

# 營養標示 Helper

# 營養標示 Helper

Embedded system with  
camera extension

Nutrition image maybe  
in multi-language?



營養標示			
每份重量32公克 本包裝含43份			
		每份	每100公克
熱量	Energy	163 大卡	511 大卡
蛋白質	Protein	7.7 公克	24.0 公克
脂肪	Fat	9.2 公克	28.8 公克
飽和脂肪	Saturated Fat	6.0 公克	18.8 公克
反式脂肪	Trans Fat	0.4 公克	1.4 公克
碳水化合物	Carbohydrate	12.5 公克	39.0 公克
糖	Sugar	12.5 公克	39.0 公克
鈉	Sodium	90 毫克	280 毫克
維生素A	Vitamin A	67 微克RE	210 微克RE
維生素B <sub>1</sub>	Vitamin B <sub>1</sub>	0.06 毫克	0.2 毫克
維生素B <sub>2</sub>	Vitamin B <sub>2</sub>	0.4 毫克	1.4 毫克
泛酸	Pantothenic Acid	0.8 毫克	2.5 毫克
葉酸	Folate	11 微克	35 微克
生物素	Biotin	7 微克	22 微克
鈣	Calcium	304 毫克	950 毫克
鐵	Iron	0.06 毫克	0.2 毫克
鋅	Zinc	1.0 毫克	3.0 毫克
鎂	Magnesium	26 毫克	80 毫克
鉀	Potassium	384 毫克	1200 毫克
磷	Phosphorus	230 毫克	720 毫克

# 營養標示 Helper

Nutrition image maybe  
in multi-language?



營養標示

每份量32公克  
本包裝含43份

		每份	每 100 公克
熱量	Energy	163 大卡	511 大卡
蛋白質	Protein	7.7 公克	24.0 公克
脂肪	Fat	9.2 公克	28.8 公克
飽和脂肪	Saturated Fat	6.0 公克	18.8 公克
反式脂肪	Trans Fat	0.4 公克	1.4 公克
碳水化合物	Carbohydrate	12.5 公克	39.0 公克
糖	Sugar	12.5 公克	39.0 公克
鈉	Sodium	90 毫克	280 毫克
維生素A	Vitamin A	67 微克RE	210 微克RE
維生素B1	Vitamin B1	0.06 毫克	0.2 毫克
維生素B2	Vitamin B2	0.4 毫克	1.4 毫克
泛酸	Pantothenic Acid	0.8 毫克	2.5 毫克
葉酸	Folate	11 微克	35 微克
生物素	Biotin	7 微克	22 微克
鈣	Calcium	304 毫克	950 毫克
鐵	Iron	0.06 毫克	0.2 毫克
鋅	Zinc	1.0 毫克	3.0 毫克
鎂	Magnesium	26 毫克	80 毫克
鉀	Potassium	384 毫克	1200 毫克
磷	Phosphorus	230 毫克	720 毫克

Package used for recognize the  
text in the image

User specify or do some ML  
magic to recognize which  
languages used in the image.

Optical Character Recognition

OCR

Text and string.



<https://hackmd.io/@coherent17/H1AM2ce1o>

# 營養標示 Helper

Put the result into the pre-build hash table.  
(using C/C++ for fast and portable reason)

## Hash table structure

```
1  #define TABLE_SIZE 100
2
3  typedef struct{
4      char *nutrition_name;    //ex: Na (鈉)
5      char *unit               //ex: mg (毫克)
6      int max_intake;          //ex: 2000 (mg)
7      int already_intake;      //ex: 1000 (mg)
8  }Nutrition_Component;
9
10 Nutrition_Component *hash_table[TABLE_SIZE];
```

Found in hash table: put it in.

If not found in hash table, insert it. And it will be found next time!

Text and string.

Since the result might be wrong some time, we need some mechanism to fix the text.

Do some string or text preprocessing(like auto complete) and show the result to the user.  
Or maybe let the user to fix the result text.

