

# database\_schemas

March 19, 2019

## 1 Database Schemas

Material for the hackathon ENSAE / Red Cross / 2015. Schema of the databases.

```
In [1]: %matplotlib inline
import matplotlib.pyplot as plt
plt.style.use('ggplot')
```

```
In [2]: from jyquickhelper import add_notebook_menu
add_notebook_menu()
```

```
Out[2]: <IPython.core.display.HTML object>
```

### 1.1 Connection to the cluster

You can set your credentials and retrieve them with module [keyring](#).

```
In [3]: # don't leave these line in the notebook.
# keyring.set_password("CRCREDENTIALS", "blobstorage", "...")
# keyring.set_password("CRCREDENTIALS", "password", "...")
```

```
In [4]: import keyring
hackathon = {}
hackathon["blob_storage"] = keyring.get_password("CRCREDENTIALS", "blobstorage")
hackathon["password"] = keyring.get_password("CRCREDENTIALS", "password")
```

```
In [5]: blobstorage = hackathon["blob_storage"]
blobpassword = hackathon["password"]
```

```
In [6]: %load_ext pyensae
```

```
In [7]: cl, bs = %blob_open
cl, bs
```

```
Out[7]: (<pyensae.remote.azure_connection.AzureClient at 0x9272828>,
<azure.storage.blob.blobservice.BlobService at 0x9272860>)
```

### 1.2 Variables and tables

```
In [8]: %blob_down croix-rouge/data/ITMMASTER.schema.txt ITMMASTER.schema.txt
%blob_down croix-rouge/data/SINVOICE.schema.txt SINVOICE.schema.txt
%blob_down croix-rouge/data/SINVOICEV.schema.txt SINVOICEV.schema.txt
```

```
Out[8]: 'SINVOICEV.schema.txt'
```

## 1.2.1 ITMMASTER

```
In [9]: %blob_ls croix-rouge
```

```
Out[9]:
```

|    | name                           | last_modified                 | \ |
|----|--------------------------------|-------------------------------|---|
| 0  | build/SINVOICE_M.csv           | Wed, 18 Nov 2015 18:56:27 GMT |   |
| 1  | build/Test_CRFFOR.GACCTMPD.csv | Sun, 22 Nov 2015 21:53:38 GMT |   |
| 2  | data/ITMMASTER.schema.txt      | Mon, 16 Nov 2015 23:00:34 GMT |   |
| 3  | data/ITMMASTER.txt             | Mon, 09 Nov 2015 21:41:00 GMT |   |
| 4  | data/SINVOICE.schema.txt       | Mon, 16 Nov 2015 23:00:35 GMT |   |
| 5  | data/SINVOICE.txt              | Mon, 09 Nov 2015 21:42:32 GMT |   |
| 6  | data/SINVOICEV.schema.txt      | Mon, 16 Nov 2015 23:00:35 GMT |   |
| 7  | data/SINVOICEV.txt             | Mon, 09 Nov 2015 21:44:08 GMT |   |
| 8  | data/enseignes_france.csv      | Mon, 09 Nov 2015 21:40:54 GMT |   |
| 9  | data/stojou.csv                | Mon, 09 Nov 2015 21:55:23 GMT |   |
| 10 | readme.txt                     | Sun, 22 Nov 2015 18:00:34 GMT |   |

  

|    | content_type             | content_length | blob_type |
|----|--------------------------|----------------|-----------|
| 0  | application/octet-stream | 533771533      | BlockBlob |
| 1  | application/octet-stream | 822231942      | BlockBlob |
| 2  | application/octet-stream | 5658           | BlockBlob |
| 3  | application/octet-stream | 103096479      | BlockBlob |
| 4  | application/octet-stream | 10252          | BlockBlob |
| 5  | application/octet-stream | 1362433753     | BlockBlob |
| 6  | application/octet-stream | 7999           | BlockBlob |
| 7  | application/octet-stream | 1252461865     | BlockBlob |
| 8  | application/octet-stream | 6303836        | BlockBlob |
| 9  | application/octet-stream | 8821375868     | BlockBlob |
| 10 | application/octet-stream | 45             | BlockBlob |

```
In [10]: df = %blob_head croix-rouge/data/ITMMASTER.schema.txt
df.head()
```

```
Out[10]:
```

