# 3. Cookbook



*Alex is a passionate home chef with a diverse collection of recipes gathered from all corners of the globe. As he aspires to share his culinary creations with the world, his editor has tasked him with the challenging job of organizing these recipes into a coherent cookbook.*

Write a function called **"cookbook"** to assist Alex in **arranging his recipes systematically**. The function will receive a **variable number of arguments**, passed as **tuples containing three elements**: the **first** element is the **recipe's name**, the **second** is the **cuisine**, and the **third** is a **list of ingredients** e.g., **("Recipe Name", "Cuisine", ["Ingredient 1", "Ingredient 2"])**.

The objective is to **sort the recipes by their cuisine**, arranging Alex's collection **based on the number of recipes** in each cuisine in **descending order**. In cases where two or more cuisines have the **same number of recipes**, they should be returned in **ascending order** (alphabetically) **by cuisine**.

Within each **cuisine group**, the recipes should be sorted in **ascending order** (alphabetically) **by the recipe's name**.

To aid Alex in quickly assessing the number of recipes in each cuisine group, the function should print the **count of recipes next to each cuisine**. Furthermore, for each recipe within a cuisine group, **display the necessary ingredients**.

**In the end, return** the output as described below.

***Note: Submit only the function in the judge system***

### Input

* There will be **no input from the console**, just parameters passed to your function

### Output

* The **output** should look like this (before the star, there are two empty spaces.)**:**

**"{cuisine\_1} cuisine contains {number\_of\_recipes\_in\_the\_cuisine\_group} recipes:**

**\* {recipe\_name\_1} -> Ingredients: {ingredient\_1}, {ingredient\_2}, ...**

**\* {recipe\_name\_2} -> Ingredients: {ingredient\_1}, {ingredient\_2}, ...**

**...**

**\* {recipe\_name\_n} -> Ingredients: {ingredient\_1}, {ingredient\_2}, ...**

**{cuisine\_2} cuisine contains {number\_of\_recipes\_in\_the\_cuisine\_group} recipes:**

**\* {recipe\_name\_1} -> Ingredients: {ingredient\_1}, {ingredient\_2}, ...**

**...**

**\* {recipe\_name\_n} -> Ingredients: {ingredient\_1}, {ingredient\_2}, ... {cuisine\_n} cuisine contains {number\_of\_recipes\_in\_the\_cuisine\_group} recipes:**

**\* {recipe\_name\_1} -> Ingredients: {ingredient\_1}, {ingredient\_2}, ..."**

### Constraints

* Each **tuple** provided will always contain a **unique recipe** and its **associated cuisine**.
* Alex will **never** receive the **same recipe** twice or more in the given input.
* The **list of ingredients** for each recipe will never be **empty**.

### Examples

|  |  |
| --- | --- |
| **Test Code** | **Output** |
| **print(cookbook(**  **("Spaghetti Bolognese", "Italian", ["spaghetti", "tomato sauce", "ground beef"]),**  **("Margherita Pizza", "Italian", ["pizza dough", "tomato sauce", "mozzarella"]),**  **("Tiramisu", "Italian", ["ladyfingers", "mascarpone", "coffee"]),**  **("Croissant", "French", ["flour", "butter", "yeast"]),**  **("Ratatouille", "French", ["eggplant", "zucchini", "tomatoes"])**  **))** | **Italian cuisine contains 3 recipes:**  **\* Margherita Pizza -> Ingredients: pizza dough, tomato sauce, mozzarella**  **\* Spaghetti Bolognese -> Ingredients: spaghetti, tomato sauce, ground beef**  **\* Tiramisu -> Ingredients: ladyfingers, mascarpone, coffee**  **French cuisine contains 2 recipes:**  **\* Croissant -> Ingredients: flour, butter, yeast**  **\* Ratatouille -> Ingredients: eggplant, zucchini, tomatoes** |
| print(cookbook(  ("Pad Thai", "Thai", ["rice noodles", "shrimp", "peanuts", "bean sprouts", "tamarind sauce"])  )) | Thai cuisine contains 1 recipes:  \* Pad Thai -> Ingredients: rice noodles, shrimp, peanuts, bean sprouts, tamarind sauce |
| print(cookbook(  ("Spaghetti Bolognese", "Italian", ["spaghetti", "tomato sauce", "ground beef"]),  ("Margherita Pizza", "Italian", ["pizza dough", "tomato sauce", "mozzarella"]),  ("Tiramisu", "Italian", ["ladyfingers", "mascarpone", "coffee"]),  ("Croissant", "French", ["flour", "butter", "yeast"]),  ("Ratatouille", "French", ["eggplant", "zucchini", "tomatoes"]),  ("Sushi Rolls", "Japanese", ["rice", "nori", "fish", "vegetables"]),  ("Miso Soup", "Japanese", ["tofu", "seaweed", "green onions"]),  ("Guacamole", "Mexican", ["avocado", "tomato", "onion", "lime"])  )) | Italian cuisine contains 3 recipes:  \* Margherita Pizza -> Ingredients: pizza dough, tomato sauce, mozzarella  \* Spaghetti Bolognese -> Ingredients: spaghetti, tomato sauce, ground beef  \* Tiramisu -> Ingredients: ladyfingers, mascarpone, coffee  French cuisine contains 2 recipes:  \* Croissant -> Ingredients: flour, butter, yeast  \* Ratatouille -> Ingredients: eggplant, zucchini, tomatoes  Japanese cuisine contains 2 recipes:  \* Miso Soup -> Ingredients: tofu, seaweed, green onions  \* Sushi Rolls -> Ingredients: rice, nori, fish, vegetables  Mexican cuisine contains 1 recipes:  \* Guacamole -> Ingredients: avocado, tomato, onion, lime |