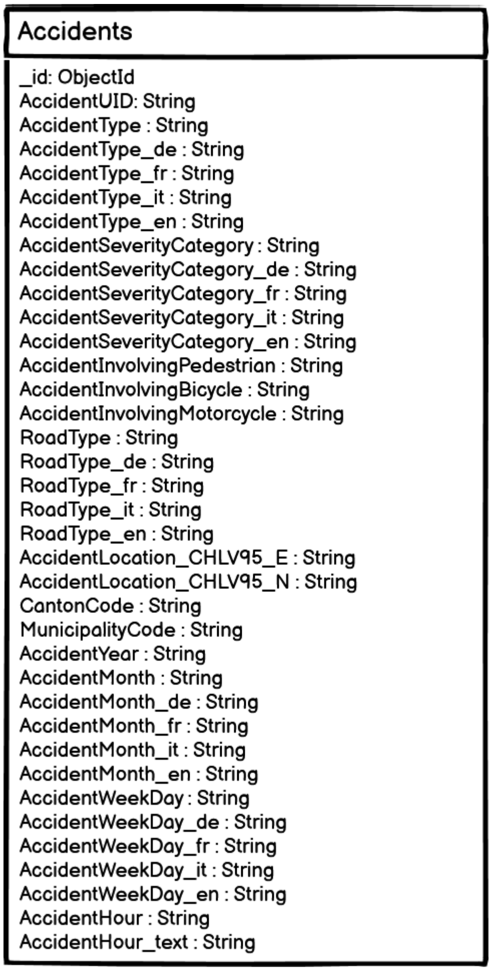
Project

May 27, 2021

1 BDL01 Semesterproject

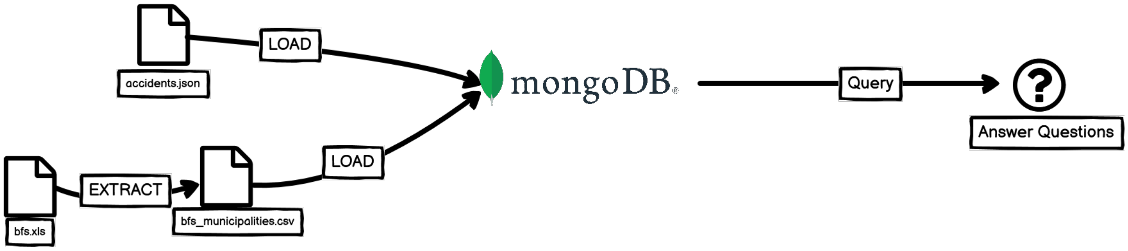
1.1 Analysis On Police-registered traffic accidents in the canton of Zurich

The traffic accident statistics of the Canton of Zurich (VUSTA) contains the road traffic accidents with personal injury and property damage registered by the Cantonal Police of Zurich, the Traffic Department of the City of Zurich and the Winterthur City Police. It is updated once a year, towards the end of the first quarter of the following year. The dataset covers the time from January 1, 2011 - December 31, 2020



Accidents

_id: ObjectId
AccidentUID: String
AccidentType : String
AccidentType_de : String
AccidentType_fr : String
AccidentType_it : String
AccidentType_en : String
AccidentSeverityCategory : String
AccidentSeverityCategory_de : String
AccidentSeverityCategory_fr : String
AccidentSeverityCategory_it : String
AccidentSeverityCategory_en : String
AccidentInvolvingPedestrian : String
AccidentInvolvingBicycle : String
AccidentInvolvingMotorcycle : String
RoadType : String
RoadType_de : String
RoadType_fr : String
RoadType_it : String
RoadType_en : String
AccidentLocation_CHLV95_E : String
AccidentLocation_CHLV95_N : String
CantonCode : String
MunicipalityCode : String
AccidentYear : String
AccidentMonth : String
AccidentMonth_de : String
AccidentMonth_fr : String
AccidentMonth_it : String
AccidentMonth_en : String
AccidentWeekDay : String
AccidentWeekDay_de : String
AccidentWeekDay_fr : String
AccidentWeekDay_it : String
AccidentWeekDay_en : String
AccidentHour : String
AccidentHour_text : String



2 Installation, Requirements

[1]:

```
#! pip3 list | grep -E 'pymongo|json|pandas|requests|matplotlib|numpy|pprint'
```

