UPC/EAN Product Code API

Implementation Guide

Revision history:

May 21, 2013	V 1.0	Initial publication
Aug 15, 2013	V 1.1	Added fields for GCP download, Thumbnail retrieval and categories
Sep 23, 2013	V 2.0	Major software release: Added export in JSON format, modified parameter names to be consistent across protocols, added structure to GCP and manufacturer sections in the XML, added requirement for language parameter to deletion request.
Oct 3, 2013	V 2.1	Added http status code documentation, Python code
Jan 8,2014	V 2.1a	Corrected at typo in the name of the downloadable code library, added return codes 1 – 7 to the data request transaction
Jun 21, 2014	V 2.2a	Clarified the handling of HTTP status codes for JSON users
June 25, 2015	V 2.3	Documented the addition of the formatted nutrition data
January 6, 2016	V 2.3a	Corrections to the PHP code snippet, courtesy of Mark Hahs
June 15, 2020	V 3.0	Implementation of the V3 endpoint that allows the retrieval of price data for an item and which also allows the user to request system 4 (private) UPC numbers in the output.
February 2, 2020	V3.1	Extension of the V3 endpoint to allow the full-text search of the UPC database by description and/or brand. The implementation adds the "search_text" and "limit" param.

CONTENTS

OVERVIEW

The Digit-Eyes UPC / EAN database contains over 250,000,000 items identified by UPC, GTIN, EAN, APN and JPN codes.   It grows continually as a result of search activity by our customers. Listings are provided in 12 languages.

The API allows appropriately licensed, identified and authenticated processes to:

- Retrieve, update and delete individual records with our JSON or SOAP XML interface;
- Retrieve record groups by brand or product name matches with JSON or XML ;
- Retrieve records for key groups where the keys are specified using data in CSV (excel) format;
- Upload record groups using data in CSV (excel) format;

There are three versions of the JSON/XML API:

- V1 did not produce JSON. The V2 endpoint provides XML output that is backward-compatible to the V1 specifications
- The V2 endpoint returns a single record per UPC number in either XML or JSON;
- The V3 endpoint produces either XML or JSON and in addition to the V2 data may, additionally:
    - If requested, provide price data
    - If requested, provide system 4 numbers (and, hence, produce multiple records per UPC number).
    - If requested, search product by full text match to product description and brand (and hence produce multiple records for search request)

Data available about a code:

- always includes the product title and UPC;
- may include the address of the owner/issuer of the UPC/EAN code plus at least one reference to a product page where information can be found and the location of an image; and
- often includes ancillary information such as the manufacturer or vendor's romance description of the product, product size, product brand, ingredients or nutritional information;
- If requested, may include category information for each item;
- if requested through the V3 endpoint, it may optionally include pricing information from various sources and data about system 4 (private) UPC numbers.   For more about private UPC numbers, see Appendix A.

- if requested through the V3 endpoint by search, may include multiple UPC numbers in response to a query on brand and/or product title.

Revenue from licensing the digit-eyes database is used to provide low-cost digital product identification services for people without vision and who use our digit-eyes app.

JSON SINGLE-TRANSACTION API FUNCTIONS

UPC/EAN DATA REQUEST

The request for information in the JSON format is submitted as a GET operation to the endpoint.

In V2 format:

```
http://digit-eyes.com/gtin/v2_0/?upc_code=x&app_key=x&signature=x&language=x&field_names=x
```

In V3 format:

```
http://digit-eyes.com/gtin/v3_0/?upc_code=x&app_key=x&signature=x&language=x&field_names=x
```

Price data (an additional charge item) is now available. Please note that if the data is requested, the charge is applied, even if the system is unable to obtain the price data. To get price data, the value ",prices" should be added to the fieldname list:

```
http://digit-eyes.com/gtin/v3_0/?upc_code=x&app_key=x&signature=x&language=x&field_names=x,
price
```

Text search by category/product name is now available. The value "search_text=" should be added to the request instead of the UPC and the signature calculated using the entire value for which search is requested (including spaces):

```
http://digit-eyes.com/gtin/v3_0/?search_text=x&app_key=x&signature=x&language=x&field_names=x,
price
```

The field "limit" can be used to limit the number of results produced in response to the text search. If not supplied, the results are limited to 20.

1. **either:**
   - o **upc_code:** the UPC, EAN, GTIN, APN or JPN code or:
   - o **search_text:** (V3 only) the text value to search to return a UPC code list.
2. **app_key:** is the application identification key of the party making the request;
3. **signature:** created using the upc_code or search text and auth_key. Code examples are in [Appendix D](#);
4. **language**: the ISO 639-1 2- character language code for the preferred language of response ("en," "es," etc.)
5. **system4 (Optional):** value 0 or 1; if 1 and a private/system 4 UPC number is provided. more than a single UPC record may be returned if the information is requested about a System 4 number. In V2, System 4 numbers are ignored.
6. **limit:** a numeric value indicating a limit to the number of results returned in the cases where multiple results may be returned (system 4 and full text search). If omitted, the maximum number of results returned will be 20.

Optional parameters:

7. **field_names:** comma-separated list of the fields requested from the database.

    Acceptable field requests for both V2 and V3 are:

    - o **description:** short product name;
    - o **uom:** unit of measure;
    - o **usage**: romance description / instructions (if available);
    - o **brand**: product labeling (if available);
    - o **language:** 2-character ISO code for the preferred language of listing;
    - o **website:** the authoritative website for the product (typically the seller or manufacturer website);
    - o **product_web_page:** address of the individual web page that describes the product, if available;
    - o **nutrition**: nutrition information in human-readable free text (if available);
    - o **formattedNutrition**: nutrition data in a structured format (US-format labels only at this time and only if the data is available)
    - o **ingredients:** ingredients of the product (if available);
    - o **manufacturer:** Manufacturer's name and address elements (if available);
    - o **gcp_name_address:** GCP owners name and address elements (if available). Please note that if you ask for GCP and the UPC information is not available, the system will return the GCP information and set http code 200. You should check the response_code to determine whether only GCP information was returned.
    - o **image:** the web address of the manufacturer's image of the product (if available)
    - o **thumbnail:** a small image of the item, as well as the dimensions
    - o **categories:** list of tags or descriptive categories associated with this item.

V3 supports the additional field requests (these field requests are accepted, but ignored by the V2 endpoint)

    - o **prices:** an array of prices, including the currency, vendor business name, vendor address, and date the price data was collected;

Demonstration: http://www.digit-eyes.com/gtin/
Signature calculator: http://www.digit-eyes.com/gtin/
Your keys: https://www.digit-eyes.com/cgi-bin/digiteyes.fcgi?action=myAccount

JSON Single-transaction API Functions

UPC/EAN DATA REQUEST RESPONSE

The http status code indicates the success or failure of the request (detail in appendix D). The JSON response for error transactions will include a return code and message that indicates the reason for the failure per Appendix C.

V2 RESULTS

The JSON response transaction for a successful UPC lookup to the V2 endpoint will have http status code 200 and a payload:

```
{
"brand" : "Grey Poupon",
 "categories": "Canned Dry & Packaged Foods, Condiment & Sauces, Condiments, Condiments & Salad
 Dressings, Condiments & Sandwich Toppings, Condiments & Sauces, Condiments Dressings, Condiments
 Mustard, Condiments Mustards, Condiments Pickles & Relishes,  Condiments Sauces & Spices,
 Condiments Vinegars & Oils, Cooking & Baking,  Cooking Sauces, Dressing, Food, Food Beverage,
 Grocery, Grocery & Gourmet Food, Grocery Food, Grypoupn, Ketchup & Mustard, Ketchups Mustards &
 Mayo, Kraft Foods, Meal Solutions Grains & Pasta, Mustard & Horseradish, Sandwich Sides, Sauces",
"description" : "Grey Poupon Mustard Dijon",
"formattedNutrition" : {
   "Biotin" : {},
   "Calcium" : {
      "dv" : "0%",
      "qty" : null
   },
   "Calories" : {
      "dv" : null,
      "qty" : "5"
   },
   "Calories From Fat" : {},
   "Calories Per Serving" : {},
   "Calories from Fat": {
      "dv" : null,
      "qty" : "0"
   },
   "Carbohydrates" : {},
   "Chloride" : {},
   "Cholesterol" : {},
   "Chromium" : {},
   "Copper" : {},
   "Dietary Fiber" : {},
   "Fiber" : {},
   "Folic Acid" : {},
   "Insoluble Fiber" : {},
   "Iodine" : {},
   "Iron" : {
      "dv" : "0%",
      "qty" : null
   },
   "Magnesium" : {},
   "Manganese" : {},
   "Molybdenum" : {},
   "Monounsaturated Fat" : {},
   "Niacin" : {},
   "Other Carbohydrates" : {},
   "Pantothenic Acid" : {},
   "Phosphorous" : {},
   "Polyunsaturated Fat" : {},
   "Potassium" : {},
   "Protein" : {
      "dv" : null,
      "qty" : "0g"
```

```json
        },
        "Riboflavin" : {},
        "Saturated Fat" : {},
        "Selenium" : {},
        "Serving Per Container" : {},
        "Serving Size" : {
            "dv" : null,
            "qty" : "1.0 tsp"
        },
        "Servings Per Container" : {
            "dv" : null,
            "qty" : "57"
        },
        "Sodium" : {
            "dv" : "5%",
            "qty" : "120mg"
        },
        "Soluble Fiber" : {},
        "Sugar" : {},
        "Sugar Alcohol" : {},
        "Sugars" : {},
        "Thiamin" : {},
        "Total Calories" : {},
        "Total Calories From Fat" : {},
        "Total Carbohydrate" : {},
        "Total Carbohydrates" : {},
        "Total Fat" : {
            "dv" : "0%",
            "qty" : "0g"
        },
        "Total carbohydrates" : {
            "dv" : "0%",
            "qty" : "0g"
        },
        "Trans Fat" : {},
        "Vitamin A" : {
            "dv" : "0%",
            "qty" : null
        },
        "Vitamin B12" : {},
        "Vitamin B6" : {},
        "Vitamin C" : {
            "dv" : "0%",
            "qty" : null
        },
        "Vitamin D" : {},
        "Vitamin E" : {},
        "Vitamin K" : {},
        "Vitamin K1" : {},
        "Zinc" : {}
    },
    "gcp" : {
        "address" : "3 Lakes Dr",
        "address2" : null,
        "city" : "Northfield",
        "company" : "Kraft Foods Group, Inc.",
        "contact" : null,
        "country" : "US",
        "fax" : null,
        "gcp" : "0054400",
        "gln" : "46800000000",
        "phone" : "(847) 646-2454",
        "postal_code" : "60093",
        "state" : "IL"
    },
    "gcp_gcp" : "54400",
    "image" : "http://images.salsify.com/image/upload/s--V03IoFC1--
/h_500,w_500/xvvzvkltnnt3atmspm0h.jpg",
    "ingredients": "Water, Vinegar, Mustard Seed, Salt, White Wine, Fruit Pectin, Citric Acid, Tartaric
Acid, Sugar, Spice.",
    "language" : "en",
```

JSON Single-transaction API Functions

```
    "manufacturer" : {
        "address" : null,
        "address2" : null,
        "city" : null,
        "company" : "Kraft Foods North America, Inc",
        "contact" : null,
        "country" : null,
        "phone" : null,
        "postal_code" : null,
        "state" : null
    },
    "nutrition" : "Serving Size 1.0 tsp Servings Per Container 57 Amount Per Serving Calories 5 Calories
from Fat 0 % Daily Value* Total Fat 0g 0% Sodium 120mg 5% Total Carbohydrate 0g 0% Protein 0g Vitamin A
0% Vitamin C 0% Calcium 0% Iron 0%",
    "product_web_page" : "https://www.kraftheinz-foodservice.com/product/10054400000525/GREY-POUPON-
Dijon-Mustard-Squeeze-Bottle-10-oz-Bottle-Pack-of-12?categoryid=0000070248",
    "return_code" : "000",
    "return_message" : "Success",
    "thumbnail" : {},
    "uom" : "10.0 ounces Bottle",
    "upc_code" : "0054400015034",
    "usage" : "Grey Poupon Dijon Mustard. Made with white wine. Directions Refrigerate after opening. Do
not freeze. Shake well before using. Remove inner seal.",
    "website" : "www.kraftheinz-foodservice.com/"
}
```

V3 RESULTS

The V3 endpoint allows access to System 4 numbers and text-based search. It also can provide price
history for each item.

