



به نام خدا

دانشکده مهندسی کامپیوتر

گروه مهندسی نرم افزار

پروژه پایانی

طراحی یک سامانه بلادرنگ داده‌های تاکسی اینترنتی

درس

کلان داده و تحلیل داده‌های حجیم

استاد مربوطه

دکتر محمدعلی نعمت‌بخش

تهیه شده توسط

عطیه نیکبخت

تیرماه ۱۴۰۱

مقدمه

در این گزارش به پیاده‌سازی گام چهارم و پنجم پروژه پرداخته می‌شود. نحوه نصب کافکا، الستیکسرچ، کینبا، Logstash و کاساندر را بیان می‌شود. اتصالات مربوطه نیز شرح داده می‌شود. لینک گیت‌هاب:

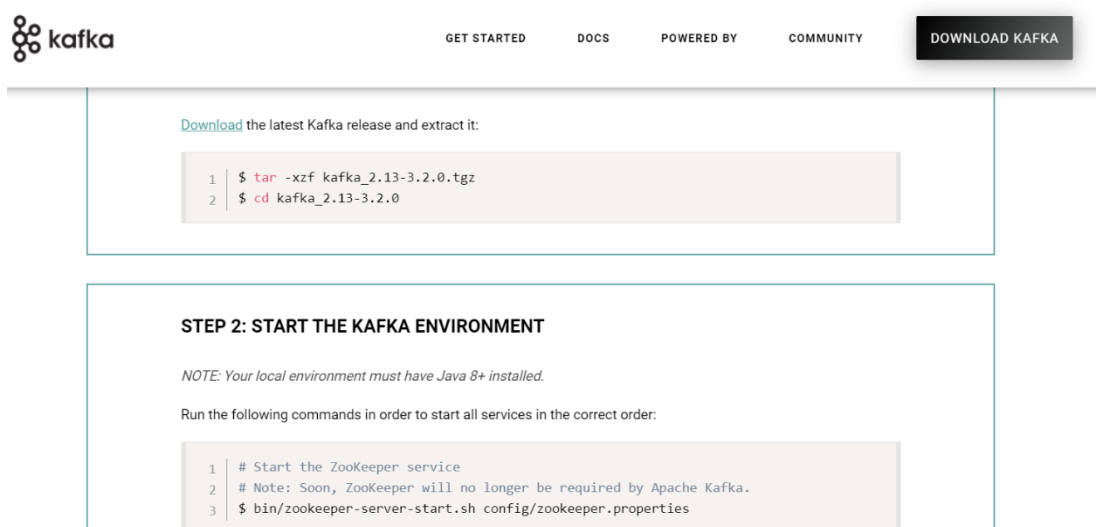
<https://github.com/AtiyehNikbakht/FinalPro.git>

۱- نصب برنامه‌ها

در این بخش به نحوه نصب هر یک از برنامه‌های مورد نیاز پرداخته و موارد مورد نیاز جهت نصب هر برنامه توضیح داده می‌شود.

۱-۱- نصب کافکا

از کافکا جهت تعامل قسمت‌های مختلف پروژه استفاده شده است. ابتدا کافکا را طبق شکل ۱ از آدرس <https://kafka.apache.org/quickstart> دانلود می‌کنیم و آن را پس از حالت فشرده در آوردن در مسیر مورد دلخواه قرار می‌دهیم. در این پیاده‌سازی تمامی برنامه‌ها در درایو C و پوشه مربوط به خود قرار دارد.



شکل ۱- دانلود برنامه کافکا

سپس به پوشه Config رفته و ابتدا فایل `zookeeper.properties` را باز می‌کنیم و مقدار فیلد مشخص شده در شکل ۲ را برابر با مسیر نصب کافکا و پوشه مشخص شده قرار می‌دهیم. سپس فایل `server.properties` را نیز باز کرده و مسیر فیلد مشخص شده در شکل ۳ را برابر با مسیر نصب کافکا و پوشه مشخص شده قرار می‌دهیم. ورژن جاوای ۱۸ برای استفاده از کافکا نصب می‌شود.

```

-
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or impli
# See the License for the specific language governing permissions and
# limitations under the License.
# the directory where the snapshot is stored.
dataDir=C:\kafka\zookeeper-data
# the port at which the clients will connect
clientPort=2181
# disable the per-ip limit on the number of connections since this is a
maxClientCnxns=0
# Disable the adminserver by default to avoid port conflicts.
# Set the port to something non-conflicting if choosing to enable this
admin.enableServer=false
# admin.serverPort=8080