[2]:

```
import pandas as pd
from pymongo import MongoClient
from pprint import pprint
import pandas as pd
import requests
from datetime import datetime
from datetime import timedelta
import time
import json
import bigjson
import numpy as np
from pymongo.errors import DuplicateKeyError
from pymongo.errors import OperationFailure, DuplicateKeyError
import matplotlib.pyplot as plt

pd.set_option('precision', 2)
pd.set_option('max_rows', 20)
pd.set_option('max_colwidth', 30)
# pd.describe_option('max_rows')
# pd.describe_option('precision')
# pd.describe_option('max_colwidth')
pd.set_option("display.max_columns", None)
plt.rcParams["figure.figsize"] = (20,10)
#Reset do default figsize
#plt.rcParams["figure.figsize"] = plt.rcParamsDefault["figure.figsize"]
```

2.0.1 Connect, dbs

[3]:

```
client = MongoClient(host="localhost",port=27017)
database = client["bdl03"]
```

3 ETL/ ELT

3.1 Import Files

The accident data can be retrieved from https://opendata.swiss/en/dataset/polizeilich-registrierte-verkehrsunfalle-im-kanton-zurich/resource/e0758b22-1e77-4d96-aacd-18ced0ba3781 or directly be read from https://www.web.statistik.zh.ch/ogd/data/kapo/RoadTrafficAccidentLocations.json

[4]:

```
request = requests.get("https://www.web.statistik.zh.ch/ogd/data/kapo/
 ↪RoadTrafficAccidentLocations.json")
if request.status_code == 200:
        data = request.json()
```

[5]:

```
#data
```

[6]:

```
    #try:
        #database.Accidents.insert_one(data)
```

[7]:

```
c = database.Accidents.aggregate([
      {"$limit": 1},
])

for doc in c:
     pprint(f"{doc}"[:5000])
```

```
("{'_id': ObjectId('60afc7769ddbd12dc82a1da1'), 'AccidentUID': "
 "'A70191D0D45E00B0E0430A83942700B0', 'AccidentType': 'at0', "
 "'AccidentType_de': 'Schleuder- oder Selbstunfall', 'AccidentType_fr': "
 "'dérapage ou perte de maîtrise', 'AccidentType_it': 'Incidente di "
 "sbandamento o per colpa propria', 'AccidentType_en': 'Accident with skidding "
 "or self-accident', 'AccidentSeverityCategory': 'as4', "
 "'AccidentSeverityCategory_de': 'Unfall mit Sachschaden', "
 "'AccidentSeverityCategory_fr': 'accident avec dommages matériels', "
 "'AccidentSeverityCategory_it': 'Incidente con danni materiali', "
 "'AccidentSeverityCategory_en': 'Accident with property damage', "
 "'AccidentInvolvingPedestrian': 'false', 'AccidentInvolvingBicycle': 'false', "
 "'AccidentInvolvingMotorcycle': 'false', 'RoadType': 'rt432', 'RoadType_de': "
 "'Hauptstrasse', 'RoadType_fr': 'route principale', 'RoadType_it': 'Strada "
 "principale', 'RoadType_en': 'Principal road', 'AccidentLocation_CHLV95_E': "
 "'2676380', 'AccidentLocation_CHLV95_N': '1250175', 'CantonCode': 'ZH', "
 "'MunicipalityCode': '0247', 'AccidentYear': '2011', 'AccidentMonth': '1', "
 "'AccidentMonth_de': 'Januar', 'AccidentMonth_fr': 'janvier', "
 "'AccidentMonth_it': 'Gennaio', 'AccidentMonth_en': 'January', "
 "'AccidentWeekDay': 'aw406', 'AccidentWeekDay_de': 'Samstag', "
 "'AccidentWeekDay_fr': 'samedi', 'AccidentWeekDay_it': 'Sabato', "
 "'AccidentWeekDay_en': 'Saturday', 'AccidentHour': '00', 'AccidentHour_text': "
 "'00h-01h'}")
```

[8]:

```python
c = database.Accidents.aggregate([
      {"$limit": 1},
])

pd.DataFrame(c)
```