PRICE DATA

```
        http://digit-eyes.com/gtin/v3_0/?upc_code=x&app_key=x&signature=x&language=x&field_names=x,
        price
```

The results:

```
{
    "entries" : "1",
    "products" : [
        {
            "brand" : "Grey Poupon",
            "description" : "Grey Poupon Mustard Dijon",
            "directions" : "http://www.digit-eyes.com/cgi-
bin/digiteyes.fcgi?action=quickScan&k=/0Sd9/4LiohG&iP=3&upcCode=0054400015034&l=en",
            "formattedNutrition" : {
                "Calcium" : {
                    "dv" : "0%",
                    "qty" : null
                },
                "Calories" : {
                    "dv" : null,
                    "qty" : "5"
                },
                "Calories from Fat" : {
                    "dv" : null,
                    "qty" : "0"
                },
                "Iron" : {
                    "dv" : "0%",
```

```
          "qty" : null
        },
        "Protein" : {
          "dv" : null,
          "qty" : "0g"
        },
        "Serving Size" : {
          "dv" : null,
          "qty" : "1.0 tsp"
        },
        "Servings Per Container" : {
          "dv" : null,
          "qty" : "57"
        },
        "Sodium" : {
          "dv" : "5%",
          "qty" : "120mg"
        },
        "Total Fat" : {
          "dv" : "0%",
          "qty" : "0g"
        },
        "Total carbohydrates" : {
          "dv" : "0%",
          "qty" : "0g"
        },
        "Vitamin A" : {
          "dv" : "0%",
          "qty" : null
        },
        "Vitamin C" : {
          "dv" : "0%",
          "qty" : null
        }
      },
      "gcp" : {
        "address" : "3 Lakes Dr",
        "address2" : null,
        "city" : "Northfield",
        "company" : "Kraft Foods Group, Inc.",
        "contact" : null,
        "country" : "US",
        "fax" : null,
        "gcp" : "0054400",
        "gln" : "46800000000",
        "phone" : "(847) 646-2454",
        "postal_code" : "60093",
        "state" : "IL"
      },
      "gcp_gcp" : "54400",
      "image" : "https://i5.walmartimages.com/asr/666478f6-aea4-4641-9f76-
b239eda920cc_2.3d0ae576c416d4422021782ae547f671.jpeg",
      "ingredients": "Water, Vinegar, Mustard Seed, Salt, White Wine, Fruit Pectin, Citric Acid,
Tartaric Acid, Sugar, Spice.",
      "language" : "en",
      "manufacturer" : {
        "address" : null,
        "address2" : null,
        "city" : null,
        "company" : "Kraft Foods North America, Inc",
        "contact" : null,
```

```
            "country" : null,
            "phone" : null,
            "postal_code" : null,
            "state" : null
        },
        "nutrition" : "Serving Size 1.0 tsp Servings Per Container 57 Amount Per Serving Calories 5
Calories from Fat 0 % Daily Value* Total Fat 0g 0% Sodium 120mg 5% Total Carbohydrate 0g 0% Protein 0g
Vitamin A 0% Vitamin C 0% Calcium 0% Iron 0%",
        "prices" : {
            "entries" : 5,
            "offers" : [
                {
                    "currencyCode" : "USD",
                    "date" : "2020-06-14 09:11:46",
                    "price" : "3.6900",
                    "url" : "http://grocery.harristeeter.com/pd/Grey-Poupon/Mustard-Dijon/10-
oz/054400015034/"
                },
                {
                    "currencyCode" : "AUD",
                    "date" : "2020-06-09 15:13:42",
                    "price" : "52.1600",
                    "url" : "https://www.fishpond.com.au/Kitchen/Grey-Poupon-Dijon-Mustard-10-
oz/0054400015034"
                },
                {
                    "currencyCode" : "USD",
                    "date" : "2020-06-05 09:57:40",
                    "price" : "4.5900",
                    "url" : "https://shopfoodex.com/grey-poupon-dijon-mustard-squeeze-p-7575.html"
                },
                {
                    "currencyCode" : "USD",
                    "date" : "2020-06-02 04:03:13",
                    "price" : "29.5900",
                    "url" : "https://www.fishpond.com/Kitchen/Grey-Poupon-Dijon-Mustard-10-
oz/0054400015034"
                },
                {
                    "currencyCode" : "USD",
                    "date" : "2020-04-12 09:08:07",
                    "price" : "2.9800",
                    "url" : "https://www.walmart.com/ip/Grey-Poupon-Dijon-Mustard-10-0-oz-Squeeze-
Bottle/10801675"
                }
            ]
        },
        "product_web_page" : "http://directionsforme.org/item/3306684",
        "return_code" : 0,
        "return_message" : "Success",
        "uom" : "10.0 ounces Bottle",
        "upc_code" : "0054400015034",
        "usage" : "Grey Poupon Dijon Mustard. Made with white wine. Directions Refrigerate after
opening. Do not freeze. Shake well before using. Remove inner seal.",
        "website" : "directionsforme.org/"
    }
  ],
  "return_code" : "000",
  "return_message" : "Success",
  "upc_code" : "0054400015034"
}
```

JSON Single-transaction API Functions

Usage, Ingredients, and Nutrition will not contain HTML. A complete list of possible data fields in the "formattedNutrition" element appears in Appendix E of this document.   The formatted data includes allergy information and cautions.

SYSTEM 4 DATA

V3 allows retrieval of private ("System4") numbers. Since these numbers are unique only to the business that creates them:

- additional fields are provided that have the name of the business that is using the number.
- It is possible that there will be more than a single product response. When handling these numbers, the designer should plan to present all as choices to the user and allow them to select the choice that best matches their product.
- See Appendix A for more information on these numbers.

As will be discussed later in this section, the text-based search implemented in V3 may also return multiple records.

The results include a confirming copy of the search argument (identified as "upc_code" or "search_text", depending on the search request) plus a count of the result. The V3 product data is structured into a "products" segment. The GCP and manufacturer segments are meaningless in the context of System 4 numbers and are not included in the results; instead, the "issuer" segment identifies the business entity that provided the information about the product.   Thumbnail data is not available for System 4 products, but if the issuer provided an image, the address of that image is available.

To obtain system 4 results, the value ",system4" should be added to the fieldname list:

```
http://digit-eyes.com/gtin/v3_0/?upc_code=x&app_key=x&signature=x&language=x&field_names=x,
system4
```

The following example shows an example where the same private number has been used for (a) plant food, (b) a private-label brand of vodka and (c) a particular vintage of wine.   Because our data comes from many sources, note that the same item may be listed under multiple names. In this case, the State of Idaho has a formal naming convention for the "Locations" red wine made by Orin Swift "Wa Washington Red Blend NV" that is not consistent with the descriptive name used for marketing:

```
{
   "entries" : "5",
   "return_code" : "000",
   "return_message" : "Success",
   "upc_code" : "0416000336108"
   "products" :      "products" : [
      {
```

```
        "brand" : null,
        "businessName" : "Central Garden Distribution",
        "businessUrl" : "www.centralgarden.com/",
        "description" : "Bgi Tomatogain Plant Food Granules 8-16-16 12ea / 2LB",
        "formattedNutrition" : null,
        "image" :
"https://www.centralgarden.com/media/catalog/product/cache/1/image/229x151/9df78eab33525d08d6e5fb8d271
36e95/1/0/100532892-a.jpg",
        "ingredients" : null,
        "issuer": {
            "address" : "http://www.centralgarden.com/contact-us/",
            "address2" : null,
            "city" : null,
            "company" : "Central Garden Distribution",
            "contact" : null,
            "country" : "US",
            "phone" : "(866) 811-1395",
            "postal_code" : null,
            "state" : null
        },
        "language" : "en",
        "nutrition" : null,
        "prices" : {
            "entries" : 0,
            "offers" : []
        },
        "product_web_page" : "https://www.centralgarden.com/fertilizer/granules/bgi-tomatogain-
tomato-food-12ea-8-16-16.html",
        "uom" : null,
        "upc_code" : "0416000336108",
        "usage": "TomatoGain granular plant food is easy to use and formulated to provide immediately
available nutrients to promote vigorous growth and high yields while preventing blossom-end rot and
splitting. Works well on all vegetables and some fruits, such as but no",
        "website" : "www.centralgarden.com/"
    },
    {
        "brand" : null,
        "businessName" : "Kosher Wine Cellar",
        "businessUrl" : "www.kosherwinecellar.co.uk/",
        "description" : "Checkmate Vodka",
        "formattedNutrition" : null,
        "image" : null,
        "ingredients" : null,
        "language" : "en",
        "nutrition" : null,
        "prices" : {
            "entries" : 1,
            "offers" : [
                {
                    "currencyCode" : "GBP",
                    "date" : "2020-04-02 03:37:45",
                    "price" : "26.9900",
                    "url" : "https://kosherwinecellar.co.uk/vodka/2306-checkmate-vodka-
416000336108.html"
                }
            ]
        },
        "product_web_page" : "https://kosherwinecellar.co.uk/vodka/2306-checkmate-vodka-
416000336108.html",
        "uom" : null,
        "upc_code" : "0416000336108",
        "usage": "Made from super premium distilled vodka with entirely natural ingredients.
Checkmate offers the true taste of smooth vodka in every sip. So pull out a glass and enjoy!",
```

```
            "website" : "kosherwinecellar.co.uk/"
        },
        {
            "brand" : null,
            "businessName" : "Triangle Wine Company",
            "businessUrl" : "www.trianglewineco.com/",
            "description" : "Locations Wa Washington Red (750ML)",
            "formattedNutrition" : null,
            "image" : "https://storage-
trianglewine.comcash.com/images/products/thumb_b425dea28f0060688b01570068d9c477.png",
            "ingredients" : null,
             issuer": {
                "address" : "2613 Chatham Road",
                "address2" : null,
                "city" : "Springfield",
                "company" : "The Corkscrew",
                "contact" : null,
                "country" : "US",
                "phone" : null,
                "postal_code" : "62704",
                "state" : "IL"
            },
            "language" : "en",
            "nutrition" : null,
            "prices" : {
                "entries" : 0,
                "offers" : []
            },
            "product_web_page" : "https://www.trianglewineco.com/catalog/product/12423/locations-wa-
washington-red.html",
            "uom" : "750 ML",
            "upc_code" : "0416000336108",
            "usage" : "",
            "website" : "www.trianglewineco.com/"
        },
        {
            "brand" : null,
            "businessName" : "theoriginalwineclub.com",
            "businessUrl" : "theoriginalwineclub.com/",
            "description" : "Orin Swift Locations Wine Wa 4 Red",
            "formattedNutrition" : null,
            "image" :
"https://theoriginalwineclub.com/media/catalog/product/cache/1/image/9df78eab33525d08d6e5fb8d27136e95/
w/i/wine_109296.png",
            "ingredients" : null,
            "language" : "en",
            "nutrition" : null,
            "prices" : {
                "entries" : 1,
                "offers" : [
                    {
                        "currencyCode" : "USD",
                        "date" : "2020-05-31 09:00:12",
                        "price" : "16.9800",
                        "url" : "https://theoriginalwineclub.com/orin-swift-locations-wine-wa-4-red.html"
                    }
                ]
            },
            "product_web_page" : "https://theoriginalwineclub.com/orin-swift-locations-wine-wa-4-
red.html",
            "uom" : "750 ML",
            "upc_code" : "0416000336108",
            "usage" : "",
```