```

شکل ۲- تغییر مسیر اولیه به مسیر مورد نظر

```

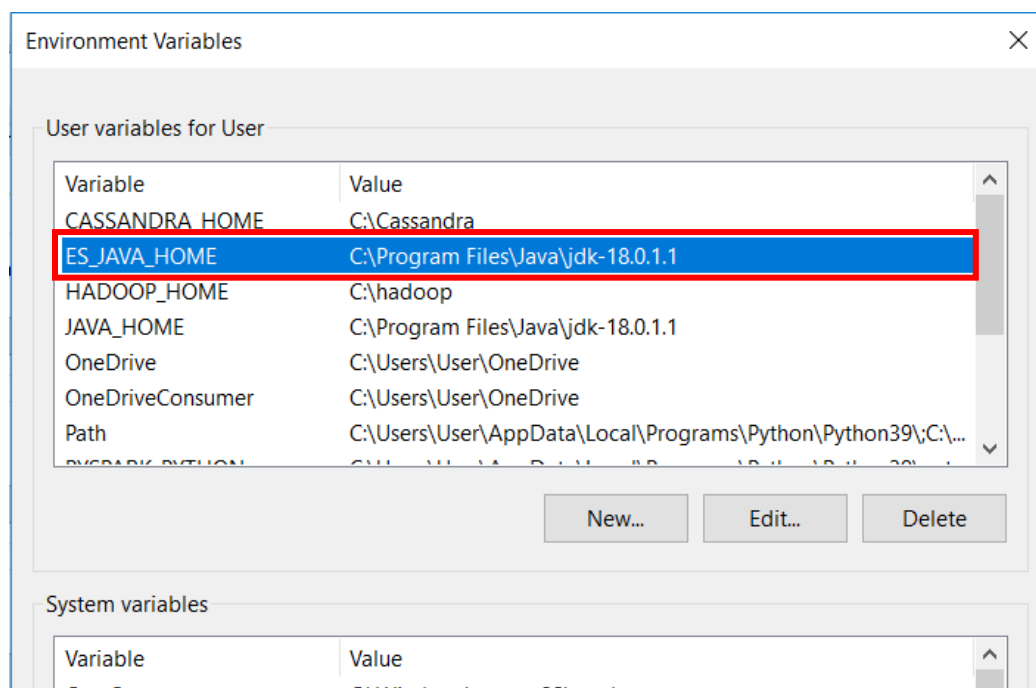
##### Log Basics #####
# A comma separated list of directories under which to store log files
log.dirs=C:\kafka\kafka-logs
# The default number of log partitions per topic. More partitions allow
# parallelism for consumption, but this will also result in more files a
# the brokers.
num.partitions=1
# The number of threads per data directory to be used for log recovery a
# This value is recommended to be increased for installations with data
num.recovery.threads.per.data.dir=1

```

شکل ۳- تغییر مسیر اولیه به مسیر مورد نظر

۲-۱- نصب الستیک سرچ، کیبانا، LogStash

برای دانلود الستیک سرچ، کیبانا و Logstash از آدرس <https://www.elastic.co/downloads> استفاده می کنیم. نسخه ۸.۱ این سه برنامه را دانلود کرده و آن را در پوشه دلخواه قرار می دهیم. همان طور که گفته شد در این پیاده سازی تمامی برنامه ها در درایو C و پوشه مربوط به خود قرار دارد. سپس همان طور که در شکل ۴ مشاهده می کنید یک متغیر به نام ES_JAVA_HOME ساخته و سپس مقدار آن را برابر با مسیر جاوای نسخه ۱۸ قرار می دهیم. این سه برنامه از این جاوا در اجرای خود استفاده می کنند.



