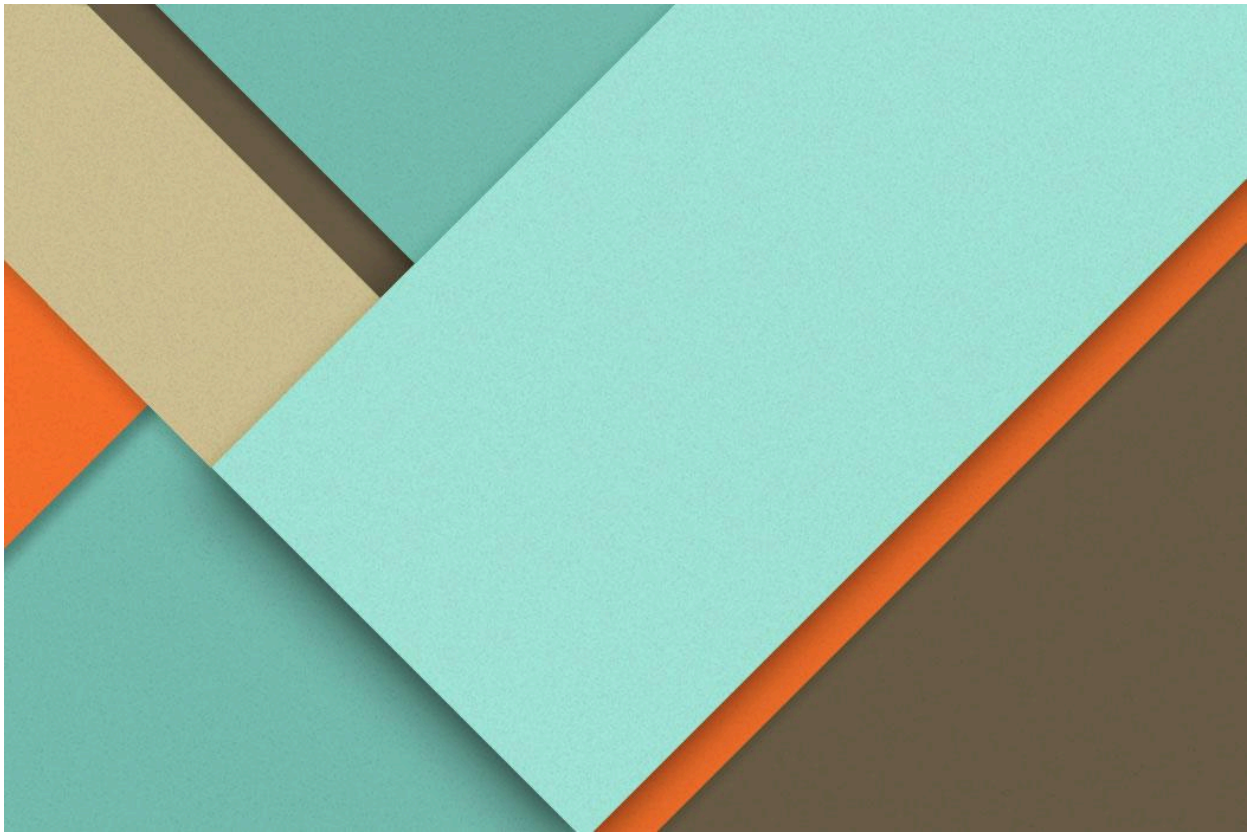


Projet 2 DAAR 2025

02.11.2025

—

MARSSO Denn 28608482 - MOUGAMADOUBOUGARY Mohamed 28604611
M2 STL

LIEN POUR LA DASHBOARD :
https://my-elasticsearch-deployment-436a04.kb.europe-west9.gcp.elastic-cloud.com/app/r/s/j6kyG