JSON Single-transaction API Functions

```
        "website" : "theoriginalwineclub.com/"
    },
    {
        "brand" : "14. 49",
        "businessName" : "Idaho ABC",
        "businessUrl" : "post.idaho.gov/",
        "description" : "Wa Washington Red Blend NV",
        "formattedNutrition" : null,
        "image" : null,
        "ingredients" : null,
        "language" : "en",
        "nutrition" : null,
        "prices" : {
            "entries" : 3,
            "offers" : [
                {
                    "currencyCode" : "USD",
                    "date" : "2020-05-09 02:49:51",
                    "price" : "167.9201",
                    "url" : "https://post.idaho.gov/AbcReporting/pricePosting/search?page=1225&size=50"
                },
                {
                    "currencyCode" : "USD",
                    "date" : "2020-05-09 02:49:51",
                    "price" : "14.4900",
                    "url" : "https://post.idaho.gov/AbcReporting/pricePosting/search?page=1225&size=50"
                },
                {
                    "currencyCode" : "USD",
                    "date" : "2020-05-09 02:49:42",
                    "price" : "12.5900",
                    "url" : "https://post.idaho.gov/AbcReporting/pricePosting/search?page=1224&size=50"
                }
            ]
        },
        "product_web_page" :
 "https://post.idaho.gov/AbcReporting/pricePosting/search?page=1225&size=50",
        "uom" : null,
        "upc_code" : "0416000336108",
        "usage" : "",
        "website" : "post.idaho.gov/"
    }
  ]
}
```

TEXT-BASED SEARCH

V3 allows retrieval based on the brand or product name. For instance, to locate listings for a dewalt DCD796B cordless drill, a search might be performed for the value "DCD796B":

https://digit-eyes.com/cgi-bin/digiteyes/gtinTemp.cgi?search_text=DCD796B&limit=10&json=1&field_names=brand,description,language,image,upc,price&language=en&app_key=[k]&signature=[x]

might yield the following results

```
{
        entries: "2",
        products:
        [
                {
                        brand: null,
                        description: "Dewalt DCD796B 20V Max XR Brushless Compact Hammerdrill Bare",
                        image: "http://cdn.shopify.com/s/files/1/0105/8992/9572/products/DeWalt-
DCD796B_01_97a8d520-fd48-4230-b699-e697f4c725e8_1200x630.jpg?v=1620425572",
                        language: "en",
                        prices:
                        {
                                entries: 1,
                                offers:
                                [
                                        {
                                        currencyCode: "USD",
                                        date: "2021-06-19 21:04:37",
                                        price: "164.6000",
                                        url: "https://mytoolstore.com/products/dewalt-dcd796b-20v-max-
xr-brushless-compact-hammerdrill-bare-1"
                                        }
                                ]
                        },
                        return_code: 0,
                        return_message: "Success",
                        thumbnail: { },
                        upc_code: "0885911513401"
                },
                {
                        brand: "Dewalt",
                        description: "Dewalt DCD796B 20-volt Max XR Lithium-ion 1 / 2 In Cordless
Brushless Compact",
                        image: "https://i5.walmartimages.com/asr/d3eca4d6-12b3-4ede-9325-
ea90ecaec7e7.0ded1d7a0630da0cce060ed739cd0e60.jpeg",
                        language: "en",
                        prices:
                        {
                                entries: 4,
                                offers:
                                [
                                        {
                                                currencyCode: "USD",
                                                date: "2020-10-22 16:09:52",
                                                price: "89.9500",
                                                url: "http://www.walmart.com/ip/Dewalt-DCD796B-20-Volt-
Max-XR-Lithium-Ion-1-2-in-Cordless-Brushless-Compact/266383014"
                                        },
                                        {
                                                currencyCode: "USD",
                                                date: "2020-06-17 15:26:27",
                                                price: "92.9500",
                                                url: "http://www.walmart.com/ip/Dewalt-DCD796B-20-Volt-
Max-XR-Lithium-Ion-1-2-in-Cordless-Brushless-Compact/266383014"
                                        },
                                        {
                                                currencyCode: "USD",
                                                date: "2020-06-12 13:08:23",
                                                price: "92.9500",
                                                url: "https://www.walmart.com/browse/home-
improvement/dewalt-tools/dewalt/drills-drivers-parts-
accessories/1072864_1031899_1574888_2643784_1794063/YnJhbmQ6REVXQUxUfHxjYXRlZ29yeTpEcmlsbHMgJiiBEcml2
ZXJzOiiBQYXJ0cyAmIEFjY2Vzc29yaWVz?page=1"
                                        },
                                        {
                                                currencyCode: "USD",
                                                date: "2020-06-12 01:13:05",
                                                price: "92.9500",
                                                url: "https://www.walmart.com/browse/auto-tires/power-
drills/dewalt/drills-drivers-parts-
```

```
accessories/91083_1074767_4582892_5036061_1066925_3413875/YnJhbmQ6REVXQUxUfHxjYXRlZ29yeTpEcmlsbHMgJii
BEcml2ZXJzOiiBQYXJ0cyAmIEFjY2Vzc29yaWVz?page=1"
                                    }
                                ]
                    },
                    return_code: 0,
                    return_message: "Success",
                    thumbnail: { },
                    upc_code: "7426885126322"
            }
        ],
        return_code: "000",
        return_message: "Success",
        search_text: "DCD796B"
  }
```

UPC/EAN DELETION/CORRECTION REQUEST

The JSON request for deletion is submitted as a DELETE operation to the endpoint:

```
http://digit-eyes.com/gtin/v2_0/?upc_code=x&app_key=x&signature=x&language=x[&reason=x]
```

This transaction does not actually delete the code but flags it for the attention of the data curator. After flagging, the listing becomes invisible to other users until either confirmed or denied. If the code cannot be confirmed as accurate, the curator will delete it.

Delete transactions may be submitted only for individual UPCs, not for groups identified by text.

Required parameters:

1. **upc_code**: is the UPC, EAN, GTIN, APN or JPN code for which deletion is requested;
2. **app_key:** is the application identification key of the person or entity making the request (this is issued with the API key).
3. **signature:** created using the upc_code and auth_key. Code examples are in Appendix D;
4. **language**: the two-character ISO 6389-1 language code for the preferred language of response ("en," "es," etc.)

Optional (but recommended) Parameters:

5. **reason:** up to 512 characters of human-readable text explaining the cause of the request.   This is an optional field, but failure to provide it may cause the data curator to deny the request or to send an email inquiring as to the reason if the code appears to be valid.

Demonstration: http://www.digit-eyes.com/gtin/
Signature calculator: http://www.digit-eyes.com/gtin/
Your keys: https://www.digit-eyes.com/cgi-bin/digiteyes.fcgi?action=myAccount

Note to developers: Before activating your account for production, deletion requests will be parsed and returned with appropriate response codes, but the deletion will not be performed.

UPC/EAN DELETION/CORRECTION REQUEST RESPONSE

The successful JSON-format UPC deletion request will have http status code 204 and information formatted as shown:

```
{

        "upc_code": "37600138727",

}
```

The JSON response for error transactions will include a return code and message that indicates the reason for the failure per Appendix C.

UPC/EAN UPSERT REQUEST

The request for information upsert in JSON format is submitted as a POST operation to the endpoint:

```
http://digit-eyes.com/gtin/v2_0/
?upc_code=n&app_key=x&signature=x&language=x&description=x[&optional parameters]
```

Upsert transactions may be submitted only for individual UPCs, not for groups identified by text.

Required parameters:

1. **upc_code:** is the UPC, EAN, GTIN, APN or JPN code for which information is desired;
2. **app_key:** is the application identification key of the party making the request;
3. **signature:** created using the upc_code and auth_key. Code examples are in [Appendix D](#);
4. **language**: the ISO 639-1 2 character language code for the preferred language of response ("en"," es", etc.)
5. **description:** up to 255 characters of short descriptive text.

Optional parameters:

6. **uom:** up to 128 characters that describe the item unit of measure or size such as '24 ounces', '12 pieces', or even just "1";
7. **website:** the website where more information can be found about this item (512 characters; the root of the product web page will be used if that is supplied and this field is empty.);
8. **product_web_page:** web page where more information can be found about this item (512 characters);
9. **image:** location of an image of the item (512 characters);
10. **usage:** romance description of the item, usage instructions if available (no length limit);
11. **ingredients:** constituent elements of the item (no length limit);
12. **nutrition:** if nutritional data if appropriate (no length limit);
13. **brand:** product brand;
14. **categories:** comma-separated list of categories or tags that should be associated with the item, listed in increasing order of importance (the tag on the right end is treated as more important than the tag on the left when the system evaluated lexical proximity);
15. **manufacturer_name[1]:** manufacturer name;
16. **manufacturer_address:** manufacturer location: street address;
17. **manufacturer_address2:** manufacturer location: suite, additional address;
18. **manufacturer_city:** manufacturer location: city;

---

[1] At a minimum, city and country must be provided with the manufacturer name if manufacturer information is supplied

19. **manufacturer_state:** manufacturer location: state (2 characters max);

20. **manufacturer_postal_code:** manufacturer location: postal code;

21. **manufacturer_country:** manufacturer country;

22. **manufacturer_phone:** manufacturer phone;

23. **manufacturer_contact:** manufacturer email or contact name.