|   | Intitulé long       | Options | Table liée | Expression de lien       | \       |
|---|---------------------|---------|------------|--------------------------|---------|
| 0 | Article             | NaN     | ITMMASTER  |                          | NaN     |
| 1 | Famille statistique | NaN     | ATABDIV    | indice+20;TSICOD(indice) |         |
| 2 | Catégorie article   | NaN     | ITMCATEG   |                          | ;TCLCOD |
| 3 | Clé recherche       | NaN     | NaN        |                          | NaN     |
| 4 | Code axe            | NaN     | GDIE       |                          | NaN     |

  

|   | Copie législation | Annulation  | Vérification | Obligatoire | RAZ | Mot-clé d'aide |
|---|-------------------|-------------|--------------|-------------|-----|----------------|
| 0 | NaN               | Suppression | Oui          | Non         | Non | NaN            |
| 1 | NaN               | Bloquant    | Oui          | Non         | Non | NaN            |
| 2 | NaN               | Bloquant    | Oui          | Oui         | Non | NaN            |
| 3 | NaN               | NaN         | NaN          | Non         | Non | NaN            |
| 4 | NaN               | Bloquant    | Oui          | Non         | Non | NaN            |

```
In [11]: from pyquickhelper.pandashelper import df2rst
with open("sch_itmmaster.txt", "w", encoding="utf8") as f:
    dfi = df.reset_index(drop=False)
    dfi["index"] = dfi["index"]+1
    f.write(df2rst(dfi.fillna("")))


```

## 1.2.2 SINVOICE

```
In [12]: from ensae_projects.datainc.croix_rouge import get_meaning
meaning = get_meaning("invoice")
```

```
meaning.head()
```

```
Out[12]:
```

|   | ç Champ | Description                                       |
|---|---------|---|
| 0 | NUM     | Numéro de pièce                                   |
| 1 | BPR     | Tiers (identifiant du beneficiaire)               |
| 2 | FCY     | Site (code U2A)                                   |
| 3 | ACCDAT  | Date comptable                                    |
| 4 | ACCNUM  | Numéro interne de la transaction (ID unique po... |

```
In [13]: meaning.columns
```

```
Out[13]: Index(['ç Champ', 'Description'], dtype='object')
```

```
In [14]: df = %blob_head croix-rouge/data/SINVOICE.schema.txt  
df.head()
```

```
Out[14]:
```

|   | Zone   | Typ | Menu | Long | Act | Dim | Intitulé normal   | Intitulé abrégé | \ |
|---|--------|-----|------|------|-----|-----|-------------------|-----------------|---|
| 0 | SIVTYP | TSV | 0    | NaN  | NaN | 1   | Type facture      | Type fac        |   |
| 1 | INVTYP | M   | 645  | 15   | NaN | 1   | Catégorie facture | Catég fac       |   |
| 2 | NUM    | VCR | 0    | NaN  | NaN | 1   | Numéro de pièce   | Numéro          |   |
| 3 | ORIMOD | M   | 14   | 10   | NaN | 1   | Module origine    | Module          |   |
| 4 | BPR    | BPR | 0    | NaN  | NaN | 1   | Tiers             | Tiers           |   |

  

|   | Intitulé long     | Options | Table liée | Expression de lien | Copie législation | \ |
|---|-------------------|---------|------------|--------------------|-------------------|---|
| 0 | Type facture      | NaN     | TABSIVTYP  | NaN                | NaN               |   |
| 1 | Catégorie facture | NaN     | NaN        | NaN                | NaN               |   |
| 2 | Numéro de pièce   | NaN     | NaN        | NaN                | NaN               |   |
| 3 | Module origine    | NaN     | NaN        | NaN                | NaN               |   |
| 4 | Tiers             | NaN     | BPARTNER   | NaN                | NaN               |   |