I. Elastic API requests

Q1-Q2

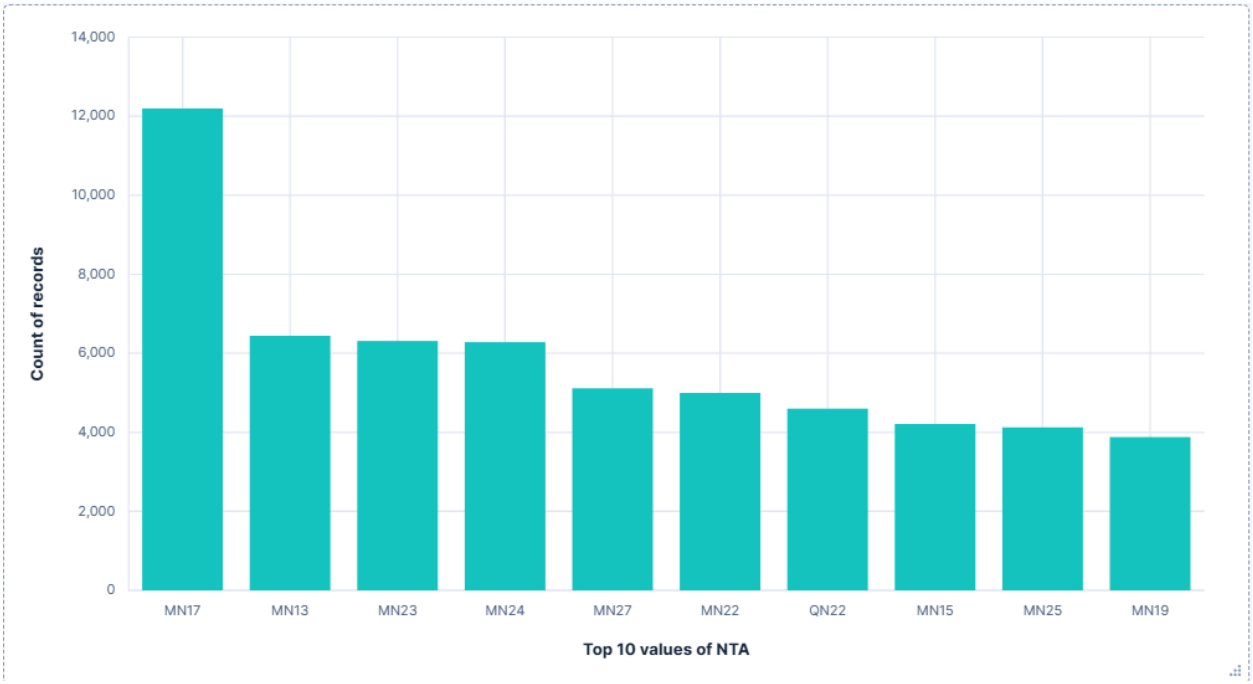
```
GET ny_restaurants/_search
{
  "size": 0,
  "aggs": {
    "unique_neighborhoods": {
      "cardinality": {
        "field": "NTA"
      }
    },
    "neighborhoods": {
      "terms": {
        "field": "NTA",
        "size": 1000,
        "order": { "_count": "desc" }
      },
      "aggs": {
        "boro": {
          "terms": { "field": "BORO", "size": 1 }
        }
      }
    }
  }
}
```

Cette requête permet à la fois de lister tous les quartiers de New York présents dans le dataset et d'identifier celui avec le plus de restaurants. NTA signifie Neighborhood Tabulation Area et sert à regrouper les restaurants par quartier géographique au sein des 5 boroughs de New York.

La première partie de la requête utilise une agrégation de type *cardinality* sur le champ NTA pour avoir le nombre total de quartiers distincts.

La deuxième partie regroupe les restaurants par quartier grâce à une agrégation *terms* sur le même champ, triée par nombre de documents décroissant.

Une sous-agrégation sur le champ *BORO* permet de savoir à quel arrondissement appartient chaque quartier (ce n'était pas demandé, je l'ai affiché sans aucune raison).



```
{
  "took": 66,
  "timed_out": false,
  "_shards": {
    "total": 1,
    "successful": 1,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 10000,
      "relation": "gte"
    },
    "max_score": null,
    "hits": []
  },
  "aggregations": {
    "unique_neighborhoods": {
      "value": 193
    },
    "neighborhoods": {
      "doc_count_error_upper_bound": 0,
      "sum_other_doc_count": 0,
      "buckets": [
        {
          "key": "MN17",
          "doc_count": 12195,
          "boro": {
            "doc_count_error_upper_bound": 0,
            "sum_other_doc_count": 0,
            "buckets": [
              {
                "key": "Manhattan",
                "doc_count": 12195
              }
            ]
          }
        },
        {
          "key": "MN13",
          "doc_count": 6442,
          "boro": {
            "doc_count_error_upper_bound": 0,
            "sum_other_doc_count": 0,
            "buckets": [
              {
                "key": "Manhattan",
                "doc_count": 6442
              }
            ]
          }
        },
        {
          "key": "MN23",
          "doc_count": 6312,
          "boro": {
            "doc_count_error_upper_bound": 0,
            "sum_other_doc_count": 0,
            "buckets": [
              {
                "key": "Manhattan",
                "doc_count": 6312
              }
            ]
          }
        },
        {
          "key": "MN24",
          "doc_count": 6281,
          "boro": {
            "doc_count_error_upper_bound": 0,
            "sum_other_doc_count": 0,
            "buckets": [
              {
                "key": "Manhattan",
                "doc_count": 6281
              }
            ]
          }
        },
```


Le résultat montre que le dataset contient 193 quartiers distincts à New York (`"value": 193` dans l'agrégation `unique_neighborhoods`). Ensuite, la partie `neighborhoods` affiche les quartiers triés par nombre de restaurants décroissant.

Le premier élément du tableau (`"key": "MN17"`) correspond donc au quartier ayant le plus de restaurants, avec un total de 12 195 établissements (`"doc_count"`). La sous-agrégation `boro` indique que ce quartier appartient à Manhattan. Le second qui contient 6442 établissements est également situé à Manhattan et ainsi de suite.

Pour résumer, il y a 193 quartiers répertoriés dans la base de données, et celui codé MN17 (Manhattan) est celui qui regroupe le plus grand nombre de restaurants avec 12195 restaurants.

Q3

```
GET ny_restaurants/_search
{
  "_source": ["VIOLATION CODE", "VIOLATION DESCRIPTION"],
  "query": {
    "term": {
      "VIOLATION CODE": "04N"
    }
  },
  "size": 1
}
```

On filtre ici les documents dont le champ *VIOLATION CODE* est égal à “04N” et renvoie la description associée.