Note to developers: Before activating your account for production, upsert requests will be parsed and returned with appropriate response codes, but will not be executed. After activation, all upsert requests are visible but provisional until reviewed and approved by the data curator.

Signature calculator: http://www.digit-eyes.com/gtin/
Your keys: https://www.digit-eyes.com/cgi-bin/digiteyes.fcgi?action=myAccount

UPC/EAN UPSERT REQUEST RESPONSE

The successful JSON-format upsert request will have http status code 201 and information formatted as shown:

```
{

        "upc_code": "37600138727",

}
```

The JSON response for error transactions will include a return code and message that indicates the reason for the failure per Appendix C.

XML (SOAP) SINGLE-TRANSACTION API FUNCTIONS

UPC/EAN DATA REQUEST

The request for information in XML format is submitted to the endpoint

In V2 format:

```
http://digit-eyes.com/gtin/v2_0/xml/?upc_code=x&app_key=x&signature=x&language=x&field_names=x
```

In V3 format:

```
http://digit-eyes.com/gtin/v3_0/xml/?upc_code=x&app_key=x&signature=x&language=x&field_names=x
```

Where "x" is the value of the upc code, your app_key and the signature you've computed from the upc_code and your private key.   The parameter field_names should be "all" or the names of the fields you want (see below)

If you want price data (an extra charge item) include the value ",price" in the fieldname list:

```
http://digit-
eyes.com/gtin/v3_0/xml/?upc_code=x&app_key=x&signature=x&language=x&field_names=x,price
```

Required parameters:

1. **upc_code:** is the UPC, EAN, GTIN, APN or JPN code for which information is desired;
2. **app_key:** is the application identification key of the party making the request;
3. **signature:** created using the upc_code and auth_key. Code examples are in Appendix D;
4. **language**: the ISO 639-1 2- character language code for the preferred language of response ("en", "es", etc.)

Optional parameters:

5. **field_names:** comma-separated list of the fields requested from the database. Acceptable field names are:

    Acceptable field requests for both V2 and V3 are:

    o   **description:** short product name;
    o   **uom:** unit of measure;
    o   **usage**: romance description / instructions (if available);
    o   **brand**: product labeling (if available);
    o   **language:** 2-character ISO code for the preferred language of listing;

- o **website:** the authoritative website for the product (typically the seller or manufacturer website);
- o **product_web_page:** an individual web page that describes the product, if available;
- o **nutrition:** nutrition information for the product (if available);
- o **formattedNutrition**: nutrition information divided into discrete elements (US-format nutrition information only and only if available for the item);
- o **ingredients:** ingredients of the product (if available);
- o **manufacturer:** Manufacturer's name and address elements (if available);
- o **gcp_name_address:** GCP owners name and address elements (if available); Please note that if you ask for GCP and the UPC information is not available, the system will return the GCP information and set http code 200. You should check the response_code to determine whether only GCP information was returned.
- o **image:** the web address of the manufacturer's image of the product (if available);
- o **thumbnail:** a small image of the item, as well as the dimensions;
- o **categories:** list of tags or descriptive categories associated with this item.

V3 supports the additional field requests (the V2 endpoint ignores them)

- o **prices:** an array of prices, including the currency, vendor business name, vendor address, and date the price data was collected;
- o **system4:** value 0 or 1; if 1 and a private/system 4 UPC number is provided. more than one UPC record may be returned if a System 4 number is provided. In V2, data requests for System 4 numbers are ignored.

Please note: for backward compatibility, the parameters used by API versions 1.0 and 1.1 are still supported, but the use is deprecated:

- • u: synonym for upc_code ;
- • k: synonym for app_key;
- • m: synonym for signature;
- • l: synonym for language.

Demonstration: http://www.digit-eyes.com/gtin/
Signature calculator: http://www.digit-eyes.com/gtin/
Your keys: https://www.digit-eyes.com/cgi-bin/digiteyes.fcgi?action=myAccount

UPC/EAN DATA REQUEST RESPONSE

V2 RESULTS

The successful XML-format data request to the V2 endpoint will have http status code 200 and information formatted as shown:

```xml
<?xml version="1.0" standalone="yes"?>
<UPCInformation>
<Version>2.0</Version>
<Header><Sender>
  <Name>Digital Miracles</Name>
</Sender>
</Header>

  <Body>
    <ReturnCode>000</ReturnCode>
    <ReturnMessage>Success</ReturnMessage>
    <UPC>0054400015034</UPC>
    <Uom>10.0 ounces Bottle</Uom>
    <Description><![CDATA[Grey Poupon Mustard Dijon]]></Description>
    <Language>en</Language>
    <Website>www.kraftheinz-foodservice.com</Website>
    <ProductWebPage> https://www.kraftheinz-foodservice.com/product/10054400000525/GREY-POUPON-Dijon-
Mustard-Squeeze-Bottle-10-oz-Bottle-Pack-of-12?categoryid=0000070248</ProductWebPage>
    <Usage><![CDATA[Grey Poupon Dijon Mustard. Made with white wine. Directions Refrigerate after
opening. Do not freeze. Shake well before using. Remove inner seal.]]></Usage>
    <Ingredients><![CDATA[Water, Vinegar, Mustard Seed, Salt, White Wine, Fruit Pectin, Citric Acid,
    Tartaric Acid, Sugar, Spice.]]></Ingredients>
    <Nutrition><![CDATA[Serving Size 1.0 tsp Servings Per Container 57 Amount Per Serving Calories 5
    Calories from Fat 0 % Daily Value* Total Fat 0g 0% Sodium 120mg 5% Total Carbohydrate 0g 0%
    Protein 0g Vitamin A 0% Vitamin C 0% Calcium 0% Iron 0%]]></Nutrition>
    <Image> http://images.salsify.com/image/upload/s--V03IoFC1--
    /h_500,w_500/xvvzvkltnnt3atmspm0h.jpg</Image>
    <Brand><![CDATA[Grey Poupon]]></Brand>
    <Categories><![CDATA[Amy's Kitchen Inc, Available Items, Canned Dry & Packaged Foods, Condiment &
    Sauces, Condiments, Condiments & Salad Dressings, Condiments & Sandwich Toppings, Condiments &
    Sauces, Condiments Dressings, Condiments Mustard, Condiments Mustards, Condiments Pickles &
    Relishes, Condiments Sauces & Spices, Condiments Vinegars & Oils, Cooking & Baking, Cooking
    Sauces, Dressing, Food, Food Beverage, Grocery, Grocery & Gourmet Food, Grocery Food, Grypoupn,
    Ketchup & Mustard, Ketchups Mustards & Mayo, Kraft Foods, Meal Solutions Grains & Pasta, Mustard &
    Horseradish, Sandwich Sides, Sauces]]></Categories>
    <Manufacturer>
      <Company><![CDATA[Kraft Foods North America, Inc]]></Company>
    </Manufacturer>
    <GCP>
      <Company><![CDATA[Kraft Foods Group, Inc.]]></Company>
      <Address>3 Lakes Dr</Address>
      <City>Northfield</City>
      <State>IL</State>
      <Postal_Code>60093</Postal_Code>
      <Country>US</Country>
      <Phone>(847) 646-2454</Phone>
    </GCP>
    <FormattedNutrition>
      <serving_size><![CDATA[1.0 tsp]]></serving_size>
      <servings_per_container><![CDATA[57]]></servings_per_container>
      <calories><![CDATA[5]]></calories>
      <total_fat_qty><![CDATA[0g]]></total_fat_qty>
      <total_fat_dv><![CDATA[0%]]></total_fat_dv>
      <sodium_qty><![CDATA[120mg]]></sodium_qty>
      <sodium_dv><![CDATA[5%]]></sodium_dv>
      <protein_qty><![CDATA[0g]]></protein_qty>
      <vitamin_a_dv><![CDATA[0%]]></vitamin_a_dv>
      <vitamin_c_dv><![CDATA[0%]]></vitamin_c_dv>
      <calcium_dv><![CDATA[0%]]></calcium_dv>
      <iron_dv><![CDATA[0%]]></iron_dv>
```

```
    </FormattedNutrition>
  </Body>
</UPCInformation>
```

Usage, Ingredients, and Nutrition may contain character data but will not contain HTML. A complete list of possible data fields in the "formattedNutrition" element is included in Appendix E of this document. Formatted nutrition data includes allergy and other cautionary information.

V3 RESULTS

The V3 endpoint allows access to System 4 numbers. System 4 numbers are not unique, which means that multiple products may be available for a single number. The UPC and a count of the products is provided before any product data. The product data follows, structured into a "products" segment. The GCP and manufacturer segments are meaningless in the context of System 4 numbers; instead, the "issuer" segment identifies the business entity that provided the information about the product.   Thumbnail data is not available for System 4 products, but if the issuer provided an image, the address of that image is available.

If you want private URLs (system 4 URLs), include the value ",system4" in the fieldname list:

```
http://digit-
eyes.com/gtin/v3_0/xml/?upc_code=x&app_key=x&signature=x&language=x&field_names=x,system4
```

```
<?xml version="1.0" standalone="yes"?>
<UPCInformation>
<Version>3.0</Version>
<Header><Sender>
  <Name>Digital Miracles</Name>
</Sender>
</Header>

  <Body>
    <ReturnCode>000</ReturnCode>
    <ReturnMessage>Success</ReturnMessage>
    <UPC>4160003361084</UPC>
    <Entries>3</Enries>
    <Products>
      <Product>
            <Uom>2 LB</Uom>
            <Description><![CDATA[Bgi Tomatogain Plant Food Granules 8-16-16 12ea /
            2LB]]></Description>
            <Language>en</Language>
            <Website>www.centralgarden.com/</Website>
            <ProductWebPage> https://www.centralgarden.com/fertilizer/granules/bgi-tomatogain-
            tomato-food-12ea-8-16-16.html </ProductWebPage>
            <Usage><![CDATA[TomatoGain granular plant food is easy to use and formulated to provide
            immediately available nutrients to promote vigorous growth and high yields while
            preventing blossom-end rot and splitting. Works well on all vegetables and some
            fruits]]></Usage>
            <Ingredients><Ingredients>
            <Nutrition>Nutrition>
            <Image>https://www.centralgarden.com/media/catalog/product/cache/1/image/229x151/9df78ea
            b33525d08d6e5fb8d27136e95/1/0/100532892-a.jpg </Image>
            <Brand></Brand>
            <Categories></Categories>
            <Issuer>
              <Company><![CDATA[Central Garden Distribution Company>
              <Address>http://www.centralgarden.com/contact-us </Address>
```

```
            <City></City>
            <State></State>
            <Postal_Code>60093</Postal_Code>
            <Country>US</Country>
            <Phone>"(866) 811-1395</Phone>
          </Issuer>
      </Product>
      <Product>
            <Uom></Uom>
            <Description><![CDATA[Eno Leave No Trace Doublenest Hammock | Rei Co-op]]></Description>
            <Language>en</Language>
            <Website>www.rei.com/</Website>
            <ProductWebPage>https://www.rei.com/product/886801/eno-leave-no-trace-doublenest-
            hammock</ProductWebPage>
            <Usage></Usage>
            <Ingredients><Ingredients>
            <Nutrition>Nutrition>
            <Image>https://www.rei.com/media/product/886801g</Image>
            <Brand>REI</Brand>
            <Categories></Categories>
          <Issuer>
            <Company><![CDATA[Central Garden Distribution Company.]]></Company>
            <Address></Address>
            <City>Seattle</City>
            <State>WA</State>
            <Postal_Code></Postal_Code>
            <Country>US</Country>
            <Phone>866) 811-1395</Phone>
          </Issuer>
      </Product>
      <Product>
            <Uom>750 ML</Uom>
            <Description><![CDATA[Eno Leave No Trace Doublenest Hammock | Rei Co-op]]></Description>
            <Language>en</Language>
            <Website>www.thecorkscrew.com/</Website>
            <ProductWebPage> https://www.trianglewineco.com/catalog/product/12423/locations-wa-
            washington-red.html"</ProductWebPage>
            <Usage></Usage>
            <Ingredients><Ingredients>
            <Nutrition>Nutrition>
            <Image>https://storage-
            trianglewine.comcash.com/images/products/thumb_b425dea28f0060688b01570068d9c477.png
            </Image>
            <Brand>Locations</Brand>
            <Categories>Orin Swift</Categories>
          <Issuer>
            <Company><![CDATA[Central Garden Distribution Company.]]></Company>
            <Address>2613 Chatham Road</Address>
            <City>Springfield</City>
            <State>IL</State>
            <Postal_Code>62704</Postal_Code>
            <Country>US</Country>
            <Phone></Phone>
          </Issuer>
      </Product>
    </Products>
  </Body>
</UPCInformation>
```