  

|   | Annulation | Vérification | Obligatoire | RAZ | Mot-clé d'aide |
|---|------------|--------------|-------------|-----|----------------|
| 0 | Bloquant   | Oui          | Non         | Non | NaN            |
| 1 | NaN        | NaN          | Non         | Non | NaN            |
| 2 | NaN        | NaN          | Non         | Non | NaN            |
| 3 | NaN        | NaN          | Non         | Non | NaN            |
| 4 | Bloquant   | Oui          | Oui         | Non | NaN            |

```
In [15]: merge = df.merge(meaning, right_on=meaning.columns[0], left_on="Zone", how="outer")  
print(merge.shape, df.shape, meaning.shape)  
merge = merge[~merge["Zone"].isnull()]  
print(merge.shape)  
merge.head(n=2)
```

```
(146, 20) (124, 18) (80, 2)
```

```
(124, 20)
```

```
Out[15]:
```

|   | Zone   | Typ | Menu | Long | Act | Dim | Intitulé normal   | Intitulé abrégé | \ |
|---|--------|-----|------|------|-----|-----|-------------------|-----------------|---|
| 0 | SIVTYP | TSV | 0    | NaN  | NaN | 1   | Type facture      | Type fac        |   |
| 1 | INVTYP | M   | 645  | 15   | NaN | 1   | Catégorie facture | Catég fac       |   |

  

|   | Intitulé long     | Options | Table liée | Expression de lien | Copie législation | \ |
|---|-------------------|---------|------------|--------------------|-------------------|---|
| 0 | Type facture      | NaN     | TABSIVTYP  | NaN                | NaN               |   |
| 1 | Catégorie facture | NaN     | NaN        | NaN                | NaN               |   |

  

|  | Annulation | Vérification | Obligatoire | RAZ | Mot-clé d'aide | ç Champ | \ |
|--|------------|--------------|-------------|-----|----------------|---------|---|
|--|------------|--------------|-------------|-----|----------------|---------|---|

```

0 Bloquant Oui Non Non NaN SIVTYP
1 NaN NaN Non Non NaN INVTYP

```

```

Description
0 Type facture vente utilisateur (toujours egal ...
1 Catégorie facture (toujours egal a 1 ~ inutile)

```

```

In [16]: from pyquickhelper.pandashelper import df2rst
with open("sch_invoice.txt", "w", encoding="utf8") as f:
    dfi = merge.reset_index(drop=False)
    dfi["index"] = dfi["index"]+1
    f.write(df2rst(dfi.fillna("")))

```

### 1.2.3 INVOICE\_V

```

In [17]: df = %blob_head croix-rouge/data/SINVOICEV.schema.txt
df.head(n=2)

```

```

Out[17]: Zone Typ Menu Long Act Dim Intitulé normal Intitulé abrégé \
0 NUM VCR 0 NaN NaN 1 No facture Facture
1 CPY CPY 0 NaN NaN 1 Société Société

Intitulé long Options Table liée Expression de lien Copie législation \
0 No facture NaN SINVOICE NaN NaN
1 Société NaN COMPANY NaN NaN

Annulation Vérification Obligatoire RAZ Mot-clé d'aide
0 Autre Oui Non Non NaN
1 Bloquant Oui Oui Non NaN

```

```

In [18]: mergev = df.merge(meaning, right_on=meaning.columns[0], left_on="Zone", how="outer")
print(mergev.shape, df.shape, meaning.shape)
mergev = mergev[~mergev["Zone"].isnull()]
print(mergev.shape)
mergev.head(n=2)

```

```

(140, 20) (91, 18) (80, 2)
(91, 20)

```

```

Out[18]: Zone Typ Menu Long Act Dim Intitulé normal Intitulé abrégé \
0 NUM VCR 0 NaN NaN 1 No facture Facture
1 CPY CPY 0 NaN NaN 1 Société Société

Intitulé long Options Table liée Expression de lien Copie législation \
0 No facture NaN SINVOICE NaN NaN
1 Société NaN COMPANY NaN NaN

Annulation Vérification Obligatoire RAZ Mot-clé d'aide ; Champ \
0 Autre Oui Non Non NaN NUM
1 Bloquant Oui Oui Non NaN CPY

Description
0 Numéro de pièce
1 Société (toujours egal a CRF)