Le paramètre `_source` limite les résultats aux champs à afficher, ici le code et sa description, et `size: 1` suffit puisqu’un seul exemple de ce code permet de connaître sa signification.


```
{
  "took": 5,
  "timed_out": false,
  "_shards": {
    "total": 1,
    "successful": 1,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 10000,
      "relation": "gte"
    },
    "max_score": 3.090052,
    "hits": [
      {
        "_index": "ny_restaurants",
        "_id": "j9q2L5oBqrQtmEC6nocT",
        "_score": 3.090052,
        "_source": {
          "VIOLATION CODE": "04N",
          "VIOLATION DESCRIPTION": "Filth flies or food/refuse/sewage associated
            with (FRSA) flies or other nuisance pests in establishment's food and/
            or non-food areas. FRSA flies include house flies, blow flies, bottle
            flies, flesh flies, drain flies, Phorid flies and fruit flies."
        }
      }
    ]
  }
}
```


On a donc le code “04N” qui correspond à la mention *“Filth flies or food/refuse/sewage associated with (FRSA) flies or other nuisance pests…”*, c’est-à-dire la présence de mouches ou d’autres nuisibles dans les zones alimentaires ou non alimentaires de l'établissement.

04

```
GET ny_restaurants/_search
{
  "_source": ["DBA","BORO","BUILDING","STREET","ZIPCODE","NTA","GRADE"],
  "query": {
    "term": { "GRADE": "A" }
  },
  "collapse": {
    "field": "CAMIS"
  },
  "sort": [
    { "NTA": { "order": "asc" } },
    { "CAMIS": { "order": "asc" } }
  ],
  "size": 10000,
  "track_total_hits": true
}
```

Le filtre `term` sur le champ GRADE sélectionne uniquement les établissements notés “A”.

Le paramètre `_source` limite les champs renvoyés aux informations principales : le nom du restaurant (*DBA*), son adresse (*BUILDING*, *STREET*, *ZIPCODE*), le borough (*BORO*) et le quartier (*NTA*). L’option `collapse` sur le champ *CAMIS* évite les doublons en ne gardant qu’un seul document par restaurant. Les résultats sont triés par quartier (*NTA*) puis par identifiant (*CAMIS*), et `track_total_hits` permet d’obtenir le nombre total d’établissements trouvés.

Le résultat ci-dessous indique qu’il y a environ 75 988 restaurants notés “A”.

À chaque fois, on affiche le nom du restaurant, son adresse complète et son quartier. Par exemple, on voit que “*FASCATI’S PIZZERIA*”, situé sur Henry Street à Brooklyn (*NTA* : BK09), apparaît en premier car on a mis qu’on affiche dans l’ordre alphabétique de leur *NTA* et de leur *CAMIS*.

```
{
  "took": 314,
  "timed_out": false,
  "_shards": {
    "total": 1,
    "successful": 1,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 75988,
      "relation": "eq"
    },
    "max_score": null,
    "hits": [
      {
        "_index": "ny_restaurants",
        "_id": "CUG2L5oBIftoHvtj0ioc",
        "_score": null,
        "_source": {
          "DBA": "FASCATI'S PIZZERIA",
          "BUILDING": "80",
          "BORO": "Brooklyn",
          "ZIPCODE": "11201",
          "STREET": "HENRY STREET",
          "GRADE": "A",
          "NTA": "BK09"
        },
        "fields": {
          "CAMIS": [
            40366652
          ]
        },
        "sort": [
          "BK09",
          40366652
        ]
      },
      {
        "_index": "ny_restaurants",
        "_id": "Atq2L5oBqrQtmEC6zePg",
        "_score": null,
        "_source": {
          "DBA": "MONTERO BAR & GRILL"
```


Q5

```
GET ny_restaurants/_search
{
  "size": 0,
  "aggs": {
    "cuisines": {
      "terms": {
        "field": "CUISINE DESCRIPTION",
        "size": 1,
        "order": { "unique_restaurants": "desc" }
      },
      "aggs": {
        "unique_restaurants": {
          "cardinality": { "field": "CAMIS"}
        }
      }
    }
  }
}
```

On effectue une agrégation sur le champ *CUISINE DESCRIPTION*, qui contient la catégorie culinaire de chaque restaurant (par exemple : *American*, *Chinese*, *Italian*, etc.). Pour éviter de compter plusieurs fois un même restaurant ayant plusieurs inspections, on utilise une

sous-agrégation *cardinality* sur le champ *CAMIS*, qui représente l'identifiant unique d'un établissement.

Le tri `order: { "unique_restaurants": "desc" }` classe les types de cuisine selon le nombre de restaurants uniques, et `size: 1` fait en sorte que seul le plus populaire soit renvoyé.


```
{
  "took": 85,
  "timed_out": false,
  "_shards": {
    "total": 1,
    "successful": 1,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 10000,
      "relation": "gte"
    },
    "max_score": null,
    "hits": []
  },
  "aggregations": {
    "cuisines": {
      "doc_count_error_upper_bound": -1,
      "sum_other_doc_count": 184606,
      "buckets": [
        {
          "key": "American",
          "doc_count": 36510,
          "unique_restaurants": {
            "value": 5045
          }
        }
      ]
    }
  }
}
```


Le résultat montre que la cuisine américaine (`"key": "American"`) est la plus représentée, avec environ 5 045 restaurants distincts sur un total de plus de 36 000 inspections liées à cette catégorie.