Additionally, the V3 endpoint may return a price segment if prices are requested:

If you want price data (an extra charge item) include the value ",price" in the fieldname list:

```
http://digit-eyes.com/gtin/v3_0/xml/?upc_code=x&app_key=x&signature=x&language=x&field_names=x,price
```

```
    <Offers>
    <Offer>
      <CurrencyCode>AUD</CurrencyCode>
```

```
    <Date>2020-06-09 15:13:42</Date>
    <Price>52.1600</Price>
    <URL>https://www.fishpond.com.au/Kitchen/Grey-Poupon-Dijon-Mustard-10-oz/0054400015034</URL>
</Offer>
<Offer>
    <CurrencyCode>USD</CurrencyCode>
    <Date>2020-06-05 09:57:40</Date>
    <Price>4.5900</Price>
    <URL>https://shopfoodex.com/grey-poupon-dijon-mustard-squeeze-p-7575.html</URL>
</Offer>
<Offer>
    <CurrencyCode>USD</CurrencyCode>
    <Date>2020-06-02 04:03:13</Date>
    <Price>29.5900</Price>
    <URL>https://www.fishpond.com/Kitchen/Grey-Poupon-Dijon-Mustard-10-oz/0054400015034</URL>
</Offer>
<Offer>
    <CurrencyCode>USD</CurrencyCode>
    <Date>2020-04-12 09:08:07</Date>
    <Price>2.9800</Price>
    <URL>https://www.walmart.com/ip/Grey-Poupon-Dijon-Mustard-10-0-oz-Squeeze-Bottle/10801675</URL>
</Offer>
</Offers>
```

The V3 Endpoint can do system 4 numbers and also search by text.

UPC/EAN CORRECTION/DELETION REQUEST

The request for deletion to which an XML response is requested includes the "r" parameter:

```
http://digit-eyes.com/gtin/v2_0/xml/?upc_code=n&app_key=n&signature=x&language=x&r=1&reason=x
```

This transaction does not actually delete the code but flags it for the attention of the data curator. After flagging, the listing becomes invisible to other users until either confirmed or denied. If the code cannot be verified, the curator will delete it.

Correction transactions may be submitted only for individual UPCs, not for groups identified by text.

Required parameters:

1. **upc_code:** is the UPC, EAN, GTIN, APN or JPN code for which information is desired;
2. **app_key:** is the application identification key of the party making the request;
3. **signature:** created using the upc_code and auth_key. Code examples are in Appendix D;
4. **language**: the two-character ISO language code for the preferred language of response ("en"," es", etc.)
5. **r**: suspense requested flag: the only acceptable value is '1'.

Optional (but recommended) Parameters:

6. **reason:** up to 512 characters of human-readable text explaining the cause of the request. This is an optional field, but failure to provide it may cause the data curator to deny the request or to send an email inquiring as to the reason if the code appears to be valid.

Signature calculator: http://www.digit-eyes.com/gtin/
Your keys: https://www.digit-eyes.com/cgi-bin/digiteyes.fcgi?action=myAccount

Note to developers: Before activating your account for production, deletion requests will be parsed and returned with appropriate response codes, but no records will actually be deleted.

UPC/EAN DELETION/CORRECTION REQUEST RESPONSE

The successful XML-format UPC deletion request will have http status code 204 and information formatted as shown:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<UPCDelete>
  <Version>2.0</Version>
  <Header>
    <Sender>
      <Name>Digital Miracles</Name>
    </Sender>
  </Header>
  <Body>
    <ReturnCode>100</ReturnCode>
    <UPC>9780545137645</UPC>
  </Body>
</UPCDelete>
```

UPC/EAN UPSERT REQUEST

An upsert request for which an XML response is required is submitted to the endpoint in the following format:

```
http://digit-eyes.com/gtin/v2_0/xml/
?upc_code=x&app_key=x&signature=x&language=x&description=x[&optional parameters]
```

Upsert transactions may be submitted only for individual UPCs, not for groups identified by text.

Required parameters

1. **upc_code:** is the UPC, EAN, GTIN, APN or JPN code for which information is desired;
2. **app_key:** is the application identification key of the party making the request;
3. **signature:** created using the upc_code and auth_key. Code examples are in [Appendix D];
4. **language**: the ISO 639-1 2-character language code identifying the preferred language of response ("en"," es", etc.);
5. **description:** up to 255 characters of short descriptive text for the item.

Optional Parameters

6. **uom:** Unit of measure: up to 128 characters that describe the item size such as '24 ounces', '12 pieces', or even just "1".
7. **website:** the web site where more information can be found about this item (512 characters; the root of the product web page will be used if that is supplied and this field is empty.)
8. **product_web_page:** web page where more information can be found about this item (512 characters);
9. **image:** location of an image of the item (512 characters);
10. **usage:** romance description of the item, usage instructions if available (no length limit);
11. **ingredients:** elements from which the item is made (no length limit);
12. **nutrition:** if nutritional data if appropriate (no length limit);
13. **brand:** product brand
14. **categories:** comma-separated list of categories or tags that should be associated with the item, listed in increasing order of importance (most important tag is on the right end of the string, least important on the left);
15. **manufacturer_name[2]:** manufacturer name;
16. **manufacturer_address:** manufacturer location: street address;
17. **manufacturer_address2:** manufacturer location: suite, additional address;

---

[2] At a minimum, city and country must be provided with the manufacturer name when supplying manufacturer information.

18. **manufacturer_city:** manufacturer location: city;

19. **manufacturer_state:** manufacturer location: state (2 characters max);

20. **manufacturer_postal_code:** manufacturer location: postal code;

21. **manufacturer_country:** manufacturer country;

22. **manufacturer_phone:** manufacturer phone;

23. **manufacturer_contact:** manufacturer email or contact name.

Note to developers: Before activating your account for production, upsert requests will be parsed and returned with appropriate response codes, but will not be applied to the database. After activation, all upsert requests are provisional until reviewed and approved by the data curator.

Signature calculator: http://www.digit-eyes.com/gtin/
Your keys: https://www.digit-eyes.com/cgi-bin/digiteyes.fcgi?action=myAccount

UPC/EAN UPSERT REQUEST RESPONSE

The successful XML-format UPC deletion request will have http status code 201 and information formatted as shown:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<UPCUpsert>
  <Version>2.0</Version>
  <Header>
    <Sender>
      <Name>Digital Miracles</Name>
    </Sender>
  </Header>
  <Body>
    <ReturnCode>0</ReturnCode>
    <ReturnCode>Success</ReturnCode>
    <UPC>9780545137645</UPC>
  </Body>
</UPCUpsert>
```

CSV MULTIPLE-TRANSACTION API FUNCTIONS

UPC/EAN DATA REQUEST

This batch feature allows appropriately authorized and authenticated users to submit a list of UPC or EAN numbers for which information is requested. Batch transactions may be submitted only for individual UPCs, not for text arguments.

The endpoint is accessed:

```
https://www.digit-eyes.com/cgi-bin/digiteyes/api.cgi?action=Download
```

The response is an HTML page with a link to a .csv file with the results of the transaction.

The numbers are submitted in a file formatted as comma-separated values (.CSV format). CSV is a standard file format that can be produced by Excel and other similar programs. A sample download-format CSV file is here:

http://www.digit-eyes.com/data/demoDownloadList.csv

The first row of the .csv file conventionally contains the titles of each column of data in the spreadsheet; the rest of the rows contain the data.

Only one field is required for retrieval:

1. **upc code:** UPC or EAN code. Should be 12 or 13 characters (leading zeros may be omitted). If this field cannot be validated, an error message will appear in the output file.

In the first row of the input file, include the names of the fields that should be downloaded, one field name per column. Spacing and capitalization are ignored within the titles, but spelling is important. Fields names that are not recognized will be ignored and an error message shown after processing. Field names that may be requested:

1. **description:** The title (short description) of the item (up to 255 characters)
2. **uom:** Unit of measure: up to 128 characters that describe the item size or weight. Examples might be '24 ounces', '750 ml', '12 pieces', or even just "1".
3. **brand:** product brand
4. **language:** the language of listing (ISO 639-1 2-character language code)
5. **website:** the web site where more information can be found about this item (512 characters; the root of the product web page will be used if that is supplied and this field is empty.)
6. **product web page:** web page where more information can be found about this item (512 characters);

7. **image:** location of the authoritative image of the item (512 characters);
8. **thumbnail URL:** location of a thumbnail image of the item (512 characters). Please note this is an extra charge feature, it is charged only if the requested thumbnail is available and supplied;
9. **thumbnail width:** location of an image of the item (512 characters);
10. **thumbnail height:** location of an image of the item (512 characters);
11. **usage:** romance description of the item, usage instructions if available (no length limit);
12. **ingredients:** elements from which the item is made (no length limit);
13. **nutrition:** nutritional data if available (no length limit);
14. **formattedNutrition:** JSON-formatted nutrition data if available (no length limit; applies to data with US-format nutrition information only.)
15. **categories:** provide a list of product categories and tags that apply to this item in a comma-separated list (no length limit);
16. **manufacturer name:** manufacturer name
17. **manufacturer address:** manufacturer location: street address
18. **manufacturer address 2:** manufacturer location: suite, additional address
19. **manufacturer city:** manufacturer location: city;
20. **manufacturer state:** manufacturer location: state or political entity;
21. **manufacturer postal code:** manufacturer location: postal code
22. **manufacturer country:** manufacturer country
23. **manufacturer phone:** manufacturer phone
24. **manufacturer contact:** manufacturer email or contact name
25. **GCP:** Global Company Prefix
26. **GCP Company:** Company name of the CGP holder
27. **GCP GLN:** Global Location Number
28. **GCP address 1:** GCP holder's location: street address
29. **GCP address 2:** GCP holder's location: suite, additional address
30. **GCP city:** GCP holder's location: city;
31. **GCP state:** GCP holder's location: state or political entity;
32. **GCP postal code:** GCP holder's   location: postal code
33. **GCP country:** GCP holder's country
34. **GCP phone:** GCP holder's phone
35. **GCP contact:** GCP holder's email or contact name
36. **GCP phone:** GCP holder's email or contact name
37. **GCP fax:** GCP holder's email or contact name

V3 data (price data and System 4 numbers) are not currently available in the CSV content.

