräutermischung
- 1,5 TL Salz
- wenig Pfeffer

Und so wirds gemacht:
Zwiebel und Knoblauch schälen, fein hacken. Rübli und Sellerie schälen, in Würfel schneiden.
1 EL Öl in einer beschichteten Bratpfanne warm werden lassen. Zwiebel und Knoblauch andampfen. Rübli und Sellerie kurz mitdampfen. Herausnehmen, beiseite stellen.
Restliches Öl in derselben Pfanne heiß werden lassen. Fleisch ca. 5 Min. anbraten, Tomatenpüree begeben, kurz mitbraten. Hitze reduzieren, Wein dazugießen und vollständig einkochen.
Tomaten in Würfel schneiden, mit Rosmarin, Zucker, Kräutermischung und dem beiseite gestellten Gemüse begeben, zugedeckt unter gelegentlichem Rühren bei kleiner Hitze ca. 20 Min. köcheln. Rosmarin entfernen, würzen.

Zusatzinformationen
- Schneller: Statt frischen Tomaten 1 Dose gehackte Tomaten (ca. 800 g), abgetropft, verwenden.
- Rinds-Hackfleisch oder gemischtes Hackfleisch (Schwein und Rind) verwenden.



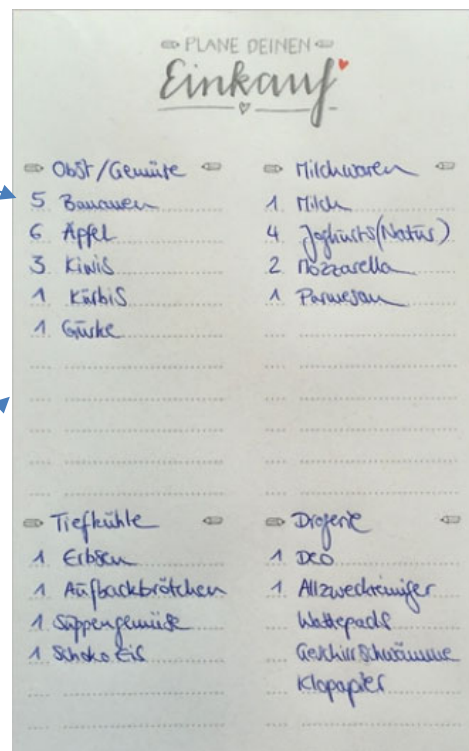
Schokoladecreme
Ein Klassiker und beliebt bei Jung und Alt: Eine Schokoladecreme kommt immer gut an und passt zu praktisch jedem Gericht.
★★★★★

⌚ Vor- und zubereiten: ca. 25 Min.
❄️ Kühl stellen: ca. 2 Std.
[Jetzt kochen](#)
[Auf die Favoriten-Liste](#)
[Rezept herunterladen](#)

Zutaten
Anzahl Personen: 4

- 1 Vanillestängel
- 5 dl Milch
- 1 EL Maizena
- 2 frische Eier
- 3 EL Zucker
- 100 g dunkle Schokolade

Und so wirds gemacht:
Vanillestängel längs aufschneiden, Samen auskratzen. Milch, Vanillesamen, Maizena, Eier und Zucker mit dem Schwingbesen verrühren, unter ständigem Rühren bei mittlerer Hitze bis vors Kochen bringen. Sobald die Masse bindet, die Pfanne sofort von der Platte nehmen.
Schokolade fein hacken, unter die Creme rühren, schmelzen. Creme durch ein Sieb in eine Schüssel gießen, auskühlen. Zugedeckt ca. 2 Std. kühl stellen.



Quelle: https://www.eine-der-guten.de/einkaufen/product_info.php?info=p1818_einkaufszettel-block---einkaufslisten-notizblock--50-blatt-design-abreissblock--weiss-beige--klimaneutral-produziert--feuerkueche--buero-familie-und-wg.html (24.11.2021)

2. Anforderungen

2.1. Menüführung

Schreibe eine Konsolenapplikation. Die Applikation soll über ein Menü folgende Funktionen anbieten:

1. Gericht erfassen
2. Gerichte anzeigen
3. Gericht mutieren
4. Gericht löschen
5. Einkaufsliste anzeigen
6. Programm beenden

Erstelle eine Bedienoberfläche, die optisch ansprechend und intuitiv zu bedienen ist.

2.2. Gericht erfassen

In diesem Teil der Applikation kann man Gerichte erfassen. Ein Gericht besteht aus einem Namen und einer Liste von Zutaten.

Zutaten bestehen aus einem Namen, einer Menge und einer Einheit (Enum: Liter, Esslöffel, Kilogramm, ...).

Beispiel: Siehe unten Kapitel 2.3 «Gerichte anzeigen».

2.3. Gerichte anzeigen

Es soll möglich sein, eine Liste der Gerichte anzuzeigen.

Beispiel:

Liste der Gerichte

```
Spaghetti bolognese [Gerichtname]
    [Zutatennamen]      [Menge] [Einheit]
- Gehacktes Rindfleisch:    175 Gramm
- Zwiebel:                  1 Stück
- Knoblauchzehe:           0.5 Stück
- Spaghetti:               250 Gramm
- Zutat X:                  5 Esslöffel
```

```
Schokoladecreme
- Vanillestängel:          0.5 Stück
- Milch:                   2.5 Deziliter
- Maizena:                 0.5 Esslöffel
- Eier:                    1 Stück
- Schokolade:              50 Gramm
- Zutat X:                  7 Esslöffel
```

2.4. Gericht mutieren

Der Name eines Gerichts kann verändert werden.

