# Overview

There are three types of data that need to be imported/exported at the same time: SKUs, Ingredients, and Product Lines. Each of these can be naturally and nicely fit into a single table, so it remains to represent the relations (SKU-Ingredient) and (SKU-Product Line). Two additional tables are used to express these relations.

CSV files do a good job representing table data, and are extremely easy to parse. In addition, CSVs are among the most space-efficient formats, so long as the tables they are representing are themselves the minimal. As discussed above, **four** CSV (comma-separated values) files are specified, and their headers are defined below:

| **Filename Prefix** | **Header** |
|---|---|
| skus | **`SKU#`**`,Name,Case UPC,Unit UPC,Unit size,Count per case,Product Line Name,Comment` |
| ingredients | **`Ingr#`**`,Name,Vendor Info,Size,Cost,Comment` |
| product_lines | **`Name`** |
| formulas | **`SKU#`**`,Ingr#,Quantity` |

# File format

1. The names of the files must be prefixed by an identifying string, as defined above. The filenames must end in ".csv". More precisely, the filenames must constitute matches for the following regular expressions conforming to PCRE standard:
    - SKUs: `skus(\S)*\.csv`
    - Ingredients: `ingredients(\S)*\.csv`
    - Product Lines: `product_lines(\S)*\.csv`
    - Formulas: `formulas(\S)*\.csv`
2. The CSV files must be properly delimited and escaped. RFC4180 specifies the format of a CSV file. Note specifically:
    - Spaces (and other space-like characters) must not be ignored if they are part of a field.
    - Null values, if they are used, must be denoted by an empty string (i.e., two consecutive commas, or a comma followed by a `CRLF` character).
3. A valid header, as defined above, must be included as the first line in each file. The header is case-sensitive.
4. The columns must conform to the data types specified on the Evolution requirements.
5. Unless otherwise specified in the Evolution requirements, all VARCHAR-typed columns must be no more than 1000 characters.
6. Unless otherwise specified in the Evolution requirements, all Integer-typed columns must be within range of $[-2^{31}, 2^{31})$ (i.e., a 32-bit signed integer).
7. Unless otherwise specified in the Evolution requirements, all floating-point numbers must be in decimal, and have a leading 0 (before the decimal place) if the value is less than 1.0.
8. Each file must contain no more than 1,000,000 lines. The **records** may be in no particular order.

# Upload Specifications

1. Only one file may be submitted in one import **session**. This file may be of any type.
2. All submitted **CSV file(s)** may be assumed to have MIME type `text/csv`, as stipulated in RFC4180 §3.
3. The following checks must be performed on each file individually. If any of the check fails, the file must be rejected as a whole, and an appropriate error message must be displayed for the administrator.
    - File validity: the file must be standards-compliant, and contain the required headers in the order specified.
    - **Duplicate records**: the file must not contain **duplicate records** in itself. The file may contain **collision(s)** to **records** existing in the system. Refer to the Collision Resolution algorithm in the next section when a **collision** is detected.
    - Data validity: for example, the UPC numbers must conform to the UPC-A standard, as specified in the **Evolution requirements**.
    - Empty attributes: all required attribute(s) must be supplied, except when both of the following are true:
        - It is marked as **autogeneratable** in the **Evolution requirements**;
        - The record whose required attribute is empty contains no **collision**.
4. **Referential integrity** must be preserved in the file.
   Note that the following referential relations exist among the files / corresponding entries in the system:
    - In `skus.csv`,
        - `Product Line Name` must be existing in the system.
    - In `formulas.csv`:
        - `SKU#` must be existing in the system.
        - `Ingr#` must be existing in the system.

Example skus.csv:

SKU#,Name,Case UPC,Unit UPC,Unit size,Count per case,Product Line Name,Comment
12345,Tomato soup,005102218476,005102218476,28oz,24,Canned goods,Good food
12346,Chicken soup,005102218377,005102218377,28oz,24,Canned goods,Good food
13346,Broccoli cheddar soup,005103218376,005103218376,28oz,24,Canned goods,Good food

Example ingredients.csv:

Ingr#,Name,Vendor Info,Size,Cost,Comment
888893,Flour,Utah Vendors,60lb,100,
888894,Not Flour,Utah Vendors,60lb,100,

Example product_lines.csv:

Name
Canned goods

Example forumas.csv:

SKU#,Ingr#,Quantity
12345,888894,0.3