Product specifications

Thank you for buying a GeoPostcodes product, we appreciate your business. Hereafter you will find some technical information on GeoPostcodes database and how to import the data in your application.

USE OF GEOPOSTCODES DATABASE IS SUBJECT TO OUR TERMS AND CONDITIONS. YOU MAY NOT USE THE DATABASE IN ANY MANNER UNLESS YOU ACCEPT THE TERMS AND CONDITIONS OF THE LICENSE. IF YOU DO NOT ACCEPT ALL OF THE TERMS AND CONDITIONS OF THE LICENSE, DO NOT USE THE DATABASE IN ANY MANNER.

Database fields

All our database files are based on the same structure, however and depending on the country, some fields may remain void.

Field name	Data type	Description
Country	Char(2)	ISO 3166-1 country code
Language	Char(2)	ISO 639-1 language code
ID	Long	Record counter
ISO2	Char(6)	ISO 3166-2 region code
Region1	Char(60)	Administrative region level 1
Region2	Char(60)	Administrative region level 2
Region3	Char(60)	Administrative region level 3
Region4	Char(60)	Administrative region level 4
ZIP	Char(10)	ZIP/Postal code
City	Char(60)	Locality name
Area1	Char(80)	Additional information (street, block, district, ..)
Area2	Char(80)	Additional information (street, block, district, ..)
Lat	Double	WGS 84 Geographic latitude in decimal format
Lng	Double	WGS 84 Geographic longitude in decimal format
TZ	Char(30)	Time zone (Olson code)
UTC	Char(10)	Local time
DST	Char(1)	Daylight saving time

Available Files Formats

File extension	Encoding	Description
.csv	UTF-8	Comma separated values
.xls	UTF-8	Microsoft Excel - limited to 65536 row (*)
.asc	ASCII	Unformatted ASCII Text
.xml	UTF-8	XML structure
.sql	UTF-8	MySQL compatible
.dat	UTF-16	SQL Server UTF-16 compatible format

Microsoft Excel (.xls)

The .xls format is compatible with all Excel versions but is limited to 65536 rows and may not be working for non-latin chars. If you have Excel 2007 or 2010 you can use the .csv file to get up to 1 million rows.

(*) If there is more than 65536 rows, you can import the .csv file manually as follow :

- In the File|Open menu, select your text file (GeoPC_xx.csv)
- In import wizard, select "Delimited" data type and "65001 : Unicode (UTF-8)" encoding
- Click on Next, select Semicolon as delimiter and " as text qualifier and click on the Finish button

Maximum number of rows:

- Excel 2003 and earlier : 65536 rows (.xls)
- Excel 2007 : 1048576 rows (.csv)

Microsoft Access (.mdb)

We do not provide native Access databases, however it is really easy to import a GeoPostcodes database in Access, here is how to do :

- Open or create a new Access database
- Go to File|Get External Data|Import in the menu, select your file (GeoPC_xx.csv)
- Select Delimited, click on Advanced and select Unicode (UTF-8) as Code Page
- Click Next, select Semicolon as delimiter and " as Text qualifier
- Tick First row contains field names and click on Finish

Text delimited (.csv)

This is the most common format, most applications can import text delimited files. Fields are separated by a semicolon, text are enclosed by double quotes ("). UTF-8 encoding.

Unformatted ASCII Text (.asc)

The format of this file is the same than above .csv but without accented or special characters used in foreign languages. You will find more information on the ASCII format on our website : http://www.geopostcodes.com/encoding

XML encoding (.xml)

Another common format, can be used for website development.

MySQL database (.sql)

The .sql file contains both queries and data to import your GeoPostcodes file in a MySQL database.

phpMyAdmin :
Click on the Import tab, select your GeoPC_xx.sql file and utf8 as Character set.