```

```
In [19]: from pyquickhelper.pandashelper import df2rst
with open("sch_invoice_v.txt", "w", encoding="utf8") as f:
    dfi = mergev.reset_index(drop=False)
    dfi["index"] = dfi["index"]+1
    f.write(df2rst(dfi.fillna("")))

```

Differences between INVOICE\_V and INVOICE

```
In [20]: merge["table"] = "INVOICE"
mergev["table"] = "INVOICE_V"
diff = merge[["Zone", "table"]].merge(mergev[["Zone", "table"]], how="outer", on="Zone", suffixes=("_", "_V"))
diff["COMMON"] = True
diff.loc[diff["table"].isnull() | diff["table_V"].isnull(), "COMMON"] = False
diff = diff.sort_values(["COMMON", "Zone"])
diff.shape

```

Out[20]: (201, 4)

```
In [21]: from pyquickhelper.pandashelper import df2rst
with open("sch_invoice_diff.txt", "w", encoding="utf8") as f:
    dfi = diff.reset_index(drop=False)
    dfi["index"] = dfi["index"]+1
    f.write(df2rst(dfi.fillna("")))

```

## 1.2.4 stojou

```
In [22]: df = cl.df_head(bs, "croix-rouge", "data/stojou.csv", as_df=False, stop_at=2000)
res = df.split("\r")[0]
res

```

Out[22]: 'STOFCY\_0,UPDCOD\_0,ITMREF\_0,IPTDAT\_0,MVTSEQ\_0,MVTIND\_0,CSTDAT\_0,CSTTIM\_0,CSTCOU\_0,OWNER\_0,LOT\_0'

```
In [23]: import pandas
stojou = pandas.DataFrame(data={ "Zone":[ _.replace("_0", "") for _ in res.split(",")]})
stojou.to_csv("stojou.schema.txt")
stojou["table_SJ"] = "stojou"
stojou.head()

```

```
Out[23]:      Zone table_SJ
0  STOFCY  stojou
1  UPDCOD  stojou
2  ITMREF  stojou
3  IPTDAT  stojou
4  MVTSEQ  stojou

```

```
In [24]: diff.head()

```

```
Out[24]:      Zone  table table_V COMMON
10  ACCDAT  INVOICE      NaN  False
11  ACCNUM  INVOICE      NaN  False
44  AMTATI  INVOICE      NaN  False
47  AMTATIL  INVOICE      NaN  False
45  AMTNOT  INVOICE      NaN  False

```

```
In [25]: final = diff[["Zone", "table", "table_V"]].merge(stojou, how="outer", on="Zone", suffixes=("_", "_V"))
final = final.sort_values(["table_SJ", "Zone"])
final.shape

```

```
Out[25]: (313, 4)
```

```
In [26]: from pyquickhelper.pandashelper import df2rst
with open("sch_stojou.txt", "w", encoding="utf8") as f:
    dfi = final.reset_index(drop=False)
    dfi["index"] = dfi["index"]+1
    f.write(df2rst(dfi.fillna("")))
```

### 1.3 Overview of the data

```
In [27]: df = %blob_head croix-rouge/build/SINVOICE_M.csv --sep=,
df.head()
```

```
Out[27]:
```

|   | NUM             | STARPT | CREUSR | CREDAT   | BPR       | FCY   | ACCDAT   | \ |
|---|-----------------|--------|--------|----------|-----------|-------|----------|---|
| 0 | DIS100101000001 | 2      | REYC   | 25/11/10 | 100000249 | U8201 | 01/01/10 |   |
| 1 | DIS100101000002 | 2      | REYC   | 25/11/10 | 100000174 | U8201 | 01/01/10 |   |
| 2 | DIS100101000003 | 2      | REYC   | 25/11/10 | 100000105 | U8201 | 01/01/10 |   |
| 3 | DIS100101000004 | 2      | REYC   | 25/11/10 | 100000244 | U8201 | 01/01/10 |   |
| 4 | DIS100101000005 | 2      | REYC   | 25/11/10 | 100000390 | U8201 | 01/01/10 |   |