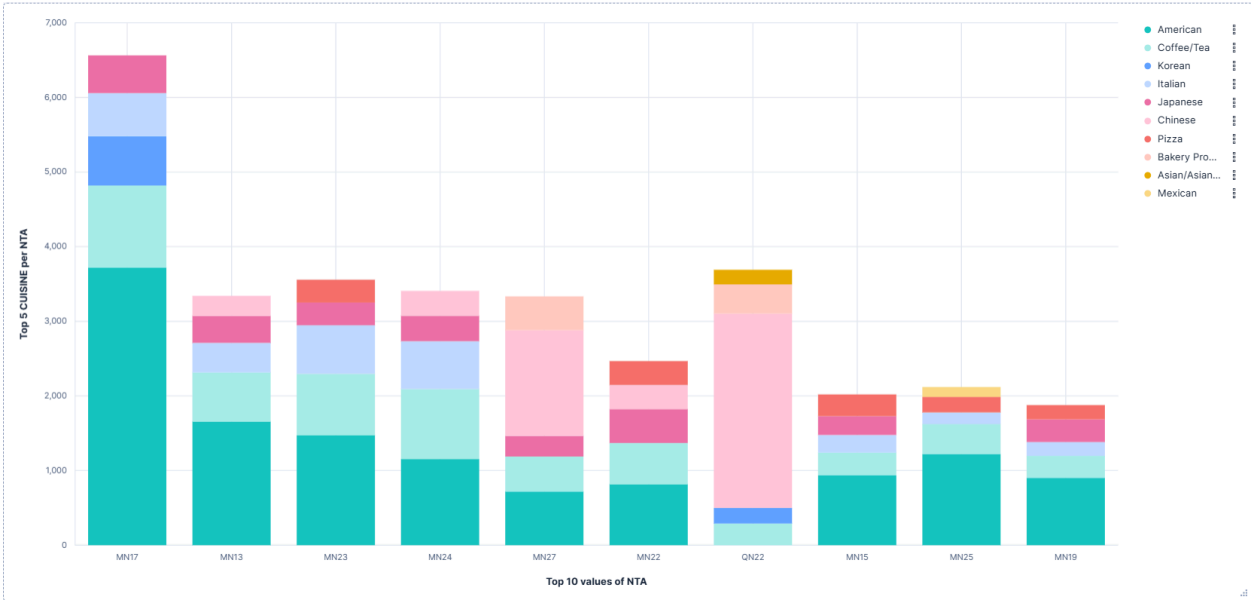
Pour répondre à la deuxième partie de la question (“et par quartier”), il suffirait d'ajouter une sous-agrégation *terms* sur le champ *NTA* pour compter les cuisines les plus populaires dans chaque quartier.

```
GET ny_restaurants/_search
{
  "size": 0,
  "aggs": {
    "by_nta": {
      "terms": {
        "field": "NTA",
        "size": 1000,
        "order": { "_key": "asc" }
      },
      "aggs": {
        "top_cuisine": {
          "terms": {
            "field": "CUISINE DESCRIPTION",
            "size": 1,
            "order": { "unique_restaurants": "desc" }
          },
          "aggs": {
            "unique_restaurants": {
              "cardinality": { "field": "CAMIS"}
            }
          }
        }
      }
    }
  }
}
```

Ici on regroupe d'abord les données par quartier (*NTA*), puis par type de cuisine (*CUISINE DESCRIPTION*).

Le résultat montre qu'en général la cuisine américaine est la plus présente dans plusieurs quartiers. Par exemple, dans le quartier BK09 à Brooklyn, elle apparaît comme la plus populaire avec 28 restaurants distincts avec la cuisine *American*.

Ci-dessous, on affiche les 5 cuisines les plus populaires dans 10 quartiers différents (*NTA*).



Q6

```
GET ny_restaurants/_search
{
  "size": 0,
  "aggs": {
    "last_inspection": {
      "max": {
        "field": "INSPECTION DATE",
        "format": "yyyy-MM-dd"
      }
    }
  }
}
```

Cette requête cherche la date la plus récente d'inspection en calculant le maximum sur le champ *INSPECTION DATE*.

On ne renvoie pas de documents, seulement l'agrégation `last_inspection`.

Le paramètre `"format": "yyyy-MM-dd"` force un affichage clair de la date, pour ne pas confondre DD-MM-YYYY et MM-DD-YYYY.

Dans le résultat ci-dessous, la valeur lisible est `aggregations.last_inspection.value_as_string = "2024-04-02"` (2 avril 2024) : c'est donc la dernière date d'inspection du dataset.


```
{
  "took": 0,
  "timed_out": false,
  "_shards": {
    "total": 1,
    "successful": 1,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 10000,
      "relation": "gte"
    },
    "max_score": null,
    "hits": []
  },
  "aggregations": {
    "last_inspection": {
      "value": 1712016000000,
      "value_as_string": "2024-04-02"
    }
  }
}
```

Q7

```
GET ny_restaurants/_search
{
  "_source": ["DBA","BUILDING","STREET","BORO","ZIPCODE","NTA","GRADE","CUISINE DESCRIPTION"],
  "track_total_hits": true,
  "query": {
    "bool": {
      "must": [
        { "term": { "BORO": "Brooklyn" } },
        { "term": { "GRADE": "A" } },
        { "term": { "CUISINE DESCRIPTION": "Chinese" } }
      ]
    }
  },
  "collapse": { "field": "CAMIS" },
  "aggs": {
    "distinct_restaurants": {
      "cardinality": { "field": "CAMIS"}
    }
  },
  "sort": [
    { "NTA":   { "order": "asc" } },
    { "CAMIS": { "order": "asc" } }
  ],
  "size": 10000
}
```

Cette requête extrait la liste des restaurants chinois notés A à Brooklyn.