[8]:

```
                        _id                    AccidentUID AccidentType  \
0  60afc7769ddbd12dc82a1da1  A70191D0D45E00B0E0430A8394…          at0   

                AccidentType_de                AccidentType_fr  \
0  Schleuder- oder Selbstunfall  dérapage ou perte de maîtrise   

                 AccidentType_it                AccidentType_en  \
0  Incidente di sbandamento o…  Accident with skidding or …   

  AccidentSeverityCategory AccidentSeverityCategory_de  \
0                      as4      Unfall mit Sachschaden   

     AccidentSeverityCategory_fr    AccidentSeverityCategory_it  \
0  accident avec dommages mat…  Incidente con danni materiali   

     AccidentSeverityCategory_en AccidentInvolvingPedestrian  \
0  Accident with property damage                       false   

  AccidentInvolvingBicycle AccidentInvolvingMotorcycle RoadType   RoadType_de  \
0                    false                       false    rt432  Hauptstrasse   

        RoadType_fr        RoadType_it     RoadType_en  \
0  route principale  Strada principale  Principal road   

  AccidentLocation_CHLV95_E AccidentLocation_CHLV95_N CantonCode  \
0                   2676380                   1250175         ZH   

  MunicipalityCode AccidentYear AccidentMonth AccidentMonth_de  \
0             0247         2011             1           Januar   

  AccidentMonth_fr AccidentMonth_it AccidentMonth_en AccidentWeekDay  \
0          janvier          Gennaio          January           aw406   

  AccidentWeekDay_de AccidentWeekDay_fr AccidentWeekDay_it AccidentWeekDay_en  \
0            Samstag             samedi             Sabato           Saturday   

  AccidentHour AccidentHour_text
0           00           00h-01h
```

3.2 Validate Fields

[9]:
```
c = database.Accidents.aggregate([
    {"$match": {"AccidentType_en": {"$exists" : False}}},
])

pd.DataFrame(c)
```

[9]:
```
Empty DataFrame
Columns: []
Index: []
```

[10]:
```
c = database.Accidents.aggregate([
    {"$match": {'AccidentSeverityCategory_de': {"$in" : ['Unfall mit␣
 ↪Sachschaden']}}},
])


df = pd.DataFrame(c)
```

3.2.1 (Re)Create Accidents collection

[11]:
```
c = database.Accidents.aggregate([
    {"$project": {"_id": "$AccidentUID", 'AccidentType':1, 'AccidentType_en':1,
       'AccidentSeverityCategory':1,
       'AccidentSeverityCategory_en':1, 'AccidentInvolvingPedestrian':1,
       'AccidentInvolvingBicycle':1, 'AccidentInvolvingMotorcycle':1,␣
 ↪'RoadType':1, 'RoadType_en':1,'CantonCode':1,
       'MunicipalityCode':1, 'AccidentYear':1, 'AccidentMonth':
 ↪1,'AccidentMonth_en':1,
       'AccidentWeekDay':1, 'AccidentWeekDay_en':1, 'AccidentHour':1,
       'AccidentHour_text':1}},
])
df = pd.DataFrame(c)
```

[12]:
```
df.head()
```

[12]:
```
  AccidentType                AccidentType_en AccidentSeverityCategory  \
0          at0  Accident with skidding or …                      as4
1          at0  Accident with skidding or …                      as4
2          at0  Accident with skidding or …                      as3
3          at0  Accident with skidding or …                      as4
4          at0  Accident with skidding or …                      as4

     AccidentSeverityCategory_en AccidentInvolvingPedestrian  \
0  Accident with property damage                       false
1  Accident with property damage                       false
2   Accident with light injuries                       false
```