  

|   | ACCNUM | BPRPAY    | VAC | ... | EECSCH | EECSCHR | EECLOC | DSPTOTQTY | \ |
|---|--------|-----------|-----|-----|--------|---------|--------|-----------|---|
| 0 | 705    | 100000249 |     | ... | FR21   | FR26    | 0      | 19        |   |
| 1 | 707    | 100000174 |     | ... | FR21   | FR26    | 0      | 24        |   |
| 2 | 709    | 100000105 |     | ... | FR21   | FR26    | 0      | 80        |   |
| 3 | 711    | 100000244 |     | ... | FR21   | FR26    | 0      | 52        |   |
| 4 | 713    | 100000390 |     | ... | FR21   | FR26    | 0      | 22        |   |

  

|   | DSPTOTWEI | DSPTOTVOL | DSPWEU | DSPVOU | YCODTR | ZBATCHFLG |
|---|-----------|-----------|--------|--------|--------|-----------|
| 0 | 10,353    | 0,25      | KG     | L      | DIS    | 2         |
| 1 | 16,98     | 0,62      | KG     | L      | DIS    | 2         |
| 2 | 58,919    | 1,75      | KG     | L      | DIS    | 2         |
| 3 | 34,426    | 1,87      | KG     | L      | DIS    | 2         |
| 4 | 10,356    | 0,25      | KG     | L      | DIS    | 2         |

[5 rows x 80 columns]

```
In [28]: df.columns
```

```
Out[28]: Index(['NUM', 'STARPT', 'CREUSR', 'CREDAT', 'BPR', 'FCY', 'ACCDAT', 'ACCNUM',
'BPRPAY', 'VAC', 'STA', 'AMTATI', 'VATDAT', 'NBRTAX', 'TAX', 'BPAINV',
'YRESTEAV', 'YNBR', 'YTYPOFAM', 'YCSP', 'YSITPAR', 'YSITTRA', 'YSITLOG',
'YSSITLOG', 'YCRY', 'YT1', 'YT2', 'YT3', 'YT4', 'YT5', 'YT6', 'YT7',
'YT8', 'YSEXMAL', 'YSEXFEM', 'YSOLDINITIAL', 'YSOLDEACTUAL',
'YMTSANSREDUC', 'YMTAVECREDUC', 'YMTREDUCTION', 'YMTVALMER', 'YORIG1',
'YORIG2', 'YORIG3', 'YORIG4', 'YORIG5', 'YORIG6', 'YORIG7', 'YFREQPASS',
'YCSPCHEF', 'YTYPPROV', 'YREVREF', 'YQUOTFAM', 'YCNTTYP', 'CPY',
'BPAADD', 'CNINAM', 'BPDCRYNAM', 'SIVTYP', 'INVTYP', 'CUR', 'STOMVTFLG',
'PRITYP', 'LAN', 'ORIFCY', 'LINNBR', 'INVDTAAMT', 'TRSCOD', 'ENTCOD',
'EECNAT', 'EECSCH', 'EECSCHR', 'EECLOC', 'DSPTOTQTY', 'DSPTOTWEI',
'DSPTOTVOL', 'DSPWEU', 'DSPVOU', 'YCODTR', 'ZBATCHFLG'],
dtype='object')
```

### 1.4 Header for the dataframe

Just to check that suffix\_0 is implicit in the documentation.

```
In [29]: %blob_ls croix-rouge
```

```
Out[29]:
```

|    | name                           | last_modified                 | \ |
|----|--------------------------------|-------------------------------|---|
| 0  | build/SINVOICE_M.csv           | Wed, 18 Nov 2015 18:56:27 GMT |   |
| 1  | build/Test_CRFFOR.GACCTMPD.csv | Sun, 22 Nov 2015 21:53:38 GMT |   |
| 2  | data/ITMMASTER.schema.txt      | Mon, 16 Nov 2015 23:00:34 GMT |   |
| 3  | data/ITMMASTER.txt             | Mon, 09 Nov 2015 21:41:00 GMT |   |
| 4  | data/SINVOICE.schema.txt       | Mon, 16 Nov 2015 23:00:35 GMT |   |
| 5  | data/SINVOICE.txt              | Mon, 09 Nov 2015 21:42:32 GMT |   |
| 6  | data/SINVOICEV.schema.txt      | Mon, 16 Nov 2015 23:00:35 GMT |   |
| 7  | data/SINVOICEV.txt             | Mon, 09 Nov 2015 21:44:08 GMT |   |
| 8  | data/enseignes_france.csv      | Mon, 09 Nov 2015 21:40:54 GMT |   |
| 9  | data/stojou.csv                | Mon, 09 Nov 2015 21:55:23 GMT |   |
| 10 | readme.txt                     | Sun, 22 Nov 2015 18:00:34 GMT |   |