CSV Multiple-Transaction API

Demonstration: http://www.digit-eyes.com/gtin/
Signature calculator: http://www.digit-eyes.com/gtin/
Your keys: https://www.digit-eyes.com/cgi-bin/digiteyes.fcgi?action=myAccount

UPC/EAN DATA REQUEST RESPONSE

The status and results of the processing response appear in an HTML page with an http status code of 200. The page includes a link to a downloadable .csv-format file. Data is also provided to show:

- The on-screen report lists and UPC/EAN codes that could not be processed due to checksum errors or the presence of non-numeric characters and gives processing totals;
- Within the output:
    - codes that could not be processed have an error messages in the "description" field;
    - codes that were not found have an "item not found" error message in the "description" field;
    - input codes appear in column "A"; if the code had to be expanded or corrected prior to lookup, the corrected code is listed n column "B"

UPC/EAN UPSERT REQUEST

This batch feature allows appropriately authorized and authenticated users to submit a file of UPC/EAN data to be added or updated into the database.   The data is formatted as a comma-separated value (.csv) file. CSV is a common file format that can be produced by spreadsheet programs such as Excel.

The first row of the .csv file conventionally contains the titles of each column in the spreadsheet; the rest of the rows contain the data.

Batch transactions may be submitted only for individual UPCs, not for text arguments.

Required fields:

1. **code:** UPC or EAN code. Should be 12 or 13 characters (leading zeros may be omitted). If this field cannot be validated, the entire row of data will be omitted;
2. **description:** The title of the item (up to 255 characters);
3. **language:** the ISO 639-1 2- character language code. The following languages are currently supported:

a. da: Danish	e. fr: French	i. no: Norwegian
b. de: German	f. fi: Finnish	j. pl: Polish
c. en: English	g. it: Italian	k. pt: Portuguese
d. es: Spanish	h. nl: Dutch	l. sv: Swedish

Optional fields

4. **uom:** Unit of measure: up to 128 characters that describe the item size such as "24 ounces", "750 ml", "12 pieces", or even just "1".
5. **website:** the web site where more information can be found about this item (512 characters; the root of the product web page will be used if that is supplied and this field is empty.)
6. **product_web_page:** web page where more information can be found about this item (512 characters);
7. **image:** location of an image of the item (512 characters);
8. **usage:** romance description of the item, usage instructions if available (no length limit);
9. **ingredients:** elements from which the item is made (no length limit);
10. **nutrition:** if nutritional data if appropriate (no length limit);
11. **brand:** product brand
12. **categories:** comma-separated list of categories or tags that should be associated with the item, listed in increasing order of importance (most important tag is on the right end of the string);

13. **manufacturer_name:** manufacturer name;

14. **manufacturer_address:** manufacturer location: street address;

15. **manufacturer_address2:** manufacturer location: suite, additional address;

16. **manufacturer_city:** manufacturer location: city;

17. **manufacturer_state:** manufacturer location: state (2 characters max);

18. **manufacturer_postal_code:** manufacturer location: postal code;

19. **manufacturer_country:** manufacturer country;

20. **manufacturer_phone:** manufacturer phone;

21. **manufacturer_contact:** manufacturer email or contact name.

Note to developers: Before activating your account for production, upsert requests of this type will not be processed. After activation, all upsert requests are provisional until reviewed and approved by the data curator.

A sample CSV upload file is available here: http://www.digit-eyes.com/data/demoDownloadList.csv

Signature calculator: http://www.digit-eyes.com/gtin/
Your keys: https://www.digit-eyes.com/cgi-bin/digiteyes.fcgi?action=myAccount

UPC/EAN UPSERT REQUEST RESPONSE

The result of a batch file upsert operation is an on-screen report that totals the number of records added.   Any requests that could not be processed will be shown with an error message explaining the error.

DELAYED RETRIEVAL

Growth of the database is stimulated, in part, by unsuccessful search activity.   When an item is not found, particularly if the action is initiated through the API, the hypersearch operation attempts to locate it.   The impact of this process is that items that are frequently or recently searched tend to have information added to the database shortly after a failing access.

This batch feature enables customers who are using the Single-Transaction API to receive notification and updates when the indexers locate information for an item number that had been previously requested and returned as "not found".

The request for delayed retrieval in CSV format is submitted as a POST operation to the endpoint:

```
http://digit-eyes.com/gtin/v2_0/csv/delayedRetrieval
?app_key=x&auth_key=x&language=x&field_names=x
```

The Universal Product Code (UPC) is a barcode symbology (i.e., a specific type of barcode) that is widely used in North America, in the United Kingdom, Australia, New Zealand, and other countries for tracking trade items in stores. Its most common form, the UPC-A, consists of 12 digits uniquely assigned to each trade item. Along with the related EAN barcode, the UPC is the barcode mainly used for scanning trade items at the point of sale, per GS1 specifications.[3]   UPC data structures are a component of GTINs (Global Trade Item Numbers). All of these data structures follow the global GS1 specification, which is based on international standards. Some retailers (clothing, furniture) do not use the GS1 System but use other bar code symbologies such as code 128 or code 39 and other article number systems. Other retailers use the EAN/UPC bar code symbology but without using a GTIN (for products brands sold at such retailers only)[4].

The UPC/EAN numbers are managed worldwide by GS1, an international not-for-profit association with member organizations in most countries[5].   The GS1 association issues the codes in the U.S.

The UPC/EAN does not indicate the product manufacturer (as many people believe) but, instead, identifies the party that has trade responsibility for the item. While these are often the same entity, this is not necessarily always the case.

UPC/EAN codes are composed of several sections, each of which is useful. Consider this code: 080480280024

0   080480   280024

- The above is a 13-digit version of a full 14-byte GTIN number 000 80408 28002 4;
- The first three digits tell you which country (in this case, **000** or **00** means the U.S.).   A complete list of code ranges and their assignments is in Appendix B of this document.
- The value **80480** indicates the party that has trade responsibility for the item. This number is within the address space of the U.S. numbers; the same number in another country will indicate a different business entity. The local authority for the country assigns the number. In this case, the number was issued by the U.S. GS1 and is assigned to *Bacardi USA, Inc. 2701 South Le Juene Road, Coral Gables FL, 33134, US*[6]

---

[3] EAN/UPC http://www.gs1us.org/resources/standards/ean-upc
[4] Universal Product Code, Wikipedia http://en.wikipedia.org/wiki/Universal_Product_Code
[5] Overview of GS1, GS1, http://www.gs1.org/
[6] For a useful source of such information: http://gepir.gs1.org/v32/xx/gtin.aspx?Lang=en-US

Appendix A: About UPC/EAN Codes

- **28002** is the inventory stocking number of the item.   Once issued, the business "owns" and administers its own address space. Bacardi ("**80480")** assigned the value ("**28002").**
- The last number, **4**, is a check digit for the other numbers. This number is calculated from the preceding values, as will be explained below.
- [Looking up the number in the database](), you'll find that it is Grey Goose Vodka.
- Cheers!

Calculating the check digit for a UPC is simple and should be routinely performed as part of scanning to ensure the scan occurred correctly.

- The check digit is calculated by taking the base number without the check digit: ("008048028002") and, if necessary, adding a leading zero to pad the number to an odd number of digits. Starting little-endian numbering of the digit positions at 0:
    - the digits in the even-numbered positions are added and the sum multiplied by 3;
    - The digits in the odd-numbered positions are added;
    - The sum of the two operations is divided by 10, and the remainder subtracted from the next highest multiple of 10;
    - If the answer is 10 (because the remainder is zero), the check digit is 0;
    - Given a 14-byte array named @parts with individually-addressable bytes numbered 0-13:
      checkDigit = (10 -(((((@parts[0] + @parts[2] + @parts[4] + @parts[6] + @parts[8] + @parts[10] + @parts[12]) *3) + (@parts[1] + @parts[3] + @parts[5] + @parts[7] + @parts[9] + @parts[11] + @parts[13])) % 10));
- In the case above:
    - Start with the base number: 080480280024
    - Remove the existing check digit (4): 08048028002
    - zero pad to the left to create a 13-byte root: 0008048028002
    - Perform the following calculation to determine whether the check digit is correct
      0008048028002
      The digits in the odd-numbered (green) positions are added together and the sum is multiplied by 3. The other numbers are added together also and added to the first sum. The result is divided by 10, and the remainder (indicated by the "%" sign in the following example) is subtracted from 10.   The calculation result is truncated to the right-most byte and becomes the check digit:

      checkDigit = (10 – ((((0+0 + 8 + 2 + 0 + 2) * 3) + (0 + 8 + 4 + 0 + 8 + 0)) % 10));
      (simplified to) checkDigit = (10 – (((12* 3) + 20) % 10));
      (simplified to) checkDigit = (10 –((36 + 20) %10));

Appendix A: About UPC/EAN Codes

> (simplified to) checkDigit = (10 – (46 % 10));
> (simplified to) checkDigit = (10 – 6);
> (simplified to) checkDigit = 4;

- o the 14-digit GTIN number, thus, is 000 80480 28002 4
- o the number that will probably appear on this item in the U.S. is 0 80480 28002 4

As a parenthetical note, some UPC/EAN codes are compressed and must be unpacked before use. These codes are eight characters in length (or less) and start with a zero to indicate that the number is zero-suppressed.

These are typically used for where the label has to fit in a tiny space and may consist of as few as five digits. An example would be a can of tuna that is curved too acutely to allow the code to be read when placed around the can and where the can is not tall enough to allow the use of a complete code when the code is placed vertically.

The digit in the last position indicates where zeros are to be interpolated into the number and how many zeros should be interpolated. The code, once expanded, is the same as any other UPC/EAN code in format. Compressed UPCs are uncommon, so the method of their expansion is not discussed here.