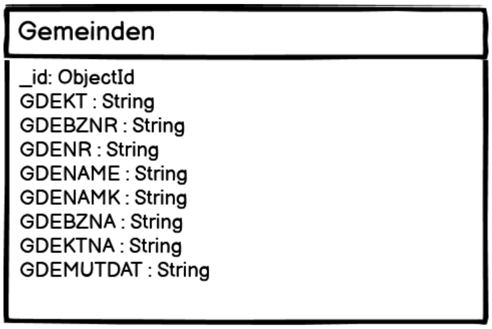
```
3  Accident with property damage                       false
4  Accident with property damage                       false

  AccidentInvolvingBicycle AccidentInvolvingMotorcycle RoadType  \
0                    false                       false    rt432
1                    false                       false    rt433
2                     true                       false    rt433
3                    false                       false    rt430
4                    false                       false    rt439

      RoadType_en CantonCode MunicipalityCode AccidentYear AccidentMonth  \
0  Principal road         ZH             0247         2011             1
1      Minor road         ZH             0261         2011             1
2      Minor road         ZH             0261         2011             1
3        Motorway         ZH             0251         2011             1
4           Other         ZH             0261         2011             1

  AccidentMonth_en AccidentWeekDay AccidentWeekDay_en AccidentHour  \
0          January           aw406           Saturday           00
1          January           aw406           Saturday           00
2          January           aw406           Saturday           01
3          January           aw406           Saturday           01
4          January           aw406           Saturday           02

  AccidentHour_text                            _id
0           00h-01h  A70191D0D45E00B0E0430A8394…
1           00h-01h  A2D2677533867004E0430A865E…
2           01h-02h  9FD6441F802C20A6E0430A865E…
3           01h-02h  A7016B9BBC3301A8E0430A8394…
4           02h-03h  9FDA0DC4856A6094E0430A865E…
```

[13]:
```
df.columns
```

[13]:
```
Index(['AccidentType', 'AccidentType_en', 'AccidentSeverityCategory',
       'AccidentSeverityCategory_en', 'AccidentInvolvingPedestrian',
       'AccidentInvolvingBicycle', 'AccidentInvolvingMotorcycle', 'RoadType',
       'RoadType_en', 'CantonCode', 'MunicipalityCode', 'AccidentYear',
       'AccidentMonth', 'AccidentMonth_en', 'AccidentWeekDay',
       'AccidentWeekDay_en', 'AccidentHour', 'AccidentHour_text', '_id'],
      dtype='object')
```

3.3 Municipality Information

To get a Municipality Name to the Municipality code an .xls file has been downloaded from https://www.bfs.admin.ch/bfs/de/home/grundlagen/agvch.assetdetail.16924990.html. I then created a .csv where only the municipalities from Zurich are listed. (bfs_municipality.csv can be found in the zip)



Gemeinden

_id: ObjectId
GDEKT : String
GDEBZNR : String
GDENR : String
GDENAME : String
GDENAMK : String
GDEBZNA : String
GDEKTNA : String
GDEMUTDAT : String

3.3.1 Drop the Gemeinden Collection

[14]:

```
database.gemeinden.drop()
c = database.list_collections()
pd.DataFrame(c)
```

[14]:

```
        name        type options                                  info  \
0  Accidents  collection      {}  {'readOnly': False, 'uuid'…

                        idIndex
0  {'v': 2, 'key': {'_id': 1}…
```

[15]:

```
gemeinden_df = pd.read_csv('bfs_municipality.csv',sep=";")   # loading csv file

row_dict={}
for column in gemeinden_df:
    row_dict[column]= []

for index, row in gemeinden_df.iterrows():
    json_row =row.to_dict()
    #print(json_row)
    database.gemeinden.insert_one(json_row)
```

[16]:

```
gemeinden_df.columns
```

[16]:

```
Index(['GDEKT', 'GDEBZNR', 'GDENR', 'GDENAME', 'GDENAMK', 'GDEBZNA', 'GDEKTNA',
       'GDEMUTDAT'],
      dtype='object')
```

[17]:

```
c=database.gemeinden.aggregate([
      {"$limit": 1},
])

pd.DataFrame(c)
```

[17]:

```
                        _id GDEKT  GDEBZNR  GDENR         GDENAME  \
0  60b00be931332501ad713488    ZH    101.0    1.0  Aeugst am Albis

           GDENAMK           GDEBZNA GDEKTNA   GDEMUTDAT
0  Aeugst am Albis  Bezirk Affoltern  Zürich  1976-11-15
```

3.4 Convert Municipality Code to Int

[18]:

```
priceQtyConversionStage = {
   '$addFields': {
      'IntMunicipalityCode': { "$toInt": "$MunicipalityCode" },
   }
}

c=database.Accidents.aggregate( [
   priceQtyConversionStage,
])
e=pd.DataFrame(c)
```

