LU02.L08 - Kochbuch

```
# Implementierung der Musterlösung für das Rezeptbuch

import json

def adjust_recipe(original_recipe, num_people):
    """
    Adjusts the recipe based on the number of people.

    Parameters:
        original_recipe (dict): The original recipe.
        num_people (int): The number of people to adjust the recipe for.

    Returns:
        dict: The adjusted recipe.
    """
    adjusted_ingredients = {}
    original_servings = original_recipe['servings']
    factor = num_people / original_servings

    for ingredient, amount in original_recipe['ingredients'].items():
        adjusted_ingredients[ingredient] = amount * factor

    adjusted_recipe = {
        'title': original_recipe['title'],
        'ingredients': adjusted_ingredients,
        'servings': num_people
    }
    return adjusted_recipe

def load_recipe(json_string):
    """
    Loads a recipe from a JSON string.

    Parameters:
        json_string (str): The JSON string containing the recipe.

    Returns:
        dict: The recipe as a Python dictionary.
    """
    return json.loads(json_string)


# Beispiel für die Anwendung der Funktionen
if __name__ == '__main__':
    # Example JSON string for a recipe
    recipe_json = '{"title": "Spaghetti Bolognese", "ingredients":
{"Spaghetti": 400, "Tomato Sauce": 300, "Minced Meat": 500}, "servings": 4}'
```

```
    # Convert JSON string to Python dictionary
    original_recipe = load_recipe(recipe_json)
    # Adjust the recipe for 2 people
    adjusted_recipe = adjust_recipe(original_recipe, 8)
    print(f"Original Recipe: {original_recipe}")
    print(f"Adjusted Recipe: {adjusted_recipe}")
```

© Kevin Maurizi

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

Permanent link:
https://wiki.bzz.ch/modul/m323/learningunits/lu02/loesungen/kochbuch

Last update: **2024/03/28 14:07**