Some UPC/EAN codes have special meanings:

- Codes that start with "02" or "2" are for private use.   They are locally assigned codes that the stores used for conveying price and/or weight information. Typical examples would be random-weight packaged meat or produce. One-half of the code is used to indicate the product, and the other is used for the number of units (e.g., ounces, grams, etc.). At checkout, the POS system for the store calculates the price to be charged for the itemby looking up the product number, finding the current price per unit, and multiplying the price from the system by the number of units stated on the UPC label. These numbers are not stored in the Digit-Eyes database.

- Codes that start with "03" or "3" are OTC pharmaceuticals.   The leading number(s) are followed by the National Drug Code ("NDC") for the product, and the check digit is calculated in the usual manner.   The NDC codes are issued by the controlling health authority for the country. In the U.S., this is the FDA[7].   These numbers are available from Digit-Eyes.

---

[7] For more information: National Drug Code Directory, US Food & Drug Administration, https://www.fda.gov/drugs/drug-approvals-and-databases/national-drug-code-directory

Appendix A: About UPC/EAN Codes

- Codes that start with "4" are also locally assigned. These are sometimes called "System 4" or "Private UPC codes" and typically indicate a product that the store has on temporary special, where the store wants to override the public code or where the item has no public code at all. Examples might be a product manufactured to the business entity's specification or a product where the store does not want the public UPC number to be known. Anyone who needs to create a UPC number for an article that doesn't already have one can use any number like this and be sure that it won't conflict with a formally assigned number. However, there is no reason that another retailer cannot use the same number.   So, such a UPC can and does mean something different depending on who you ask.   Such numbers are available in the Digit-Eyes UPC database, but there are cases where the number indicates more than one product. Therefore, looking up a system 4 number in the Digit-Eyes database may result in multiple answers, each for a different issuing entity.

- Codes that start with "05" are for coupons. It is not uncommon that the balance of the code indicates the product to which the coupon applies, but this is by no means certain. Most coupons today in the U.S. do not use UPC but instead use the GS1 DataBar code[8]. These numbers are not stored in the Digit-Eyes database.

- All numbers which start with "978" or "979" are books, and the numbers derive from the old Bookland codes. These are called ISBNs ("International Standard Book Number") and issued by the International ISBN agency rather than GS1[9]. All numbers that start with "977" are serial publications such as magazines or newspapers. The ISSN number, less its check digit is a component of a barcode of this type[10]. Digit-Eyes has some of these numbers, but since such items are not generally of interest to our unsighted users, these numbers are only added at the specific request of our customers.

---

[8] GS1 Databar Barcodes, GS1, https://www.gs1.org/standards/barcodes/databar
[9] ISBN, International ISBN Agency, https://www.isbn-international.org/content/what-isbn
[10] About ISSN Numbers, Computalabel.com, http://www.computalabel.com/aboutissn.htm

Beginning	End	Country Name or Usage	Beginning	End	Country Name or Usage
0000000000000	0190000000000	United States	6110000000000	6110000000000	Morocco
0200000000000	0290000000000	Restricted distribution	6130000000000	6130000000000	Algeria
0300000000000	0390000000000	United States	6150000000000	6150000000000	Nigeria
0400000000000	0490000000000	Restricted distribution	6160000000000	6160000000000	Kenya
0500000000000	0590000000000	Coupons	6180000000000	6180000000000	Côte d'Ivoire
0600000000000	1390000000000	United States	6190000000000	6190000000000	Tunisia
2000000000000	2990000000000	Restricted distribution	6210000000000	6210000000000	Syria
3000000000000	3790000000000	France	6220000000000	6220000000000	Egypt
3800000000000	3800000000000	Bulgaria	6240000000000	6240000000000	Libya
3830000000000	3830000000000	Slovenia	6250000000000	6250000000000	Jordan
3850000000000	3850000000000	Croatia	6260000000000	6260000000000	Iran
3870000000000	3870000000000	Bosnia and Herzegovina	6270000000000	6270000000000	Kuwait
3890000000000	3890000000000	Montenegro	6280000000000	6280000000000	Saudi Arabia
4000000000000	4400000000000	Germany	6290000000000	6290000000000	UAE
4500000000000	4590000000000	Japan	6400000000000	6490000000000	Finland
4600000000000	4690000000000	Russian Federation	6900000000000	6950000000000	China
4700000000000	4700000000000	Kyrgyzstan	7000000000000	7090000000000	Norway
4710000000000	4710000000000	Taiwan	7290000000000	7290000000000	Israel
4740000000000	4740000000000	Estonia	7300000000000	7390000000000	Sweden
4750000000000	4750000000000	Latvia	7400000000000	7400000000000	Guatemala
4760000000000	4760000000000	Azerbaijan	7410000000000	7410000000000	El Salvador
4770000000000	4770000000000	Lithuania	7420000000000	7420000000000	Honduras
4780000000000	4780000000000	Uzbekistan	7430000000000	7430000000000	Nicaragua
4790000000000	4790000000000	Sri Lanka	7440000000000	7440000000000	Costa Rica
4800000000000	4800000000000	Philippines	7450000000000	7450000000000	Panama
4810000000000	4810000000000	Belarus	7460000000000	7460000000000	Dominican Republic
4820000000000	4820000000000	Ukraine	7500000000000	7500000000000	Mexico
4840000000000	4840000000000	Moldova	7540000000000	7550000000000	Canada
4850000000000	4850000000000	Armenia	7590000000000	7590000000000	Venezuela
4860000000000	4860000000000	Georgia	7600000000000	7690000000000	Switzerland
4870000000000	4870000000000	Kazakhstan	7700000000000	7700000000000	Colombia
4880000000000	4880000000000	Tajikistan	7730000000000	7730000000000	Uruguay
4890000000000	4890000000000	Hong Kong	7750000000000	7750000000000	Peru
4900000000000	4990000000000	Japan	7770000000000	7770000000000	Bolivia
5000000000000	5090000000000	United Kingdom	7790000000000	7790000000000	Argentina
5200000000000	5200000000000	Global Office	7800000000000	7800000000000	Chile
5280000000000	5280000000000	Lebanon	7840000000000	7840000000000	Paraguay
5290000000000	5290000000000	Cyprus	7860000000000	7860000000000	Ecuador
5300000000000	5300000000000	Albania	7890000000000	7900000000000	Brazil
5310000000000	5310000000000	Macedonia	8000000000000	8390000000000	Italy
5350000000000	5350000000000	Malta	8400000000000	8490000000000	Spain
5390000000000	5390000000000	Ireland	8500000000000	8500000000000	Cuba
5400000000000	5490000000000	Belgium	8580000000000	8580000000000	Slovakia
5600000000000	5600000000000	Portugal	8590000000000	8590000000000	Czech Rep.
5690000000000	5690000000000	Iceland	8600000000000	8600000000000	Serbia
5700000000000	5790000000000	Denmark	8650000000000	8650000000000	Mongolia
5900000000000	5900000000000	Poland	8670000000000	8670000000000	Korea, Peoples Rep
5940000000000	5940000000000	Romania	8680000000000	8690000000000	Turkey
5990000000000	5990000000000	Hungary	8700000000000	8790000000000	Netherlands
6000000000000	6010000000000	South Africa	8800000000000	8800000000000	KOREA, Rep. Of
6030000000000	6030000000000	Ghana	8840000000000	8840000000000	Cambodia
6080000000000	6080000000000	Bahrain	8850000000000	8850000000000	Thailand
6090000000000	6090000000000	Mauritius	8880000000000	8880000000000	Singapore

Appendix B: UPC / EAN Country Codes

Beginning	End	Country Name or Usage
8900000000000	8900000000000	India
8930000000000	8930000000000	Vietnam
8960000000000	8960000000000	Pakistan
8990000000000	8990000000000	Indonesia
9000000000000	9190000000000	Austria
9300000000000	9390000000000	Australia
9400000000000	9490000000000	New Zealand
9500000000000	9500000000000	Global Office
9550000000000	9550000000000	Malaysia
9580000000000	9580000000000	Macau
9770000000000	9770000000000	Serial publications (ISSN)
9780000000000	9790000000000	Bookland (ISBN)
9800000000000	9800000000000	Refund receipts
9810000000000	9820000000000	Common Currency Coupons
9900000000000	9990000000000	Coupons

APPENDIX C: ERROR MESSAGES

DATA RETRIEVAL OPERATION

DATA RETRIEVAL OPERATION: HTTP STATUS CODES

It is expected that the users of JSON will use the HTTP status codes to determine success or failure. The API uses the following HTTP status codes:

> 200 ok
> 400 UPC/EAN Code or Search Text invalid
> 401 Signature invalid
> 402 Requires funding
> 404 UPC/EAN not found

DATA RETRIEVAL OPERATION: RETURN CODES

Return codes are supplied to clarify the cause of failures and to provide information about the data returned:

> 0 = Successful transaction (http status code 200). Omitted in JSON response; use HTTP status code to determine success or failure.
> 1 = UPC information not on file, but GCP information is, returning GCP company data (http status code 200)
> 2 = UPC / GCP numbers not on file; information provided from private database; (http status code 200). Private databases are available by special arrangement only; this return code will not be found in most implementations.
> 3 = UPC not found, information provided from GCP + private database; (http status code 200). Private databases are available by special arrangement only; this return code will not be found in most implementations.
> 4 = (Used only in the V2 API) This is a locally-assigned UPC (a 13-digit code starting with "04" or a 12-digit code starting with "4".) Each store uses this number differently, so this code does not have a defined meaning. (http status code is 200)
> 5 = This is a coupon code (a 13-digit code starting with "05" or a 12-digit code starting with "5".) Each manufacturer codes their coupons differently, so there is no name or value defined for this code; (http status code 200). Most coupons today do not use UPC coding, but, instead, use GS1 DataBar (which permits more explicit information about coupons.)
> 6 = This is a locally-assigned UPC used for items of random weight such as meat, fish or product. (It is a 13-digit code starting with "02" or a 12-digit code starting with "2".) Random

Appendix C: Error Messages

weight UPCs are a way of price-marking an item. This item cost $price. (http status code 200)

7 = This is a private code (a 13-digit code starting with "000000" or a 12-digit code starting with "000000".) Each store uses these numbers differently, so the code does not have a defined meaning. (http status code 200)

999 = UPC/EAN code not found (http status code 404)

998 = UPC/EAN code not supplied (http status code 400)

997 = Search request not acceptable conflict with reserved terms (http status code 400)

996 = Application key not supplied or invalid (http status code 401)

995 = UPC/EAN code not valid (http status code 400)

994 = Unknown operation requested (http status code 400)

992 = Valid signature required (http status code 401)

666 = Account requires funding, data cannot be delivered (http status code 402)

DELETE OPERATION

DELETE OPERATION: HTTP STATUS CODES:

It is expected that the users of JSON will use the HTTP status codes to determine success or failure. The codes that may be returned are:

204 ok
304 Not modified
400 UPC/EAN Code invalid
401 Signature invalid
402 Requires funding
404 UPC/EAN not found

DELETE OPERATION RETURN: CODES

Return Codes are supplied to clarify the cause of failures and to provide information about the data returned:

0 = Successful transaction (http status code 204). Omitted in JSON response.