[19]:

```
e.head()
```

[19]:

```
                        _id                    AccidentUID AccidentType  \
0  60afc7769ddbd12dc82a1da1  A70191D0D45E00B0E0430A8394…          at0
1  60afc7769ddbd12dc82a1da2  A2D2677533867004E0430A865E…          at0
2  60afc7769ddbd12dc82a1da3  9FD6441F802C20A6E0430A865E…          at0
3  60afc7769ddbd12dc82a1da4  A7016B9BBC3301A8E0430A8394…          at0
4  60afc7769ddbd12dc82a1da5  9FDA0DC4856A6094E0430A865E…          at0

                AccidentType_de                AccidentType_fr  \
0  Schleuder- oder Selbstunfall  dérapage ou perte de maîtrise
1  Schleuder- oder Selbstunfall  dérapage ou perte de maîtrise
2  Schleuder- oder Selbstunfall  dérapage ou perte de maîtrise
3  Schleuder- oder Selbstunfall  dérapage ou perte de maîtrise
4  Schleuder- oder Selbstunfall  dérapage ou perte de maîtrise

                 AccidentType_it                AccidentType_en  \
0  Incidente di sbandamento o…  Accident with skidding or …
1  Incidente di sbandamento o…  Accident with skidding or …
2  Incidente di sbandamento o…  Accident with skidding or …
3  Incidente di sbandamento o…  Accident with skidding or …
4  Incidente di sbandamento o…  Accident with skidding or …

  AccidentSeverityCategory  AccidentSeverityCategory_de  \
0                      as4       Unfall mit Sachschaden
1                      as4       Unfall mit Sachschaden
2                      as3  Unfall mit Leichtverletzten
3                      as4       Unfall mit Sachschaden
```

```
4                      as4       Unfall mit Sachschaden

     AccidentSeverityCategory_fr    AccidentSeverityCategory_it  \
0  accident avec dommages mat…  Incidente con danni materiali
1  accident avec dommages mat…  Incidente con danni materiali
2   accident avec blessés légers   Incidente con feriti leggeri
3  accident avec dommages mat…  Incidente con danni materiali
4  accident avec dommages mat…  Incidente con danni materiali

     AccidentSeverityCategory_en AccidentInvolvingPedestrian  \
0  Accident with property damage                       false
1  Accident with property damage                       false
2   Accident with light injuries                       false
3  Accident with property damage                       false
4  Accident with property damage                       false

  AccidentInvolvingBicycle AccidentInvolvingMotorcycle RoadType   RoadType_de  \
0                    false                       false    rt432  Hauptstrasse
1                    false                       false    rt433  Nebenstrasse
2                     true                       false    rt433  Nebenstrasse
3                    false                       false    rt430      Autobahn
4                    false                       false    rt439        andere

        RoadType_fr        RoadType_it     RoadType_en  \
0  route principale  Strada principale  Principal road
1  route secondaire  Strada secondaria      Minor road
2  route secondaire  Strada secondaria      Minor road
3         autoroute         Autostrada        Motorway
4             autre              Altro           Other

  AccidentLocation_CHLV95_E AccidentLocation_CHLV95_N CantonCode  \
0                   2676380                   1250175         ZH
1                   2684605                   1245194         ZH
2                   2682382                   1246980         ZH
3                   2674666                   1251733         ZH
4                   2682791                   1247749         ZH

  MunicipalityCode AccidentYear AccidentMonth AccidentMonth_de  \
0             0247         2011             1           Januar
1             0261         2011             1           Januar
2             0261         2011             1           Januar
3             0251         2011             1           Januar
4             0261         2011             1           Januar

  AccidentMonth_fr AccidentMonth_it AccidentMonth_en AccidentWeekDay  \
0          janvier          Gennaio          January           aw406
1          janvier          Gennaio          January           aw406
```

```
2          janvier          Gennaio          January           aw406
3          janvier          Gennaio          January           aw406
4          janvier          Gennaio          January           aw406

  AccidentWeekDay_de AccidentWeekDay_fr AccidentWeekDay_it AccidentWeekDay_en  \
0            Samstag             samedi             Sabato           Saturday
1            Samstag             samedi             Sabato           Saturday
2            Samstag             samedi             Sabato           Saturday
3            Samstag             samedi             Sabato           Saturday
4            Samstag             samedi             Sabato           Saturday

  AccidentHour AccidentHour_text  IntMunicipalityCode
0           00           00h-01h                  247
1           00           00h-01h                  261
2           01           01h-02h                  261
3           01           01h-02h                  251
4           02           02h-03h                  261
```

4 Data Analysis

4.1 Categories

[20]:

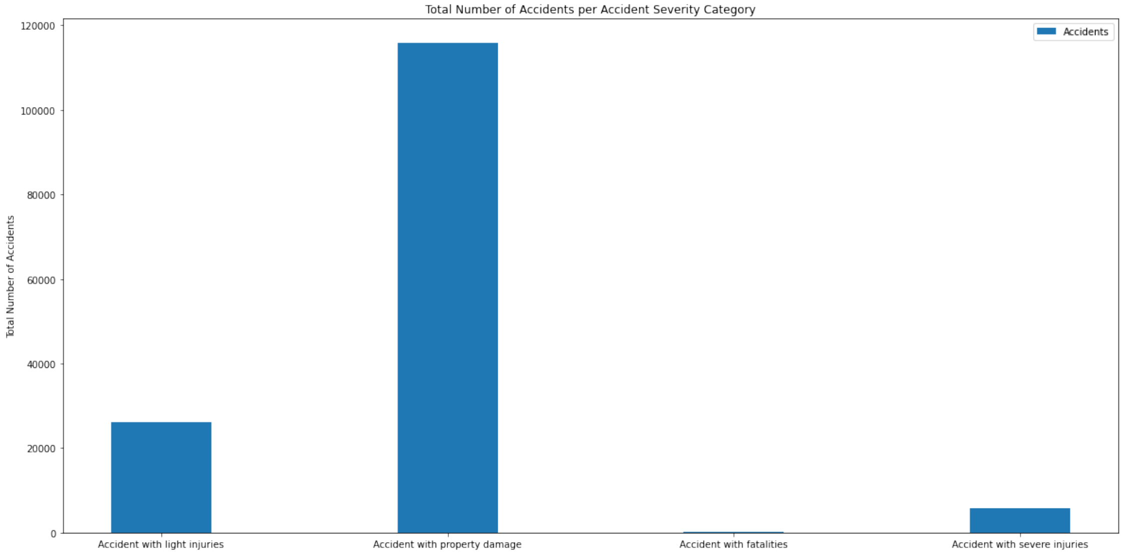
```
c = database.Accidents.aggregate([
    {"$project": {'AccidentType':0,  'AccidentSeverityCategory':0,␣
 ↪'AccidentInvolvingPedestrian':0,
        'AccidentInvolvingBicycle':0, 'AccidentInvolvingMotorcycle':0,␣
 ↪'RoadType':0,
        'RoadType_en':0, 'CantonCode':0, 'MunicipalityCode':0, 'AccidentYear':0,
        'AccidentMonth':0, 'AccidentMonth_en':0, 'AccidentWeekDay':0,
        'AccidentWeekDay_en':0, 'AccidentHour':0, 'AccidentHour_text':0}},
    {"$unwind": "$AccidentSeverityCategory_en"},
    {"$group": {"_id": "$AccidentSeverityCategory_en", "count": {"$sum": 1}}},
 ])

df = pd.DataFrame(c)
print(df)
```

```
                             _id   count
0    Accident with light injuries   26208
1   Accident with property damage  115793
2        Accident with fatalities     286
3   Accident with severe injuries    5684
```

Eventhoug there are alot of accidents in the canton of Zürich, there are gladly almost no accidents with fatalities.

[21]:

```
labels = df["_id"]

width = 0.35       # the width of the bars: can also be len(x) sequence
fig, ax = plt.subplots()

ax.bar(labels, df["count"], width, label='Accidents')

ax.set_ylabel('Total Number of Accidents')
ax.set_title('Total Number of Accidents per Accident Severity Category')
ax.legend()

plt.show()
```



[22]:

```
c = database.Accidents.aggregate([
    {"$project": {'AccidentType':0,  'AccidentSeverityCategory':0,␣
 ↪'AccidentInvolvingPedestrian':0,
       'AccidentInvolvingBicycle':0, 'AccidentInvolvingMotorcycle':0,␣
 ↪'RoadType':0,
       'RoadType_en':0, 'CantonCode':0, 'MunicipalityCode':0, 'AccidentYear':0,
       'AccidentMonth':0, 'AccidentMonth_en':0, 'AccidentWeekDay':0,
       'AccidentWeekDay_en':0, 'AccidentHour':0, 'AccidentHour_text':0}},
    {"$unwind": "$AccidentType_en"},
    {"$group": {"_id": "$AccidentType_en", "count": {"$sum": 1}}},
 ])

df = pd.DataFrame(c)
print(df)
```