شکل ۴- تنظیم مسیر ES_JAVA_HOME

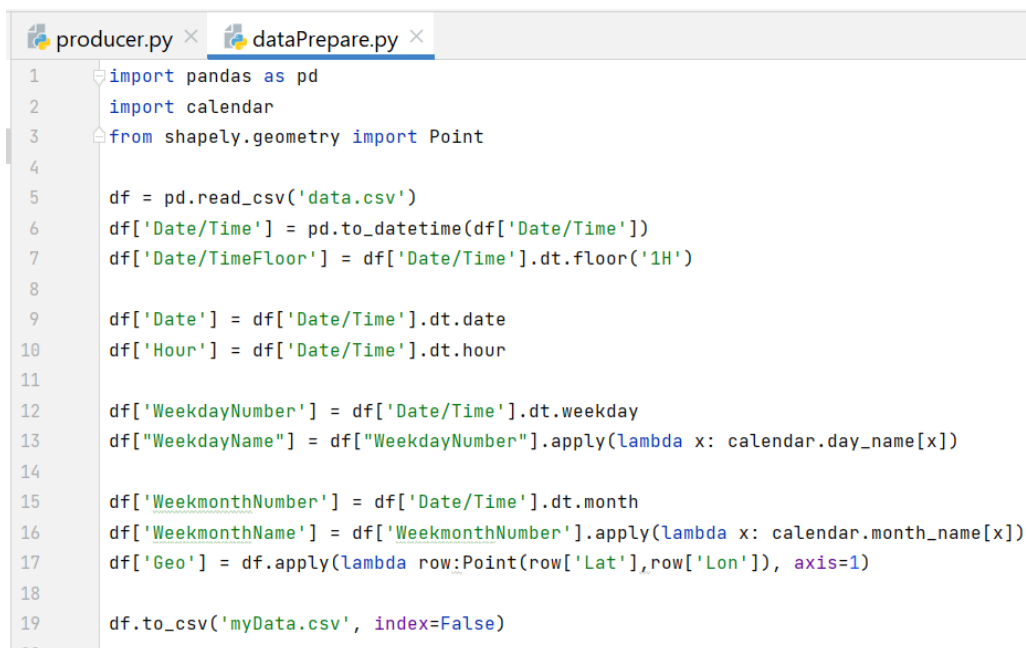
۳-۱- نصب کاساندرا

جهت نصب کاساندرا در ویندوز از آدرس <https://archive.apache.org/dist/cassandra/3.11.4> نسخه ۳.۱۱.۴ را دانلود و پس از استخراج آن از حالت فشرده، آن را در درایو C قرار می‌دهیم. نسخه جاوای مورد استفاده 1.8.0_231 است. بنابراین هنگام کار با کاساندرا در Environment Variables مسیر جاوا را تغییر داده و مسیر جاوای ذکر شده را ذخیره می‌کنیم. جهت کار کردن با cqlsh در کاساندرا باید از نسخه ۲.۷ پایتون استفاده کنیم. بنابراین این نسخه از پایتون را نیز نصب کرده و در Environment Variables در قسمت Path مربوط به System Variables مسیر این نسخه از پایتون را در ابتدای لیست درج می‌کنیم (تا نسخه‌های دیگر پایتون که در سیستم نصب شده‌اند را نادیده گرفته و پایتون ۲.۷ را استفاده کند).

۲- تغییر داده‌ها

قبل از استفاده از داده‌ها جهت ترسیم داشبورد، چند فیلد به داده‌ها اضافه می‌کنیم (انجام این بخش ضرورتی ندارد اما جهت کنترل بیشتر در هنگام ترسیم داشبورد این بخش نیز انجام شده است). بنابراین از برنامه شکل ۵ استفاده شده است. ابتدا فایل داده را خوانده و سپس فیلد Data/Time را آن را به datetime تبدیل می‌کنیم. سپس جهت اینکه

مقدار ساعات را به صورت کف نیز ذخیره کنیم به عنوان مثال به جای ۰۱:۰۰ و ۰۱:۰۴ و ... را مقدار ۰۱:۰۰ را ذخیره کنیم از دستور `dt.floor` استفاده می کنیم. سپس تاریخ، ساعت، شماره روز هفته، نام روز، شماره ماه، نام ماه و نقطه جغرافیایی شامل `Lat` و `Lon` را همان طور که در شکل ۵ مشاهده می کنید در یک فایل به نام `myData.csv` ذخیره می کنیم و در ادامه از این فایل به عنوان داده ها استفاده می کنیم.



```

1 import pandas as pd
2 import calendar
3 from shapely.geometry import Point
4
5 df = pd.read_csv('data.csv')
6 df['Date/Time'] = pd.to_datetime(df['Date/Time'])
7 df['Date/TimeFloor'] = df['Date/Time'].dt.floor('1H')
8
9 df['Date'] = df['Date/Time'].dt.date
10 df['Hour'] = df['Date/Time'].dt.hour
11
12 df['WeekdayNumber'] = df['Date/Time'].dt.weekday
13 df["WeekdayName"] = df["WeekdayNumber"].apply(lambda x: calendar.day_name[x])
14
15 df['WeekmonthNumber'] = df['Date/Time'].dt.month
16 df['WeekmonthName'] = df['WeekmonthNumber'].apply(lambda x: calendar.month_name[x])
17 df['Geo'] = df.apply(lambda row: Point(row['Lat'], row['Lon']), axis=1)
18
19 df.to_csv('myData.csv', index=False)

```

شکل ۵- برنامه `dataPrepare`

۳- اتصال الستیک سرچ به کافکا

جهت انتقال اطلاعات از کافکا به الستیک سرچ از برنامه `producer` و تنظیمات `Logstash` استفاده می کنیم. برنامه `producer` کمک می کند تا داده ها به کافکا فرستاده شوند و با استفاده از `Logstash` داده ها را دریافت کرده و در الستیک سرچ استفاده می کنیم. در ابتدا برنامه `producer` و تنظیمات `Logstash` را بررسی می کنیم و سپس نحوه اجرا را نیز شرح می دهیم.