- Le filtre `bool.must` restreint aux documents où *BORO* = "Brooklyn", *GRADE* = "A" et *CUISINE DESCRIPTION* = "*Chinese*".
- `collapse` sur *CAMIS* évite les doublons (un seul hit par restaurant).
- `_source` ne renvoie que les champs utiles (nom, adresse, quartier, grade, type de cuisine).
- Le tri se fait par quartier (*NTA*) puis par identifiant (*CAMIS*) pour un ordre stable.
- L'agg `distinct_restaurants` (cardinality sur *CAMIS*) donne le nombre de restaurants uniques correspondant.
- `track_total_hits: true` fournit le total de documents qui matchent (avant déduplication).

```
{
  "took": 17,
  "timed_out": false,
  "_shards": {
    "total": 1,
    "successful": 1,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 1883,
      "relation": "eq"
    },
    "max_score": null,
    "hits": [
      {
        "_index": "ny_restaurants",
        "_id": "UkC2L5oBIftoHvtjpcv2",
        "_score": null,
        "_source": {
          "DBA": "LICHEE NUT",
          "BUILDING": "162",
          "BORO": "Brooklyn",
          "CUISINE DESCRIPTION": "Chinese",
          "ZIPCODE": "11201",
          "STREET": "MONTAGUE STREET",
          "GRADE": "A",
          "NTA": "BK09"
        },
        "fields": {
          "CAMIS": [
            40605862
          ]
        },
        "sort": [
          "BK09",
          40605862
        ]
      },
      {
        "_index": "ny_restaurants",
        "_id": "Ytq2L5oBqrQtmEC63_Ud",
        "_score": null,
        "_source": {
```


Dans le résultat montré, `hits.total.value = 1883` indique 1883 documents correspondant au filtre, et la liste affiche des établissements comme “LICHEE NUT” (BK09, Brooklyn, A) avec leur adresse complète. L’agrégation `distinct_restaurants.value` est la valeur à reporter si on veut le compte de restos uniques.

Q8

```
GET ny_restaurants/_search
{
  "_source": ["DBA","BUILDING","STREET","BORO","ZIPCODE","NTA","CAMIS"],
  "query": {
    "match": {
      "DBA": {
        "query": "LADUREE",
        "operator": "and"
      }
    }
  },
  "aggs": {
    "distinct_restaurants": {
      "cardinality": { "field": "CAMIS" }
    }
  },
  "collapse": { "field": "CAMIS" },
  "size": 100
}
```

Une requête de type `match` a été utilisée afin de retrouver tous les documents correspondant au nom *LADUREE*.

L'opérateur `"and"` a été ajouté pour rendre la recherche plus stricte : il garantit que tous les mots saisis doivent être présents dans le nom du restaurant, évitant ainsi les correspondances partielles.

Les résultats ont été limités aux champs liés à l'adresse (*BUILDING, STREET, ZIPCODE, BORO, NTA*) et dédupliqués grâce au `collapse` sur *CAMIS*, afin de ne conserver qu'un enregistrement par établissement.

Le résultat ci-dessous montre 35 documents (plusieurs inspections) mais 2 restaurants distincts :

- LADUREE — 864 MADISON AVENUE, Manhattan 10021 (*NTA* : MN40)
- LADUREE SOHO — 398 WEST BROADWAY, Manhattan 10012 (*NTA* : MN24)

```
{
  "took": 4,
  "timed_out": false,
  "_shards": {
    "total": 1,
    "successful": 1,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 35,
      "relation": "eq"
    },
    "max_score": 11.699377,
    "hits": [
      {
        "_index": "ny_restaurants",
        "_id": "9Nq2L5oBqrQtmEC6rZHp",
        "_score": 11.699377,
        "_source": {
          "DBA": "LADUREE",
          "BUILDING": "864",
          "BORO": "Manhattan",
          "CAMIS": 50054046,
          "ZIPCODE": "10021",
          "STREET": "MADISON AVENUE",
          "NTA": "MN40"
        },
        "fields": {
          "CAMIS": [
            50054046
          ]
        }
      },
      {
        "_index": "ny_restaurants",
        "_id": "o9q2L5oBqrQtmEC6rZHo",
        "_score": 9.67333,
        "_source": {
          "DBA": "LADUREE SOHO",
          "BUILDING": "398",
          "BORO": "Manhattan",
          "CAMIS": 50006574,
          "ZIPCODE": "10012",
          "STREET": "WEST BROADWAY",
          "NTA": "MN24"
        },
        "fields": {
          "CAMIS": [
            50006574
          ]
        }
      }
    ]
  },
  "aggregations": {
    "distinct_restaurants": {
      "value": 2
    }
  }
}
```

Q9

```
GET ny_restaurants/_search
{
  "size": 0,
  "query": {
    "match_phrase": {
      "VIOLATION DESCRIPTION": "Hot TCS food item not held at or above 140 °F."
    }
  },
  "aggs": {
    "by_cuisine": {
      "terms": {
        "field": "CUISINE DESCRIPTION",
        "size": 1,
        "order": { "affected_restaurants": "desc" }
      },
      "aggs": {
        "affected_restaurants": {
          "cardinality": { "field": "CAMIS" }
        }
      }
    }
  }
}
```