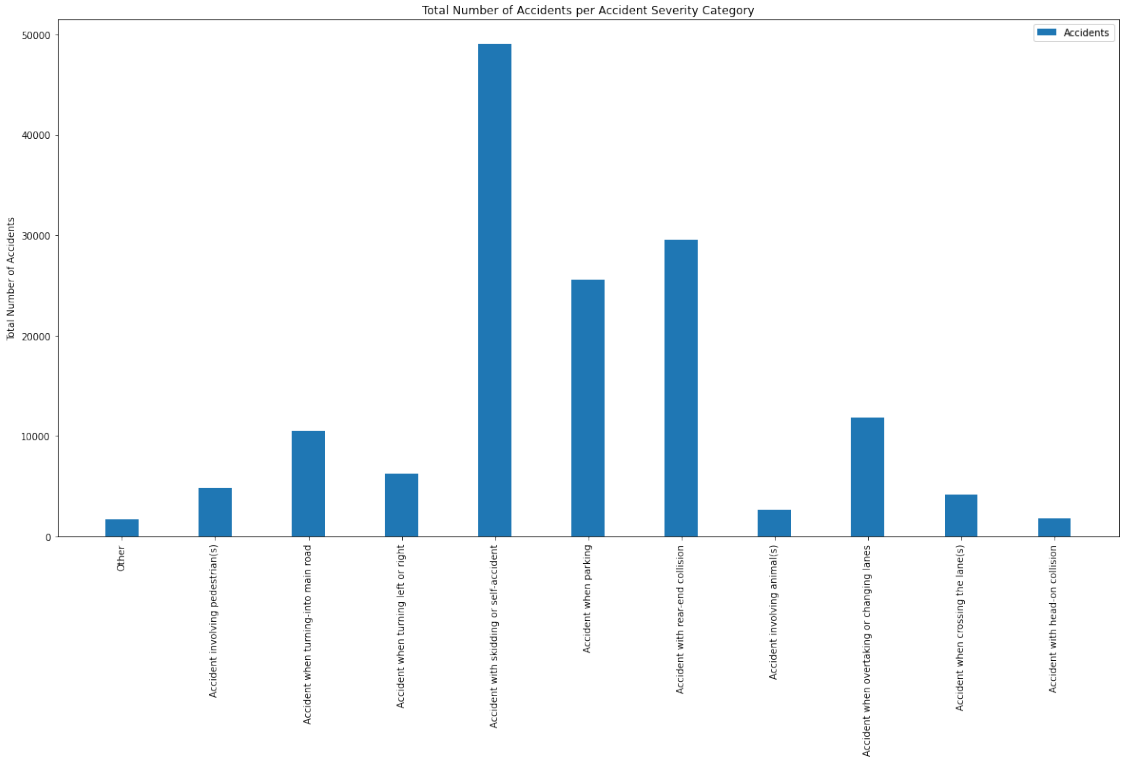
```
_id   count
```

```
0                           Other   1755
1   Accident involving pedestr…   4849
2   Accident when turning-into…  10486
3   Accident when turning left…   6295
4   Accident with skidding or …  49019
5           Accident when parking  25583
6   Accident with rear-end col…  29513
7    Accident involving animal(s)   2642
8   Accident when overtaking o…  11830
9   Accident when crossing the…   4159
10  Accident with head-on coll…   1840
```

[23]:

```
labels = df["_id"]

width = 0.35       # the width of the bars: can also be len(x) sequence
fig, ax = plt.subplots()

ax.bar(labels, df["count"], width, label='Accidents')
#plt.xticks(rotation = 45)
ax.set_xticklabels(labels, rotation = 90)
ax.set_ylabel('Total Number of Accidents')
ax.set_title('Total Number of Accidents per Accident Severity Category')
ax.legend()

plt.show()
```

```
<ipython-input-23-25c4e054aff2>:8: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(labels, rotation = 90)
```