  

|    | content_type             | content_length | blob_type |
|----|--------------------------|----------------|-----------|
| 0  | application/octet-stream | 533771533      | BlockBlob |
| 1  | application/octet-stream | 822231942      | BlockBlob |
| 2  | application/octet-stream | 5658           | BlockBlob |
| 3  | application/octet-stream | 103096479      | BlockBlob |
| 4  | application/octet-stream | 10252          | BlockBlob |
| 5  | application/octet-stream | 1362433753     | BlockBlob |
| 6  | application/octet-stream | 7999           | BlockBlob |
| 7  | application/octet-stream | 1252461865     | BlockBlob |
| 8  | application/octet-stream | 6303836        | BlockBlob |
| 9  | application/octet-stream | 8821375868     | BlockBlob |
| 10 | application/octet-stream | 45             | BlockBlob |

```
In [30]: df = cl.df_head(bs, "croix-rouge", "data/stojou.csv", as_df=False, stop_at=2000)
df.split("\r")[0]
```

```
Out[30]: 'STOFCY_0,UPDCOD_0,ITMREF_0,IPTDAT_0,MVTSEQ_0,MVTIND_0,CSTDAT_0,CSTTIM_0,CSTCOU_0,OWNER_0,LOT_0'
```

```
In [31]: df = cl.df_head(bs, "croix-rouge", "data/SINVOICEV.txt", as_df=False, stop_at=2000)
df.split("\r")[0]
```

```
Out[31]: '"NUM_0"\t"CPY_0"\t"SALFCY_0"\t"STOFCY_0"\t"BPCINV_0"\t"BPCORD_0"\t"BPCGRU_0"\t"BPAADD_0"\t"BP'
```

```
In [32]: df = cl.df_head(bs, "croix-rouge", "data/SINVOICE.txt", as_df=False, stop_at=2000)
df.split("\r")[0]
```

```
Out[32]: '"SIVTYP_0"\t"INVTYP_0"\t"NUM_0"\t"ORIMOD_0"\t"BPR_0"\t"BPRSAC_0"\t"CPY_0"\t"FCY_0"\t"GTE_0"\t'
```

```
In [33]: df = cl.df_head(bs, "croix-rouge", "data/ITMMASTER.txt", as_df=False, stop_at=2000)
df.split("\r")[0]
```

```
Out[33]: '"3166290200616","ALI","FL","LGM","HRB","HRB","PRSEC"," "," "," "," "," "," "," "," "," "," "
```

```
In [34]: df = cl.df_head(bs, "croix-rouge", "data/enseignes_france.csv", as_df=True, stop_at=None)
df.head()
```

```
Out[34]:
```