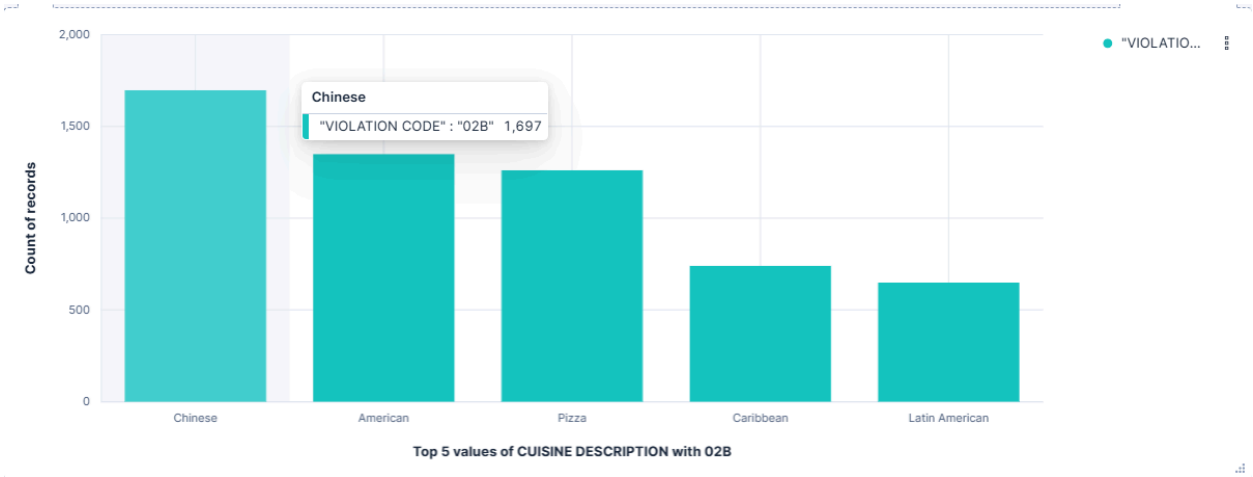
Il faut d'abord commencer par repérer la phrase exacte du problème dans les données de *VIOLATION DESCRIPTION* : "*Hot TCS food item not held at or above 140°F.*"

Le champ contenant une phrase complète, la solution la plus précise a été d'utiliser `match_phrase` plutôt qu'un simple `term`, afin de cibler uniquement les enregistrements contenant cette description exacte. Ensuite, une agrégation sur *CUISINE DESCRIPTION* a permis de compter le nombre de restaurants distincts (*CAMIS*) concernés par cette violation, et d'en déduire le type de cuisine le plus touché.

Le résultat ci-dessous montre que la cuisine chinoise est celle la plus fréquemment associée à cette violation.

```
{
  "took": 32,
  "timed_out": false,
  "_shards": {
    "total": 1,
    "successful": 1,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 7877,
      "relation": "eq"
    },
    "max_score": null,
    "hits": []
  },
  "aggregations": {
    "by_cuisine": {
      "doc_count_error_upper_bound": -1,
      "sum_other_doc_count": 6721,
      "buckets": [
        {
          "key": "Chinese",
          "doc_count": 1156,
          "affected_restaurants": {
            "value": 903
          }
        }
      ]
    }
  }
}
```

On a remarqué que ce *CODE VIOLATION*, correspond à 02B et grâce à ceci on affiche ci-dessous le graphique correspondant.



Q10

```
GET ny_restaurants/_search
{
  "size": 0,
  "aggs": {
    "top_violations": {
      "terms": {
        "field": "VIOLATION CODE",
        "size": 5,
        "order": { "_count": "desc" }
      },
      "aggs": {
        "example_description": {
          "top_hits": {
            "_source": ["VIOLATION CODE", "VIOLATION DESCRIPTION"],
            "size": 1
          }
        },
        "affected_restaurants": {
          "cardinality": { "field": "CAMIS" }
        }
      }
    }
  }
}
```

Pour trouver le Top 5 des violations, l'idée a été de regrouper par identifiant de violation plutôt que par libellé.

Le champ *VIOLATION DESCRIPTION* est textuel et peut varier (phrases longues, ponctuation), donc pas fiable pour un classement. Le champ *VIOLATION CODE* est, lui, normalisé (*keyword*) et identifie chaque type de violation de façon stable.

La requête fait donc un classement par fréquence des codes (les 5 plus fréquents). Pour rendre le résultat lisible, une sous-requête `top_hits` récupère une description d'exemple pour chaque code. Et pour mesurer l'impact, une cardinality sur *CAMIS* donne le nombre de restaurants distincts touchés par chaque code.

Interprétation rapide des résultats fournis ci-dessous :

- 10F — *Non-food contact surfaces…* → 30 893 occurrences, 19 311 restos distincts
- 08A — *Not free of conditions conducive to pests* → 23 539, 14 001
- 06D — *Food contact surface not properly washed/rinsed/sanitized* → 14 849, 11 021
- 04L — *Evidence of mice…* → 14 465, 9 190
- 02G — *Cold TCS food above required temps…* → 13 170, 9 945