در شکل ۶ برنامه `producer` را مشاهده می کنید. از کتابخانه `KafkaProducer` جهت ایجاد یک `producer` استفاده می کنیم. از کتابخانه `csv` نیز جهت خواندن یک فایل با پسوند `csv` استفاده می کنیم. سپس در تعریف `producer` مقدار پارامتر `bootstrap_servers` را برابر با آدرس محلی کافکا (که در سیستم خودمان اجرا می شود) قرار می دهیم. سپس فایل `csv` را خوانده و مقادیر آن را به صورت یک لیست در `lineList` قرار می دهیم و سطر اول که حاوی اسم ستون ها می باشد را حذف می کنیم. سپس با استفاده از تابع `join` هر خط لیست را که خود یک لیست است را به یک متن که با علامت «`,`» از هم جدا می شوند، تبدیل می کنیم. سپس آن را `utf-8` کرده و با استفاده از دستور `send` به کافکا انتقال می دهیم. پارامتر

اول این تابع، نام یک topic تعریف شده در کافکا است. در بخش ۴ نحوه تعریف این topic توضیح داده می‌شود. هر خط از داده پس از ۵ ثانیه منتقل می‌شود.

```

producer.py x dataPrepare.py x
1 from kafka import KafkaProducer
2 import time
3 import json
4 from csv import reader
5
6 producer = KafkaProducer(bootstrap_servers=['localhost:9092'])
7
8 if __name__ == '__main__':
9     with open('myData.csv', 'r') as read_obj:
10         csv_dict_reader = reader(read_obj)
11         lineList = list(csv_dict_reader)
12         del lineList[0]
13         for i in lineList:
14             data = ','.join(map(str, i))
15             print(data)
16             producer.send('testTwo', data.encode('utf-8'))
17             time.sleep(5)

```

شکل ۶- برنامه producer

جهت دریافت اطلاعات کافکا از طریق Logstash باید فایل Logstash.conf را ایجاد کرده و متناسب با کار خود آن را مقداردهی کرده و در پوشه Config مربوط به Logstash آن را ذخیره کنیم. در شکل ۷ این تنظیمات را مشاهده می‌کنید.

```

input {
  kafka {
    bootstrap_servers => ["localhost:9092"]
    topics => ["testTwo"]
  }
}

filter {
  csv {
    separator => ",",
    columns => ["Data\Time", "Lat", "Lon", "Base", "Date/TimeFloor", "Date", "Hour", "WeekdayNumber", "WeekdayName", "WeekmonthNumber", "WeekmonthName", "Geo"]
  }

  mutate {convert => ["Lat", "float"]}
  mutate {convert => ["Lon", "float"]}
  mutate {convert => ["Hour", "integer"]}
  mutate {convert => ["WeekdayNumber", "integer"]}
  mutate {convert => ["WeekmonthNumber", "integer"]}
}

output {
  elasticsearch {
    hosts => ["http://localhost:9200"]
    index => "testtwo"
  }
  stdout {}
}

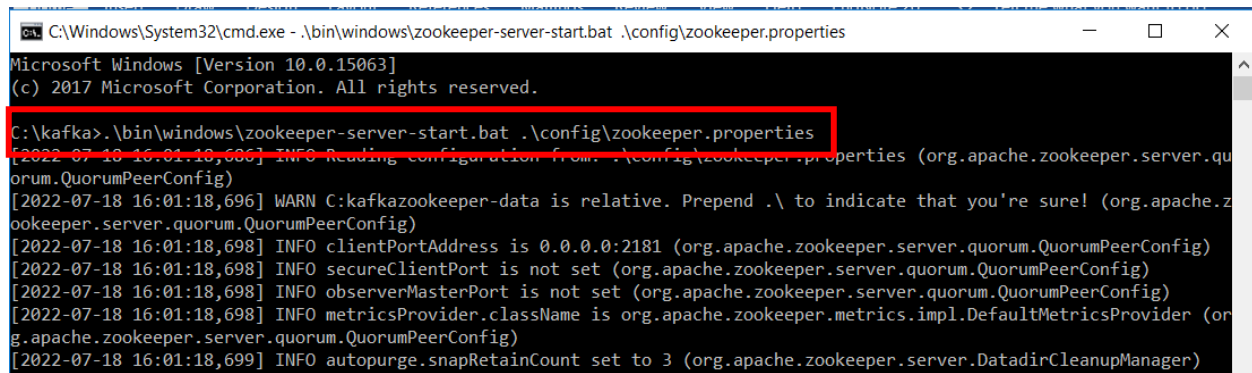
```