999 = UPC/EAN code not found (http status code 404)

998 = UPC/EAN code not supplied (http status code 400)

996 = Application key not supplied or invalid (http status code 401)

995 = UPC/EAN code not valid (http status code 400)

992 = Valid signature required (http status code 401)

Appendix C: Error Messages

991 = Deletions not allowed at this time (http status code 304)

990 = Language must be specified (http status code 400)

UPSERT OPERATION

UPSERT OPERATION: HTTP STATUS CODES:

It is expected that the users of JSON will use the HTTP status codes to determine success or failure. The codes that may be returned are:

201 ok
400 Code not supplied
401 Signature invalid
402 Requires funding
404 UPC/EAN not found

UPSERT OPERATION: RETURN CODES

Return Codes are supplied to clarify the cause of failures and to provide information about the data returned:

0 = Successful transaction (http status code 201). This element is omitted in JSON response; use HTTP status code to determine success or failure.

995 = UPC/EAN code not valid (http status code 400)

996= Application key not supplied or invalid (http status code 401)

992 = Valid signature required (http status code 401)

990 = Language must be specified (http status code 400)

DELAYED RETRIEVAL OPERATION

0 = Successful transaction (http status code 201). This element is omitted in JSON response; use HTTP status code to determine success or failure.

993 = Authorization key required (http status code 401)

666 = Account requires funding, data cannot be delivered (http status code 402)

665 = Request must be processed in batch, please contact support@digit-eyes.com (http status code 413)

APPENDIX D: SIGNING THE REQUEST

Each request for information must be signed. The "signature" is an encrypted version of the search argument: the seach_text field (including spaces), or the contents of the upc_code field. This value is SHA-1 hashed with your authorization key. Examples are shown below, and you can use the demo at http://digit-eyes.com/gtin/ to validate your signature creation results.

OBJECTIVE C EXAMPLE

To create a signature, we provide a category in objective C that takes a string and a key and returns the encoded value where UPCCode is the UPC or EAN code, and AuthKey is your authorization key (use the field "UPCCode" to contain the search text if this is a text-based search request):

```
#import "NSString+SHA1.h"
NSString *hashedValue = [UPCCode hashedValue:AuthKey];
```

Download: http://www.digit-eyes.com/specs/codeLibrary/NSString+SHA1.zip

C# EXAMPLE

Where Arg has the contents of the search_text or upc_code field, and AuthKey is your authorization key:

```
/// Product code that is to be used in the query is Arg
/// Hash that is to be used to create the siguature is AuthKey
private string GetDigitEyesVerificationCode(string Arg)
        {
            var hmac = new HMACSHA1(Encoding.UTF8.GetBytes(AuthKey));
            var m = hmac.ComputeHash(Encoding.UTF8.GetBytes(Arg));
            return Convert.ToBase64String(m);
        }
```

PHP EXAMPLE

Where $upc_code is the UPC or EAN code, and $auth_key is your authorization key (use the field "$upc_code" to contain the search text if this is a text-based search request)::

```
<?php
#
# Digit-Eyes Code to Generate and Output JSON
#
#****Dynamic UPC from form****
//$upc_code = $_GET["upc"];
#****Hard Code Test UPC****
$upc_code = '016000483668';
#****Substitute Your Auth Key****
$auth_key = 'xxxxxxxxxx';
#****Substitute Your App Key****
$app_key = 'yyyyyyyyyy';
#****Generates API Signature****
$signature = base64_encode(hash_hmac('sha1', $upc_code, $auth_key, $raw_output = true));

#***Dynamic UPC Production***
```

Appendix D: Signing the Request

```
//$url = 'https://digit-eyes.com/gtin/v2_0/?upc_code='.$_GET["upc"].'&app_key='. $app_key
.'&language=en&field_names=all&signature='. $signature .'';

#***Hard Code UPC Test***
$url = 'https://digit-eyes.com/gtin/v2_0/?upc_code='. $upc_code .'&app_key='. $app_key
.'&language=en&field_names=all&signature='. $signature .'';


$ch = curl_init($url);
curl_setopt($ch, CURLOPT_TIMEOUT, 5);
curl_setopt($ch, CURLOPT_CONNECTTIMEOUT, 5);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$data = curl_exec($ch);
curl_close($ch);
echo $data;
?>
```

PERL EXAMPLE

Where $arg has the contents of the search_text or upc_code field and $auth_key is your authorization key (use the field "UPCCode" to contain the search text if this is a text-based search request):

```
use Digest::HMAC_SHA1 qw(hmac_sha1 hmac_sha1_hex);
my $hmac = Digest::HMAC_SHA1->new($auth_key);
$hmac->add($arg);
my $signature = $hmac->b64digest;
my $paddingRequired = 4-(length($signature) % 4);
$signature .= substr("====", 0, $paddingRequired) if ($paddingRequired);
```

PYTHON EXAMPLE

Where arg has the contents of the search_text or upc_code field, and auth_key is your authorization key

```
import base64
import hashlib
import hmac

def make_auth_token(arg, auth_key):
  sha_hash = hmac.new(auth_key, arg, hashlib.sha1)
  return base64.b64encode(sha_hash.digest())
```

VBScript EXAMPLE

Where Arg has the contents of the search_text or upc_code field, and AuthKey is your authorization key:

```
Private Function GetDigitEyesVerificationCode(Arg As String) As String
Dim hmac = New HMACSHA1(Encoding.UTF8.GetBytes(AuthKey))
Dim m = hmac.ComputeHash(Encoding.UTF8.GetBytes(Arg))
Return Convert.ToBase64String(m)
End Function
```

Nutrition labeling data varies by country.   For products that have U.S. –format nutrition labeling, the information may be obtained in the "formattedNutrition" variable. The U.S. standard is explained in this online article from the FDA: How to Understand and Use the Nutrition Facts Label. Standards for displaying the label are explained in this FDA article: Guidance Documents: Labeling & Nutrition.

The formatted nutrition data is presented in JSON or XML format.   The data elements consist of a nutrient name (such as "biotin") and up to two values.   The value "qty" contains the quantity of the nutrient present, and the "dv" field contains the percent daily value for the nutrient:

- The % Daily Values are based on the daily value recommendations for key nutrients for a 2,000 calorie daily diet. As a general rule, trace nutrients such as vitamins and minerals have only the %DV field. However, in the case of products designed for infants and very young children, the trace nutrients may have the quantity field populated rather than the %DV field because the number of calories per day may vary widely by the child's age.
- Protein does not have a % DV. Per the FDA, this is because current scientific evidence indicates that protein intake in the U.S. is not a public health concern.
- Trans fats and sugars have no % DV.   Per the FDA, there is insufficient evidence to establish any recommended daily minimum of these substances.

JSON Field Name	XML Field Name	Nutrient	Qty	Unit	%DV
biotin	biotin	biotin (trace nutrient)	no*	mcg	yes*
calcium	calcium	calcium (trace nutrient)	no*	mg	yes*
calories	calories	calories	yes	calories	no
calories from fat	calories_from_fat	calories from fat	yes	calories	no
calories per serving	calories_per_serving	calories per serving	yes	calories	no
carbohydrates	carbohydrates	carbohydrates	yes	g	yes*
cholesterol	cholesterol	cholesterol	yes	mg	yes
choloride	choloride	choloride (trace nutrient)	no"	mcg	yes*
chromium]	chromium	chromium (trace nutrient)	no"	mcg	yes*
copper	copper	copper (trace nutrient)	no"	mg	yes
dietary fiber	dietary_fiber	dietary fiber	yes	g	yes
fiber	fiber	fiber (synonym for dietary fiber)	yes	g	yes
folic acid	folic_acid	folic acid (trace nutrient)	no*	mcg	yes*
insoluble fiber	insoluble_fiber	insoluble fiber	yse	g	no
iodine	iodine	iodine (trace nutrient)	no*	mcg	yes*
iron	iron	iron (trace nutrient)	no*	mg	yes*
magnesium	magnesium	magnesium (trace nutrient)	no*	mg	yes*

Appendix E: Using and Formatting Nutrition Data

JSON Field Name	XML Field Name	Nutrient	Qty	Unit	%DV
manganese	manganese	manganese (trace nutrient)	no*	mg	yes*
molybdenum	molybdenum	molybdenum (trace nutrient)	no"	mcg	yes*
monounsaturated fat	monounsaturated_fat	monounsaturated fat	yes	g	no
niacin	niacin	niacin (trace nutrient)	no*	mg	yes*
other carbohydrates	other_carbohydrates	other carbohydrates	yes	g	no
pantothenic acid	pantothenic_acid	pantothenic acid (trace nutrient)	no*	mg	yes*
phosphorus	phosphorus	phosphorus (trace nutrient)	no*	mg	yes*
polyunsaturated fat	polyunsaturated_fat	polyunsaturated fat	yes	g	no
potassium	potassium	potassium	yes	mg	yes
protein	protein	protein	yes	g	np
riboflavin	riboflavin	riboflavin (trace nutrient)	no*	mg	yes*
saturated fat	saturated_fat	saturated fat	yes	g	yes*
selenium	selenium	selenium (trace nutrient)	no*	mcg	yes*
serving size	serving_size	serving size	yes	text	no
servings per container	servings_per_container	servings per container	yes	text	no
sodium	sodium	sodium	yes	mg	yes
soluble fiber	soluble_fiber	soluble fiber	yes	g	no
sugar	sugar	sugar	yes	g	no
sugar alcohol	sugar_alcohol	sugar alcohol	yes	g	no
sugars	sugars	sugars	yes	g	no
thiamin	thiamin	thiamin (trace nutrient)	no*	mg	yes*
total calories	total_calories	total calories (synonym for calories)	yes	calories	no
total calories from fat	total_calories_from_fat	total calories from fat	yes	calories	no
total carbohydrates	total_carbohydrates	total carbohydrates	yes	g	yes
total fat	total_fat	total fat	yes	g	yes
trans fat	trans_fat	trans fat	yes	g	no
vitamin a	vitamin_a	vitamin a (trace nutrient)	no*	IU	yes*
vitamin b12	vitamin_b12	vitamin b12 (trace nutrient)	no*	mcg	yes*
vitamin b6	vitamin_b6	vitamin b6 (trace nutrient)	no*	mg	yes*
vitamin c	vitamin_c	vitamin c (trace nutrient)	no*	mb	yes*
vitamin d	vitamin_d	vitamin d (trace nutrient)	no*	IU	yes*
vitamin e	vitamin_e	vitamin e (trace nutrient) "	no*	IU	yes*
vitamin k	vitamin_k	vitamin k (trace nutrient)	no*	mcg	yes*
vitamin k1	vitamin_k1	vitamin k1 (trace nutrient)	no*	mcg	yes*
zinc	zinc	zinc (trace nutrient)	no*	mg	yes*

Two additional fields are also available in the formattedNutrition segment: Allergy warnings and Additional warnings (allergy_warnings and additional_warnings for the XML version).