Ce choix (agrégé par code) garantit un top réellement comparable, puis on ajoute la description pour l'explication et le nombre de *CAMIS* uniques pour l'ampleur réelle.


```
{
  "took": 125,
  "timed_out": false,
  "_shards": {
    "total": 1,
    "successful": 1,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 10000,
      "relation": "gte"
    },
    "max_score": null,
    "hits": []
  },
  "aggregations": {
    "top_violations": {
      "doc_count_error_upper_bound": 0,
      "sum_other_doc_count": 123052,
      "buckets": [
        {
          "key": "10F",
          "doc_count": 30893,
          "example_description": {
            "hits": {
              "total": {
                "value": 30893,
                "relation": "eq"
              },
              "max_score": 1,
              "hits": [
                {
                  "_index": "ny_restaurants",
                  "_id": "htq2L5oBqrQtmEC6nocT",
                  "_score": 1,
                  "_source": {
                    "VIOLATION CODE": "10F",
                    "VIOLATION DESCRIPTION": "Non-food contact surface improperly constructed. Unacceptable material used. Non-food contact surface or equipment improperly maintained and/or
                        not properly sealed, raised, spaced or movable to allow accessibility for cleaning on all sides, above and underneath the unit."
                  }
                }
```

```
            ]
          }
        },
        "affected_restaurants": {
          "value": 19311
        }
      },
      {
        "key": "08A",
        "doc_count": 23539,
        "example_description": {
          "hits": {
            "total": {
              "value": 23539,
              "relation": "eq"
            },
            "max_score": 1,
            "hits": [
              {
                "_index": "ny_restaurants",
                "_id": "hdq2L5oBqrQtmEC6nocT",
                "_score": 1,
                "_source": {
                  "VIOLATION CODE": "08A",
                  "VIOLATION DESCRIPTION": "Establishment is not free of harborage or conditions conducive to rodents, insects or other pests."
                }
              }
            ]
          }
        },
        "affected_restaurants": {
          "value": 14001
        }
      },
      {
        "key": "06D",
        "doc_count": 14849,
        "example_description": {
          "hits": {
            "total": {
              "value": 14849,
              "relation": "eq"
            },
            "max_score": 1,
            "hits": [
              {
                "_index": "ny_restaurants",
                "_id": "19q2L5oBqrQtmEC6nocT",
                "_score": 1,
                "_source": {
                  "VIOLATION CODE": "06D",
                  "VIOLATION DESCRIPTION": "Food contact surface not properly washed, rinsed and sanitized after each use and following any activity when contamination may have occurred."
                }
              }
            ]
          }
        },
        "affected_restaurants": {
          "value": 11021
        }
      },
      {
        "key": "04L",
        "doc_count": 14465,
        "example_description": {
          "hits": {
            "total": {
              "value": 14465,
              "relation": "eq"
            },
            "max_score": 1,
            "hits": [
              {
                "_index": "ny_restaurants",
                "_id": "g9q2L5oBqrQtmEC6nocT",
                "_score": 1,
                "_source": {
                  "VIOLATION CODE": "04L",
                  "VIOLATION DESCRIPTION": "Evidence of mice or live mice in establishment's food or non-food areas."
                }
              }
            ]
          }
        },
        "affected_restaurants": {
          "value": 9190
        }
```

```
      },
      {
        "key": "02G",
        "doc_count": 13170,
        "example_description": {
          "hits": {
            "total": {
              "value": 13170,
              "relation": "eq"
            },
            "max_score": 1,
            "hits": [
              {
                "_index": "ny_restaurants",
                "_id": "mNq2L5oBqrQtmEC6nocT",
                "_score": 1,
                "_source": {
                  "VIOLATION CODE": "02G",
                  "VIOLATION DESCRIPTION": "Cold TCS food item held above 41 °F; smoked or processed fish held above 38 °F; intact raw eggs held above 45 °F; or reduced oxygen packaged (ROP) TCS foods held above required temperatures except during active necessary preparation."
                }
              }
            ]
          }
        },
        "affected_restaurants": {
          "value": 9945
        }
      }
    ]
  }
}
}
```


II. Dashboard et MAP

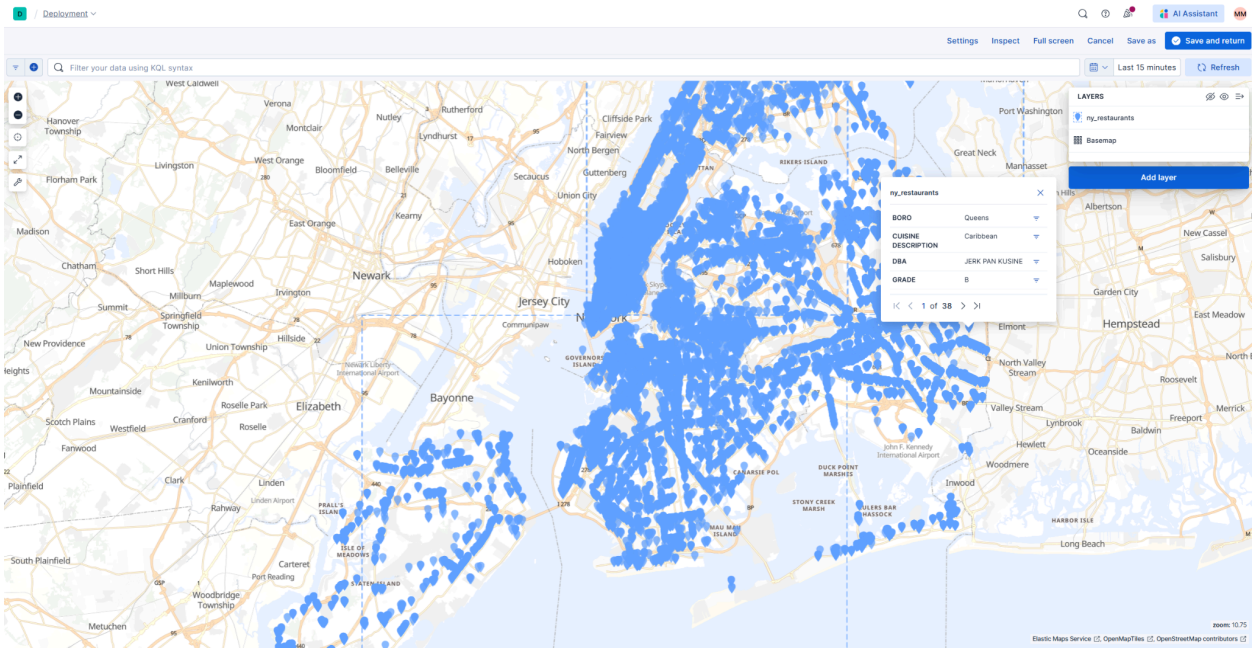
Pour pouvoir créer un *Dahsboard*, on a dû ajouter la latitude et la longitude en créant un champ *location* pour que la *MAP* puisse lire les coordonnées car on avait seulement Latitude et Longitude en `double`, donc Kibana ne proposait aucun “geo field” et affichait l’erreur “Data view does not contain any geospatial fields”.