Vereinfachung: Einmal erfasst, sind Zutaten unveränderlich.

2.5. Gericht löschen

Ein bestehendes Gericht kann gelöscht werden.

2.6. Farbliche Unterscheidung

Die Menüführung soll sich farblich von den anderen Funktionen (Ein- und Ausgabe) unterscheiden.

2.7. Einkaufsliste anzeigen

Ziel ist es, dass der Benutzer eine konsolidierte Einkaufsliste ohne Mehrfachnennungen hat. Für das Testen soll zuerst die komplette Liste inkl. Mehrfachnennungen ausgegeben werden. Danach soll eine von Duplikaten befreite konsolidierte Einkaufsliste ausgegeben werden. Das ist die eigentliche Einkaufsliste, mit der man einkaufen gehen will.

Beispiel:

Einkausliste inklusive Mehrfachnennungen

- Tomatensauce: 1 L
- Teigwaren: 1 kg
- Tomatensauce: 2 L

Einkausliste ohne Mehrfachnennungen

- Tomatensauce: 3 L
- Teigwaren: 1 kg

Die Mehrfachnennung «Tomatensauce» wurde zusammengefasst. Sie hat den gleichen Namen und die gleiche Einheit.

2.8. Applikation beenden

Die Applikation kann via Menübefehl beendet werden.

2.9. Listen von Objekten

Bisher haben wir im Unterricht nur Arrays kennengelernt, um Listen abzubilden. Daher kann man zu Vereinfachung Arrays fixer längen definieren. Beispielsweise, dass man nur maximal 5 Gerichte mit je maximal 4 Zutaten erfassen kann und dass die Einkaufsliste maximal 20 Einträge haben kann.

Die Fortgeschrittenen, die die Klasse List oder List<T> kennen, können auch die benutzen. Dann braucht es keine willkürlichen Längenbegrenzungen.

3. Code

Die bei den Hausaufgabenbesprechungen und Codereviews besprochenen Regeln sollen eingehalten werden.

4. Softwaredesign

Es sollen geeignete Klassen entworfen werden, um die nötigen Daten zu verwalten.

5. Qualitätssicherung

5.1. Testdaten

Beim Applikationsstart soll bereits eine Ausgangslage mit mindestens 3 Gerichten und je mindestens 3 Zutaten erfasst sein. Bei Gericht 1 und 2 soll mindestens eine Zutat vom gleichen Typ sein und bei Gericht 2 und 3 soll auch mindestens eine Zutat vom gleichen Typ sein, so dass sie für die Einkaufsliste zusammengefasst werden können.

Vereinfachung: Die Testdaten dürfen hartkodiert aus einer Fabrikmethode kommen. Die Testdaten können fiktiv sein, so dass sie den obigen Anforderungen genügen.

5.2. Validierung

Es sollen geeignete Validierungen durchgeführt werden, so dass der Benutzer immer weiss, wenn etwas falsch ist und so, dass die Applikation nicht abstürzt.

Die Validierungen sollen mit den im Unterricht kennengelernten Methoden umgesetzt werden, z.B. Casting, TryParse oder try-catch.

6. Dokumentation

Es soll eine minimale Dokumentation erstellt werden:

- Lösungsskizze (kann von Hand gezeichnet, fotografiert und als JPEG abgelegt werden; kein im Nachhinein generiertes Klassendiagramm)
- Screenshots aller Fenster/Masken
- Mindestens 3 Commits

7. Bewertungskriterien

1. Umsetzung der funktionalen Anforderungen (1/4)
 2. Bewertung des GUIs (1/4)
 3. Bewertung des Codes (1/4)
 4. Testen der Applikation (1/4)
- Erlaubt: Internet als Quelle, Beispiele aus dem Unterricht usw.
 - Nicht erlaubt: Kopieren grösserer Codemengen, Mitarbeit Dritter usw.
 - Selbstkontrolle: «Kann ich jede Zeile Code erklären?»
 - Missachtung der Prüfungsregeln führen zu einer Note 1.
 - Verspätete Abgabe führt zu Notenabzug:
 - Bis 6h Verspätung → 0.5 Note Abzug
 - Bis 12h Verspätung → 1.0 Note Abzug
 - >12h Verspätung → Note 1.0

8. Abgabe

Die Applikation wird via GitHub abgegeben:

1. Den Quellcode ablegen unter
CS1\02 Prüfungen\Prüfung 1\Quellcode\<Vorname Nachname>
CS1 Prüfung 1.sln
2. Das Kompilat als ZIP ablegen unter
CS1\02 Prüfungen\Prüfung 1\Kompilat\Einkaufsplaner.zip
3. Die Dokumentation ablegen unter
CS1\02 Prüfungen\Prüfung 1\Dokumentation

4. Alles hochladen

Verwende die Versionsverwaltung, um regelmässig Sicherungen des aktuellen Standes zu erstellen (mindestens drei Commits) und hochzuladen (Push).