شکل ۷- تنظیمات Logstash

همان طور که در شکل ۷ مشاهده می کنید مقدار input نشان می دهد که داده های ورودی از کافکا و با آدرس محلی گفته شده و نام topic مورد نظر دریافت می شوند. جهت دریافت ورودی به صورتی که هر فیلد موجود در فایل داده قابل دسترسی باشد و فقط در متغیر message نباشد از filter استفاده می کنیم. در filter هر خط از داده با «،» از هم جدا شده و در ستون مربوط به خود قرار می گیرد. سپس نوع ستون ها مشخص می شود. در صورتی که نوع ستون text باشد نیازی به نوشتن آن نیست اما بقیه موارد مثل float و int نوشته می شوند. سپس در output محل دریافت داده ها را مشخص می کنیم که برابر با elasticsearch است و در آدرس محلی ۹۲۰۰ قرار دارد همچنین نام index را می توان به صورت دلخواه انتخاب کرد. در اینجا testtwo انتخاب شده است. در بخش بعد اجرای کافکا و ساخت topic و همچنین داده های دریافت شده در الاستیک سرچ را شرح می دهیم.

۴- دریافت داده ها

ابتدا جهت اجرای کافکا در مسیر نصب آن رفته و cmd را باز می کنیم. سپس دستور مشخص شده در شکل ۸ را وارد می کنیم.

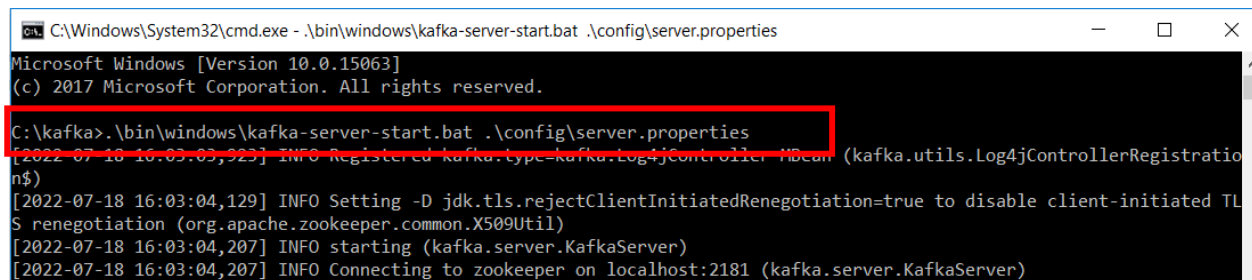


```
C:\Windows\System32\cmd.exe - .\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\kafka>.bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties
[2022-07-18 16:01:18,686] INFO Reading configuration from: .\config\zookeeper.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-07-18 16:01:18,696] WARN C:\kafkazookeeper-data is relative. Prepend .\ to indicate that you're sure! (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-07-18 16:01:18,698] INFO clientPortAddress is 0.0.0.0:2181 (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-07-18 16:01:18,698] INFO secureClientPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-07-18 16:01:18,698] INFO observerMasterPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-07-18 16:01:18,698] INFO metricsProvider.className is org.apache.zookeeper.metrics.impl.DefaultMetricsProvider (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2022-07-18 16:01:18,699] INFO autopurge.snapRetainCount set to 3 (org.apache.zookeeper.server.DataDirCleanupManager)
```

شکل ۸- دستور اجرای zookeeper

سپس دستور یک cmd دیگر در همان مسیر نصب کافکا باز می کنیم. دستور مشخص شده در شکل ۹ را در آن وارد می کنیم.



```
C:\Windows\System32\cmd.exe - .\bin\windows\kafka-server-start.bat .\config\server.properties
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\kafka>.bin\windows\kafka-server-start.bat .\config\server.properties
[2022-07-18 16:03:03,323] INFO Registered kafka.type=kafka.Log4jController-MBean (kafka.utils.Log4jControllerRegistration)
[2022-07-18 16:03:04,129] INFO Setting -D jdk.tls.rejectClientInitiatedRenegotiation=true to disable client-initiated TLS renegotiation (org.apache.zookeeper.common.X509Util)
[2022-07-18 16:03:04,207] INFO starting (kafka.server.KafkaServer)
[2022-07-18 16:03:04,207] INFO Connecting to zookeeper on localhost:2181 (kafka.server.KafkaServer)
```

شکل ۹- دستور اجرای کافکا

حال کافکا در حال اجرا است. بنابراین می توان topic مورد نظر را بسازیم. یک cmd دیگر در مسیر نصب کافکا باز کرده و از دستور مشخص شده در شکل ۱۰ استفاده می کنیم.

```
C:\kafka>.\bin\windows\kafka-topics.bat --create --topic testTwo --bootstrap-server localhost:9092
Created topic testTwo.
```