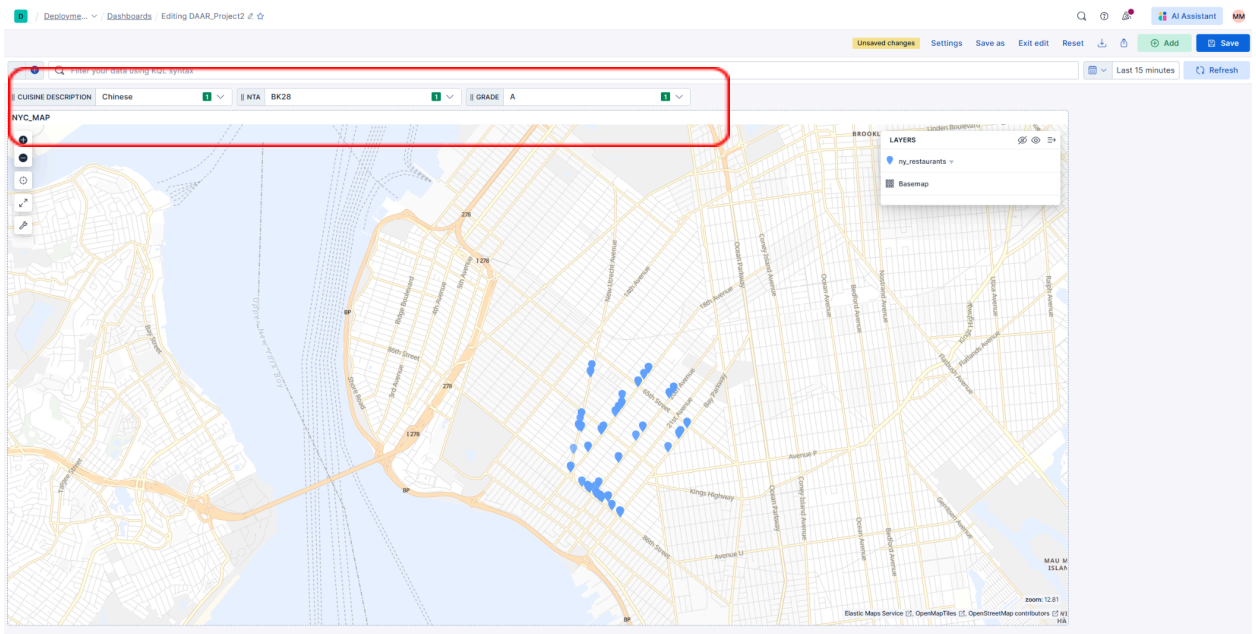
La démarche a été la suivante :

- Partir des champs existants `Latitude` / `Longitude` et créer un champ géo dédié : ajouter `location` au mapping en `geo_point`.
- Renseigner ce nouveau champ pour tous les documents en le construisant à partir des valeurs existantes, au format attendu [lon, lat]. Pour éviter de réindexer le jeu de données, utiliser un `_update_by_query` avec un script painless qui copie `Longitude` et `Latitude` vers `location`.
- Rafraîchir la data view dans Kibana, puis, dans Maps, choisir `location` comme Geospatial field. La couche “Documents” peut alors s’afficher en points sur la carte.

C’est pour cette raison qu’on a exécuté les requêtes suivantes :

```
PUT ny_restaurants/_mapping
{
  "properties": {
    "location": { "type": "geo_point" }
  }
}
POST ny_restaurants/_update_by_query
{
  "script": {
    "source": """
      if (ctx._source.containsKey('Latitude') && ctx.
        _source.containsKey('Longitude')) {
        double lat = ctx._source.Latitude;
        double lon = ctx._source.Longitude;
        ctx._source.location = [lon, lat];
      }
    """,
    "lang": "painless"
  }
}
```





On peut voir dans la capture d'écran ci-dessus, qu'on affiche les restaurants chinois avec une note "A" dans le quartier BK28 grâce aux filtres qu'on a ajoutés.