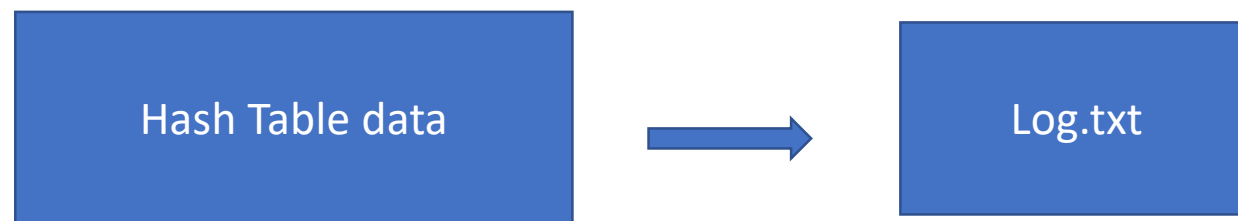
## Hash function

```
int hash(char *nutrition_name);
```

Or name-based uuid as the key of hash function?

```
1  #random type uuid
2  uuidgen
3
4  #time type uuid
5  uuidgen -t
6
7  #name type uuid
8  uuidgen --name = "some string" -s --namespace = uuid
```

# 營養標示 Helper



Since we can't not save our data after the program was terminated, we should output the data stored in the hash table as the log. And read the log next time if needed.

# 營養標示 Helper



**command line tool and library**  
for transferring data with URLs

What if user give us the nutrition that we are not familiar with? Ex: Rg

## Ask



## Fast and Quick web crawler? :libcurl

化學元素週期表

The diagram illustrates the periodic table with the following elements highlighted:

