# 3. Hourly Forecast

*Patricia wants to go on vacation for the weekend and wants to know where the weather will be the best, so she can see the most sights. Patricia is busy at work and doesn't have time to think about the perfect place for her vacation, so she wants your help.*

Write a function called **forecast** that **stores information** about the **weather**, and **returns sorted information for all locations**. The function will receive a **different number of arguments**. The arguments will be passed as **tuples with two elements** - the **first** one is the **location**, and **the second one** is the **weather**:

* **Location name**: string
  + any string
* **Weather**:string
  + "Sunny"
  + "Rainy"
  + "Cloudy"

First, **sort all** **locations by weather**. **First** are positioned the locations with **sunny weather**, next are the locations with **cloudy weather**, and **last** are the locations with **rainy weather**. For each sequence of locations (e.g. all sunny locations), sort them by their name in **ascending order** (alphabetically).

**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**:**

**"{first\_sorted\_location} - {weather}"**

**"{second\_sorted\_location} - {weather}"**

**…**

**"{last\_sorted\_location} - {weather}"**

### Constraints

* Each **tuple** given will always contain the **location** with its **weather**.
* You will **never** receive the **same location** twice or more times.

### Examples

|  |  |
| --- | --- |
| **Test Code** | **Output** |
| print(forecast(  ("Sofia", "Sunny"),  ("London", "Cloudy"),  ("New York", "Sunny"))) | New York - Sunny  Sofia - Sunny  London - Cloudy |
| print(forecast(  ("Beijing", "Sunny"),  ("Hong Kong", "Rainy"),  ("Tokyo", "Sunny"),  ("Sofia", "Cloudy"),  ("Peru", "Sunny"),  ("Florence", "Cloudy"),  ("Bourgas", "Sunny"))) | Beijing - Sunny  Bourgas - Sunny  Peru - Sunny  Tokyo - Sunny  Florence - Cloudy  Sofia - Cloudy  Hong Kong - Rainy |
| print(forecast(  ("Tokyo", "Rainy"),  ("Sofia", "Rainy"))) | Sofia - Rainy  Tokyo - Rainy |