شکل ۱۰- ساخت topic

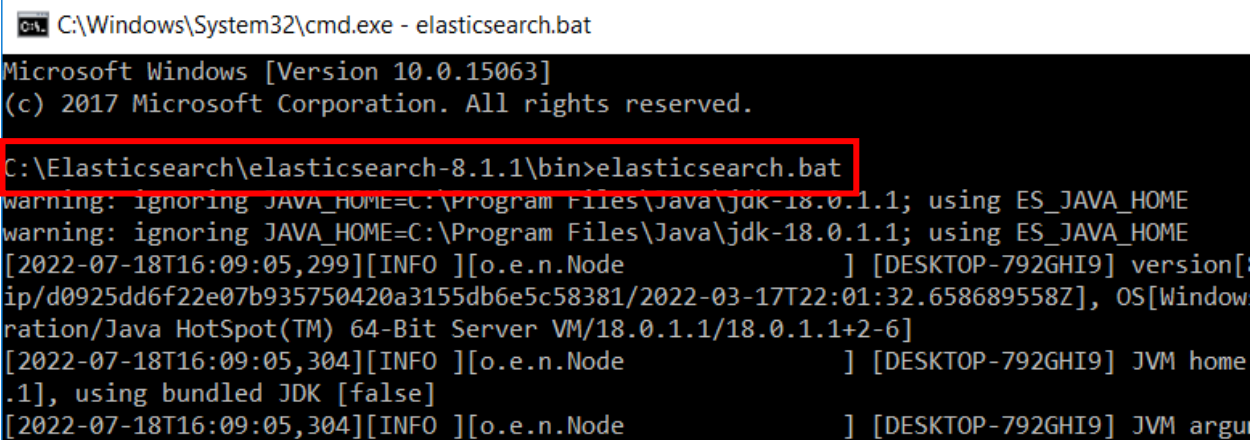
سپس فایل producer گرفته شده در بخش قبل را جهت انتقال داده ها اجرا می کنیم. در شکل ۱۱ اجرای و داده های انتقالی را مشاهده می کنید.



```
Run: producer x
C:\Users\User\AppData\Local\Programs\Python\Python39\python.exe "D:/My Project/testConnect/producer.py"
2014-09-01 00:01:00,40.2201,-74.0021,B02512,2014-09-01 00:00:00,2014-09-01,0,0,Monday,9,September,POINT (40.2201 -74.0021)
2014-09-01 00:01:00,40.75,-74.0027,B02512,2014-09-01 00:00:00,2014-09-01,0,0,Monday,9,September,POINT (40.75 -74.0027)
2014-09-01 00:03:00,40.7559,-73.9864,B02512,2014-09-01 00:00:00,2014-09-01,0,0,Monday,9,September,POINT (40.7559 -73.9864)
2014-09-01 00:06:00,40.745,-73.9889,B02512,2014-09-01 00:00:00,2014-09-01,0,0,Monday,9,September,POINT (40.745 -73.9889)
2014-09-01 00:11:00,40.8145,-73.9444,B02512,2014-09-01 00:00:00,2014-09-01,0,0,Monday,9,September,POINT (40.8145 -73.9444)
2014-09-01 00:12:00,40.6735,-73.9918,B02512,2014-09-01 00:00:00,2014-09-01,0,0,Monday,9,September,POINT (40.6735 -73.9918)
2014-09-01 00:15:00,40.7471,-73.6472,B02512,2014-09-01 00:00:00,2014-09-01,0,0,Monday,9,September,POINT (40.7471 -73.6472)
```

شکل ۱۱- اجرای برنامه و داده های انتقالی

در مرحله بعد ابتدا الستیک سرچ، کیبانا و Logstash را اجرا می کنیم. بنابراین به محل نصب هر کدام رفته و یک cmd برای هر کدام باز می کنیم. سپس دستورات مشخص شده در شکل ۱۲، ۱۳ را جهت اجرای الستیک و کیبانا وارد می کنیم. جهت اجرای Logstash دستور bin\logstash -f .\config\logstash.conf را در cmd خودش اجرا می کنیم.



```
C:\Windows\System32\cmd.exe - elasticsearch.bat
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Elasticsearch\elasticsearch-8.1.1\bin>elasticsearch.bat
warning: ignoring JAVA_HOME=C:\Program Files\Java\jdk-18.0.1.1; using ES_JAVA_HOME
warning: ignoring JAVA_HOME=C:\Program Files\Java\jdk-18.0.1.1; using ES_JAVA_HOME
[2022-07-18T16:09:05,299][INFO ][o.e.n.Node               ] [DESKTOP-792GHI9] version[
ip/d0925dd6f22e07b935750420a3155db6e5c58381/2022-03-17T22:01:32.658689558Z], OS[Window
ration/Java HotSpot(TM) 64-Bit Server VM/18.0.1.1/18.0.1.1+2-6]
[2022-07-18T16:09:05,304][INFO ][o.e.n.Node               ] [DESKTOP-792GHI9] JVM home
.1], using bundled JDK [false]
[2022-07-18T16:09:05,304][INFO ][o.e.n.Node               ] [DESKTOP-792GHI9] JVM argu
```