Command line :

```
mysql> source GeoPC_xx.sql
```

If you prefer create the table and import the data manually, here is the query :

```
DROP TABLE IF EXISTS GeoPC;

CREATE TABLE GeoPC (
  Country varchar(2) NOT NULL,
  Language varchar(2) NOT NULL,
  ID bigint(20) NOT NULL,
  RegISO2 varchar(6) NOT NULL,
  Region1 varchar(60) NOT NULL,
  Region2 varchar(60) NOT NULL,
  Region3 varchar(60) NOT NULL,
  Region4 varchar(60) NOT NULL,
  ZIP varchar(10) NOT NULL,
  City varchar(60) NOT NULL,
  Area1 varchar(80) NOT NULL,
  Area2 varchar(80) NOT NULL,
  Lat double NOT NULL,
  Lng double NOT NULL,
  TZ varchar(30) NOT NULL,
  UTC varchar(10) NOT NULL,
  DST varchar(1) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

LOAD DATA INFILE '/path/GeoPC_xx.csv' INTO TABLE yourdb.GeoPC FIELDS
TERMINATED BY ';' ENCLOSED BY '"' ESCAPED BY '\\' LINES TERMINATED BY
'\r\n';
```

Microsoft SQL Server (.dat)

To import GeoPostcodes in Microsoft SQL Server, you need to use the GeoPC_xx.dat file with the BULK INSERT command :

```
CREATE TABLE [GeoPC] (
   [Country] [nvarchar](2) NULL,
   [Language] [nvarchar](2) NULL,
   [ID] [bigint] NULL,
   [RegISO2] [nvarchar](6) NULL,
   [Region1] [nvarchar](60) NULL,
   [Region2] [nvarchar](60) NULL,
   [Region3] [nvarchar](60) NULL,
   [Region4] [nvarchar](60) NULL,
   [ZIP] [nvarchar](10) NULL,
   [City] [nvarchar](60) NULL,
   [Area1] [nvarchar](80) NULL,
   [Area2] [nvarchar](80) NULL,
   [Lat] [float] NULL,
   [Lng] [float] NULL,
   [TZ] [nvarchar](30) NULL,
   [UTC] [nvarchar](10) NULL,
   [DST] [nvarchar](1) NULL
)
GO

BULK INSERT GeoPC FROM 'Path\GeoPC_xx.dat' WITH (FIELDTERMINATOR = ';',
DATAFILETYPE = 'widechar')
```

Languages and characters encoding

All GeoPostcodes databases contain by default the names of the localities and regions written in the **local language** (they can be multiple in some countries). These files are encoded in **UTF-8** format (or **UTF-16** for the SQL Server files).

For ease of use of our databases in English, some measures have been taken to translate them from the various foreign languages :

Non-Latin languages

For languages using a non-Latin alphabet (such as Chinese, Japanese, Russian, etc), an **English version** is available in addition to the local version. These versions have been **translated** as closely as possible from the original text. However, where it is not possible to make an accurate translation that is reliable or the translation does not exist, the closest alternative, a **transliteration**, is provided.

Non formatted version in ASCII

All databases have a non formatted version using **US-ASCII** encoding.

This file (named G*eoPC_xx.asc*) is a **transliteration** according to the tables defined below, it uses the 26 letters from the Latin alphabet and does not have any accented characters. It is accepted that this is a basic match which may differ considerably from a translation.

Translations, transcriptions, transliterations

Several methods can be used to produce a version in the English language of a name written in a foreign language, the most common one being the *translation*. In this case, we assume that there is already an English name commonly used matching to the locality, for example **Lisboa** in Portugal is called **Lisbon** in English, then we talk about an **exonym**. An exonyms table is available free of charge on our website.

Since there is no translation for most of the towns and villages in the world, it is advisable to do a *transcription* or a *transliteration* in English. The **transcription** is a replacement of each character or characters group by an equivalent sound in English, it is therefore a phonetic translation. The **transliteration** (also called **romanization**) is a substitution of each character or characters group by an other equivalent according to a matching table, which does not take into consideration pronunciation.

Example :

Local Russian name : **Санкт-Петербург**

English translation : **Saint Petersburg**

Transcription : **Sankt Petersburg**

Transliteration : **Sankt-Peterburg**

US-ASCII Transliteration tables can be found on page **http://www.geopostcodes.com/encoding**