|   | ADRESSE_POSTALE_TEXT     | TEXT_1                                 | \ |
|---|--------------------------|--|---|
| 0 | Le Grand Rivolet         | Simply Market MONTCEAUX GUEREINS       |   |
| 1 | 4 Rue Du Marche,         | SPAR Montmerle Sur Saone               |   |
| 2 | Le Pré de la Cloche      | INTERMARCHE SUPER Belleville-sur-Saone |   |
| 3 | 47 Rue De La Republique, | PETIT CASINO Belleville Sur Saone      |   |

```

4 AVENUE DE VERDUN Carrefour Market BELLEVILLE SUR SAONE

IMAGE_1/_title SITE_INTERNET_LINK \
0 Simply-Market http://www.simplymarket.fr/montceaux-guereins/
1 Spar http://annuaire.casino-proximite.fr/c/c/magasi...
2 Intermarche http://www.intermarche.com/magasin_accueil/sas...
3 Petit-Casino http://annuaire.casino-proximite.fr/c/c/magasi...
4 Carrefour-Market http://www.carrefour.fr/magasin/market-bellevi...

shop_name \
0 Simply Market MONTCEAUX GUEREINS
1 SPAR Montmerle Sur Saone
2 INTERMARCHE SUPER Belleville-sur-Saone
3 PETIT CASINO Belleville Sur Saone
4 Carrefour Market BELLEVILLE SUR SAONE

IMAGE_1 \
0 http://images.grandes-enseignes.com/Simply-Mar...
1 http://images.grandes-enseignes.com/Spar.png
2 http://images.grandes-enseignes.com/Intermarch...
3 http://images.grandes-enseignes.com/Petit-Casi...
4 http://images.grandes-enseignes.com/Carrefour-...

TEXT_3 SITE_INTERNET_LINK/_text brand \
0 01090 MONTCEAUX GUEREINS Site Internet Simply-Market
1 01090 Montmerle Sur Saone Site Internet Spar
2 69220 Belleville-sur-Saone Site Internet Intermarche
3 69220 Belleville Sur Saone Site Internet Petit-Casino
4 69220 BELLEVILLE SUR SAONE Site Internet Carrefour-Market

TEXT_4 ... \
0 NaN ...
1 Tel. 04 74 06 88 04 - ...
2 Tel. 04 74 06 45 85 - ...
3 Tel. 04 74 66 10 34 - ...
4 Tel. 04 74 06 44 10 - ...

result_address result_score result_type \
0 Grand Rivolet 01090 Montceaux 0.64 locality
1 4 Rue du Marche 01090 Montmerle-sur-Saône 0.91 housenumber
2 Prés de la Cloche 69220 Belleville 0.61 locality
3 47 Rue de la République 69220 Belleville 0.74 housenumber
4 Avenue de Verdun 69220 Belleville 0.71 street

result_id result_name result_street \
0 01258_B011_254fe4 Grand Rivolet NaN
1 ADRNIVX_0000000285677493 Rue du Marche NaN
2 69019_B043_44c71c Prés de la Cloche NaN
3 ADRNIVX_0000000259819137 Rue de la République NaN
4 69019_XXXX_9d303b Avenue de Verdun NaN

result_postcode result_city result_context result_citycode
0 1090 Montceaux 01, Ain, Rhône-Alpes 1258
1 1090 Montmerle-sur-Saône 01, Ain, Rhône-Alpes 1263

```



```

2          69220          Belleville 69, Rhône, Rhône-Alpes          69019
3          69220          Belleville 69, Rhône, Rhône-Alpes          69019
4          69220          Belleville 69, Rhône, Rhône-Alpes          69019

```

```
[5 rows x 22 columns]
```

```
In [35]: df.columns
```

```
Out[35]: Index(['_id', 'ADRESSE_POSTALE_TEXT', 'TEXT_1', 'IMAGE_1/_title',
              'SITE_INTERNET_LINK', 'shop_name', 'IMAGE_1', 'TEXT_3',
              'SITE_INTERNET_LINK/_text', 'brand', 'TEXT_4', 'latitude', 'longitude',
              'result_address', 'result_score', 'result_type', 'result_id',
              'result_name', 'result_street', 'result_postcode', 'result_city',
              'result_context', 'result_citycode'],
              dtype='object')
```

```
In [36]: df = cl.df_head(bs, "croix-rouge", "build/Test_CRFFOR.GACCTMPD.csv", as_df=False, stop_at=2000)
df.split("\n")[0]
```

```
Out[36]: '"TYP_0","NUM_0","LIN_0","LEDTYP_0","LED_0","ACCCNUM_0","CHRRNUM_0","IDTLIN_0","CPY_0","FCYLIN_0"
```

## 1.5 Close connection

```
In [37]: %blob_close
```

```
Out[37]: True
```

## 1.6 Use module ensae\_projects

Module `ensae_projects` includes helpers for this event. Here is how to get the joined schemas for all tables in one Excel file. Some data is not included in the module but it is encrypted. You need a password given to any participant. You can store it in environment variable `PWDCROIXROUGE` to avoid typing it each time you need it.