- Hydrogen (H):** Atomic number 1, symbol H, name 氫.
- Helium (He):** Atomic number 2, symbol He, name 氦.
- Lithium (Li):** Atomic number 3, symbol Li, name 鋰.
- Beryllium (Be):** Atomic number 4, symbol Be, name 鈹.
- Sodium (Na):** Atomic number 11, symbol Na, name 鈉.
- Magnesium (Mg):** Atomic number 12, symbol Mg, name 鎂.
- Potassium (K):** Atomic number 19, symbol K, name 鉀.
- Calcium (Ca):** Atomic number 20, symbol Ca, name 鈣.
- Rubidium (Rb):** Atomic number 37, symbol Rb, name 鉍.
- Sr** (Strontium): Atomic number 38, symbol Sr, name 銻.
- Y** (Yttrium): Atomic number 39, symbol Y, name 鈹.
- Zr** (Zirconium): Atomic number 40, symbol Zr, name 鈷.
- Nb** (Niobium): Atomic number 41, symbol Nb, name 鈷.
- Mo** (Molybdenum): Atomic number 42, symbol Mo, name 鉬.
- Tc** (Technetium): Atomic number 43, symbol Tc, name 鈷.
- Ru** (Ruthenium): Atomic number 44, symbol Ru, name 鈷.
- Rh** (Rhodium): Atomic number 45, symbol Rh, name 鈷.
- Pd** (Palladium): Atomic number 46, symbol Pd, name 鈷.
- Ag** (Silver): Atomic number 47, symbol Ag, name 銀.
- Cd** (Cadmium): Atomic number 48, symbol Cd, name 鎘.
- In** (Indium): Atomic number 49, symbol In, name 銦.
- Sn** (Tin): Atomic number 50, symbol Sn, name 錫.
- Sb** (Antimony): Atomic number 51, symbol Sb, name 銻.
- Te** (Tellurium): Atomic number 52, symbol Te, name 碲.
- I** (Iodine): Atomic number 53, symbol I, name 碘.
- Xe** (Xenon): Atomic number 54, symbol Xe, name 氙.
- Cs** (Cesium): Atomic number 55, symbol Cs, name 銻.
- Ba** (Barium): Atomic number 56, symbol Ba, name 鋇.
- La** (Lanthanum): Atomic number 57, symbol La, name 釷.
- Hf** (Hafnium): Atomic number 72, symbol Hf, name 鈷.
- Ta** (Tantalum): Atomic number 73, symbol Ta, name 鈷.
- W** (Tungsten): Atomic number 74, symbol W, name 鈷.
- Re** (Rhenium): Atomic number 75, symbol Re, name 鈷.
- Os** (Osmium): Atomic number 76, symbol Os, name 鈷.
- Ir** (Iridium): Atomic number 77, symbol Ir, name 鈷.
- Pt** (Platinum): Atomic number 78, symbol Pt, name 鈷.
- Au** (Gold): Atomic number 79, symbol Au, name 金.
- Hg** (Mercury): Atomic number 80, symbol Hg, name 汞.
- Tl** (Thallium): Atomic number 81, symbol Tl, name 鉛.
- Pb** (Lead): Atomic number 82, symbol Pb, name 鉛.
- Bi** (Bismuth): Atomic number 83, symbol Bi, name 鉍.
- Po** (Polonium): Atomic number 84, symbol Po, name 鉍.
- At** (Astatine): Atomic number 85, symbol At, name 鉍.
- Rn** (Radon): Atomic number 86, symbol Rn, name 氡.
- Fr** (Francium): Atomic number 87, symbol Fr, name 銻.
- Ra** (Radium): Atomic number 88, symbol Ra, name 銻.
- Ac** (Actinium): Atomic number 89, symbol Ac, name 釷.
- Rf** (Rutherfordium): Atomic number 104, symbol Rf, name 鈷.
- Db** (Dubnium): Atomic number 105, symbol Db, name 鈷.
- Sg** (Seaborgium): Atomic number 106, symbol Sg, name 鈷.
- Bh** (Bohrium): Atomic number 107, symbol Bh, name 鈷.
- Hs** (Hassium): Atomic number 108, symbol Hs, name 鈷.
- Mt** (Meitnerium): Atomic number 109, symbol Mt, name 鈷.
- Ds** (Darmstadtium): Atomic number 110, symbol Ds, name 鈷.
- Rg** (Roentgenium): Atomic number 111, symbol Rg, name 鈷.
- Cn** (Copernicium): Atomic number 112, symbol Cn, name 鈷.
- Nh** (Nihonium): Atomic number 113, symbol Nh, name 鈷.
- Fl** (Flerovium): Atomic number 114, symbol Fl, name 鈷.
- Mc** (Moscovium): Atomic number 115, symbol Mc, name 鈷.
- Lv** (Livermorium): Atomic number 116, symbol Lv, name 鈷.
- Ts** (Tennessine): Atomic number 117, symbol Ts, name 鈷.
- Og** (Oganesson): Atomic number 118, symbol Og, name 鈷.

```

21 int main(){
22     CURL *curl = curl_easy_init();
23
24     if(!curl){
25         fprintf(stderr, "init failed\n");
26         return EXIT_FAILURE;
27     }
28
29     //set options
30     curl_easy_setopt(curl, CURLOPT_URL, "https://www.google.com/");
31     curl_easy_setopt(curl, CURLOPT_WRITEFUNCTION, got_data);
32
33     //perform out action
34     CURLcode result = curl_easy_perform(curl);
35     if(result!=CURLE_OK){
36         fprintf(stderr, "download problem: %s\n", curl_easy_strerror(result));
37     }
38
39
40     curl_easy_cleanup(curl);
41     return EXIT_SUCCESS;
42 }

```

# 營養標示 Helper

After doing some html string manipulation,  
we can further insert the data into our hash  
table, and that data no longer become  
strange next time!



```
hash_table_insert(Nutrition_Component new_nutrition);
```

# 營養標示 Helper

How to use or represent the data store in log.txt?

- Data Analyze with matplotlib, make some statistic chart.
- Give eating recommendation to user.

How to represent or where to data visualization?





Question?