شکل ۱۲- اجرای الستیک

```
C:\Windows\System32\cmd.exe - kibana.bat
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Elasticsearch\kibana-8.1.1\bin>kibana.bat
[2022-07-18T16:11:02.109+04:30][INFO ][plugins-service] Plugin "metricsEntities" is disabled.
[2022-07-18T16:11:02.175+04:30][INFO ][http.server.Preboot] http server running at http://localhost:5601
[2022-07-18T16:11:02.231+04:30][INFO ][plugins-system.preboot] Setting up [1] plugins: [interacti
[2022-07-18T16:11:02.238+04:30][INFO ][preboot] "interactiveSetup" plugin is holding setup: Valid
nection configuration...
[2022-07-18T16:11:02.287+04:30][INFO ][root] Holding setup until preboot stage is completed.
[2022-07-18T16:11:02.374+04:30][WARN ][config.deprecation] The default mechanism for Reporting p
erently in future versions, which will affect the behavior of this cluster. Set "xpack.reporting
e" to adopt the future behavior before upgrading
```

شکل ۱۳- اجرای کیبانا

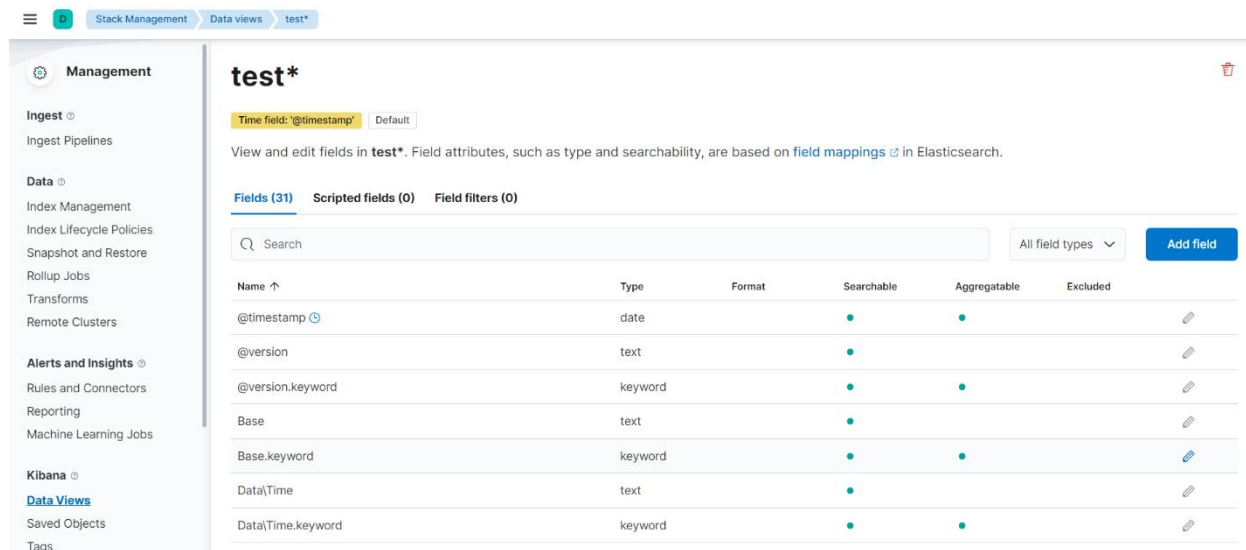
پس از اتصال Logstash داده‌ها همان‌طور که در شکل ۱۴ مشاهده می‌کنید وارد می‌شوند.

```
"Date/TimeFloor" => "2014-09-03 09:00:00",
"WeekdayNumber" => 2
}
{
  "Geo" => "POINT (40.7452 -73.9936)",
  "event" => {
    "original" => "2014-09-03 09:44:00,40.7452,-73.9936,B02512,2014-09-03 09:00:00,2014-09-03,9,2,Wednesday,9,Septem
ber,POINT (40.7452 -73.9936)"
  },
  "message" => "2014-09-03 09:44:00,40.7452,-73.9936,B02512,2014-09-03 09:00:00,2014-09-03,9,2,Wednesday,9,Sep
tember,POINT (40.7452 -73.9936)",
  "Data\\Time" => "2014-09-03 09:44:00",
  "WeekdayName" => "Wednesday",
  "Lon" => -73.9936,
  "Hour" => 9,
  "Lat" => 40.7452,
  "@version" => "1",
  "WeekmonthNumber" => 9,
  "@timestamp" => 2022-07-18T22:51:29.786584200Z,
  "Base" => "B02512",
  "WeekmonthName" => "September",
  "Date" => "2014-09-03",
  "Date/TimeFloor" => "2014-09-03 09:00:00",
  "WeekdayNumber" => 2
}
```