It is interesting to see, that most accidents are skiddin or self accidents. The next biggest accident type is accidents with rear-end collisions. We can see, that a lot of accidents could be prevented if drivers would be more attentive.

```
[24]: c = database.Accidents.aggregate([
          {"$project": {'AccidentType':0,  'AccidentSeverityCategory':0,␣
       ↪'AccidentInvolvingPedestrian':0,
              'AccidentInvolvingBicycle':0, 'AccidentInvolvingMotorcycle':0,␣
       ↪'RoadType':0,
              'RoadType_en':0, 'AccidentYear':0,
              'AccidentMonth':0, 'AccidentMonth_en':0, 'AccidentWeekDay':0,
              'AccidentWeekDay_en':0, 'AccidentHour':0, 'AccidentHour_text':0}},
          {"$unwind": "$MunicipalityCode"},
          {"$group": {"_id": "$MunicipalityCode", "count": {"$sum": 1}}},
      ])

      df = pd.DataFrame(c)
```

```
[25]: df.head()
```

```
[25]:     _id  count
      0  0112    445
      1  0031    213
```

```
2  0072    315
3  0037     98
4  0248    187
```

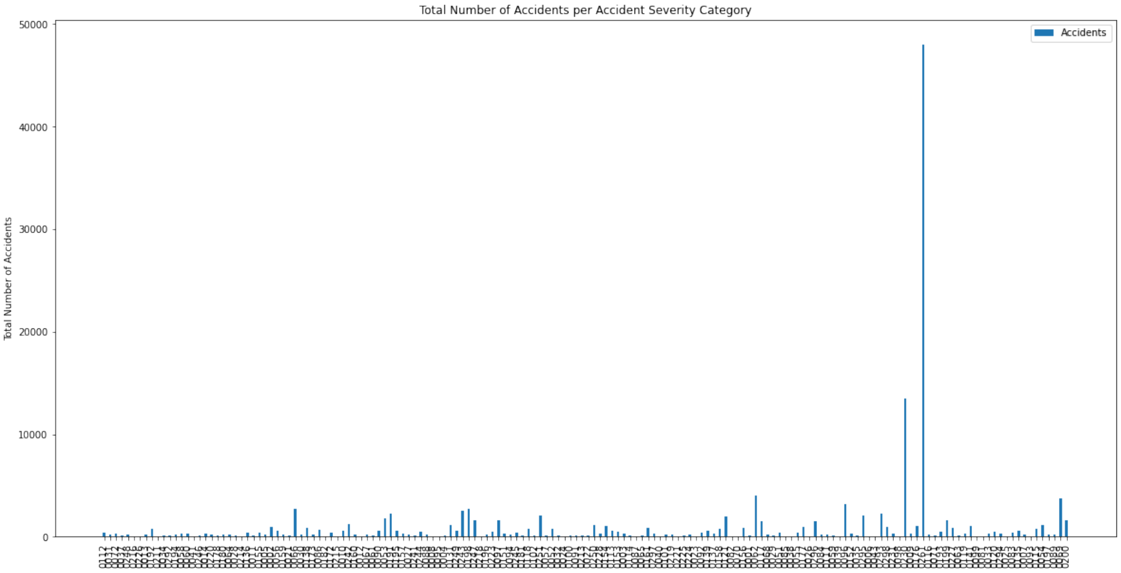
Unfortunately I was not able to merge the two collections on the MunicipalyCode. In a future project it would be very interesting to analyse in more detail, in which municipalities the most accidents happen.

The figure below shows one municipality standing out. I believe that this is the city of Zürich.

[26]:

```
labels = df["_id"]

width = 0.39       # the width of the bars: can also be len(x) sequence
fig, ax = plt.subplots()

ax.bar(labels, df["count"], width, label='Accidents')
#plt.xticks(rotation = 45)
ax.set_xticklabels(labels, rotation = 90)
ax.set_ylabel('Total Number of Accidents')
ax.set_title('Total Number of Accidents per Accident Severity Category')
ax.legend()

plt.show()
```

```
<ipython-input-26-d09722fe2115>:8: UserWarning: FixedFormatter should only be
used together with FixedLocator
  ax.set_xticklabels(labels, rotation = 90)
```



5 Conclusion

MongoDB, the most popular NoSQL database, is a relative newcomer in the database industry. It's an excellent tool for creating data warehouses, because because of its ability to fully exploit so-called "shared-nothing cluster architecture." Because it is an open-source database, it is perfect for creating high-performance data warehouses.

This semester project served as an excellent introduction to MongoDB. It needs some time to get used to the syntax. However, one rapidly grows used to it.