### 1.6.1 joined schemas

```
In [38]: from ensae_projects.datainc.croix_rouge import merge_schema
df = merge_schema()
```

```
In [39]: df.head(n=1)
```

```
Out[39]:
```

| index | Zone | name_ie | name_IR | name_SE | name_SV  | name_su | Intitulé long | \              |
|-------|------|---------|---------|---------|----------|---------|---------------|----------------|
| 0     | 0    | ACCCDAT | invoice | NaN     | SINVOICE | NaN     | NaN           | Date comptable |

  

| Typ | Menu | Long | Act | Dim | Table liée | Expression de lien | Vérification | \ |
|-----|------|------|-----|-----|------------|--------------------|--------------|---|
| 0   | D    |      |     |     |            |                    |              |   |

  

| Obligatoire | RAZ     |
|-------------|---------|
| 0           | Non Non |

```
In [40]: df.to_excel("joined_schemas.xlsx")
```

## 1.6.2 specific schemas

```
In [41]: from ensae_projects.datainc.croix_rouge import get_meaning
df = get_meaning("ITMASTER")
df.head(n=2)
```

```
Out[41]:
```

|   | Intitulé long       | Options | Table liée | Expression de lien       | \   |
|---|---------------------|---------|------------|--------------------------|-----|
| 0 | Article             | NaN     | ITMASTER   |                          | NaN |
| 1 | Famille statistique | NaN     | ATABDIV    | indice+20;TSICOD(indice) |     |

  

|   | Copie législation | Annulation  | Vérification | Obligatoire | RAZ | Zone |         |
|---|-------------------|-------------|--------------|-------------|-----|------|---------|
| 0 | NaN               | Suppression |              | Oui         | Non | Non  | ITM_001 |
| 1 | NaN               | Bloquant    | 1            | Oui         | Non | Non  | ITM_002 |

```
In [42]: from ensae_projects.datainc.croix_rouge import get_meaning
df = get_meaning("SINVOICE")
df.head(n=2)
```

```
Out[42]:
```

|   | Zone   | Typ | Menu | Long | Act | Dim | Intitulé normal   | Intitulé abrégé | \ |
|---|--------|-----|------|------|-----|-----|-------------------|-----------------|---|
| 0 | SIVTYP | TSV | 0    | NaN  | NaN | 1   | Type facture      | Type fac        |   |
| 1 | INVTYP | M   | 645  | 15   | NaN | 1   | Catégorie facture | Catég fac       |   |

  

|   | Intitulé long     | Options | Table liée | Expression de lien | Annulation | \        |
|---|-------------------|---------|------------|--------------------|------------|----------|
| 0 | Type facture      | NaN     | TABSIVTYP  |                    | NaN        | Bloquant |
| 1 | Catégorie facture | NaN     | NaN        |                    | NaN        | NaN      |

  

|   | Vérification | Obligatoire | RAZ | Mot-clé d'aide |
|---|--------------|-------------|-----|----------------|
| 0 | Oui          | Non         | Non | NaN            |
| 1 | NaN          | Non         | Non | NaN            |

```
In [43]: from ensae_projects.datainc.croix_rouge import get_meaning
df = get_meaning("SINVOICE_V")
df.head(n=2)
```

```
Out[43]:
```

|   | Zone | Typ | Menu | Long | Act | Dim | Intitulé normal | Intitulé abrégé | \ |
|---|------|-----|------|------|-----|-----|-----------------|-----------------|---|
| 0 | NUM  | VCR | 0    | NaN  | NaN | 1   | No facture      | Facture         |   |
| 1 | CPY  | CPY | 0    | NaN  | NaN | 1   | Société         | Société         |   |

  

|   | Intitulé long | Table liée | Expression de lien | Annulation | Vérification | \   |
|---|---------------|------------|--------------------|------------|--------------|-----|
| 0 | No facture    | SINVOICE   |                    | NaN        | Autre        | Oui |
| 1 | Société       | COMPANY    |                    | NaN        | Bloquant     | Oui |

  

|   | Obligatoire | RAZ |
|---|-------------|-----|
| 0 | Non         | Non |
| 1 | Oui         | Non |

```
In [44]:
```