شکل ۱۴- دریافت داده در Logstash

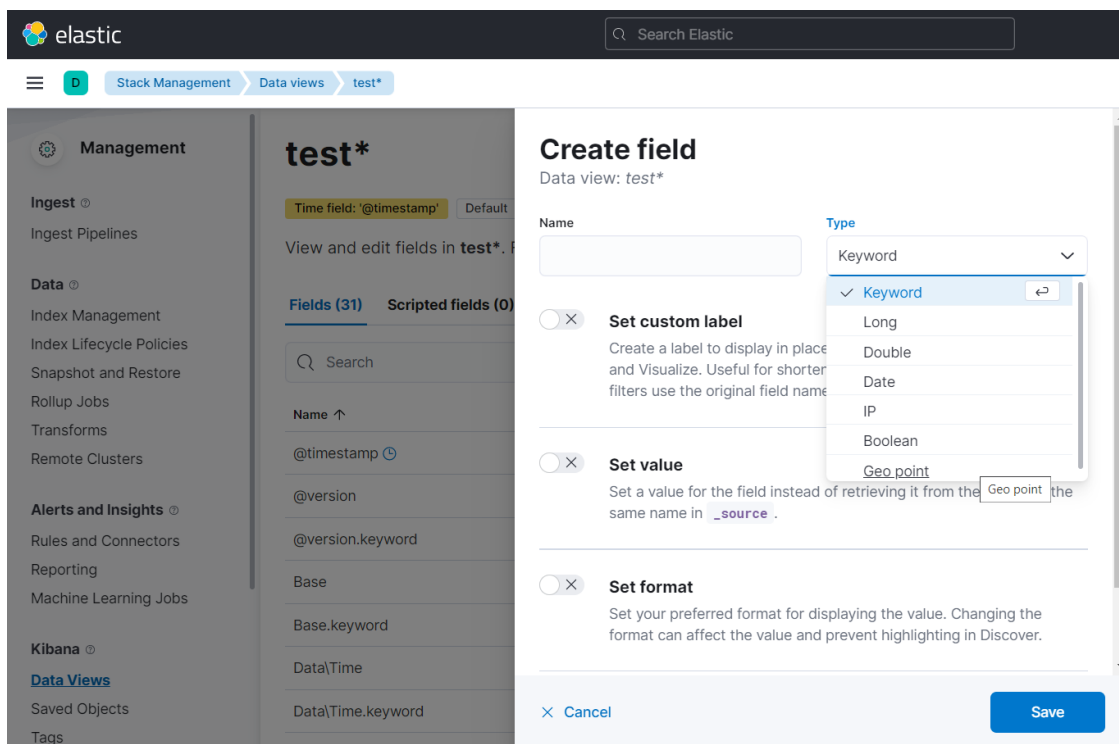
اکنون به آدرس محلی <http://localhost:5601> می‌رویم. در این آدرس الستیک و کیبانا به‌صورت محلی در دسترس هستند. سپس جهت دریافت داده‌هایی که Logstash در حال دریافت آن است ابتدا یک Data View می‌سازیم. نام آن

را test گذاشته و index pattern آن را منبع شاخص (index) testtwo قرار می‌دهیم. در شکل ۱۵ Data View ساخته شده با نام test را مشاهده می‌کنید.



شکل ۱۵- ساخت Data View

سپس از گزینه Add field یک فیلد geo_point می‌سازیم. فیلد Geo موجود در مجموعه داده دریافتی از نوع text است. نیاز به نقاطی داریم که فیلد آن از نوع geo باشد. در شکل ۱۶ و ۱۷ ساخت و مقداردهی این فیلد جدید را مشاهده می‌کنید.



شکل ۱۶- ساخت فیلد جدید geo

Define script

```
1 emit(doc['Lat'].value, doc['Lon'].value)
```

Runtime fields without a script retrieve values from `_source`. If the field doesn't exist in `_source`, a search request returns no value. [Learn about script syntax.](#)

☒ **Set format**

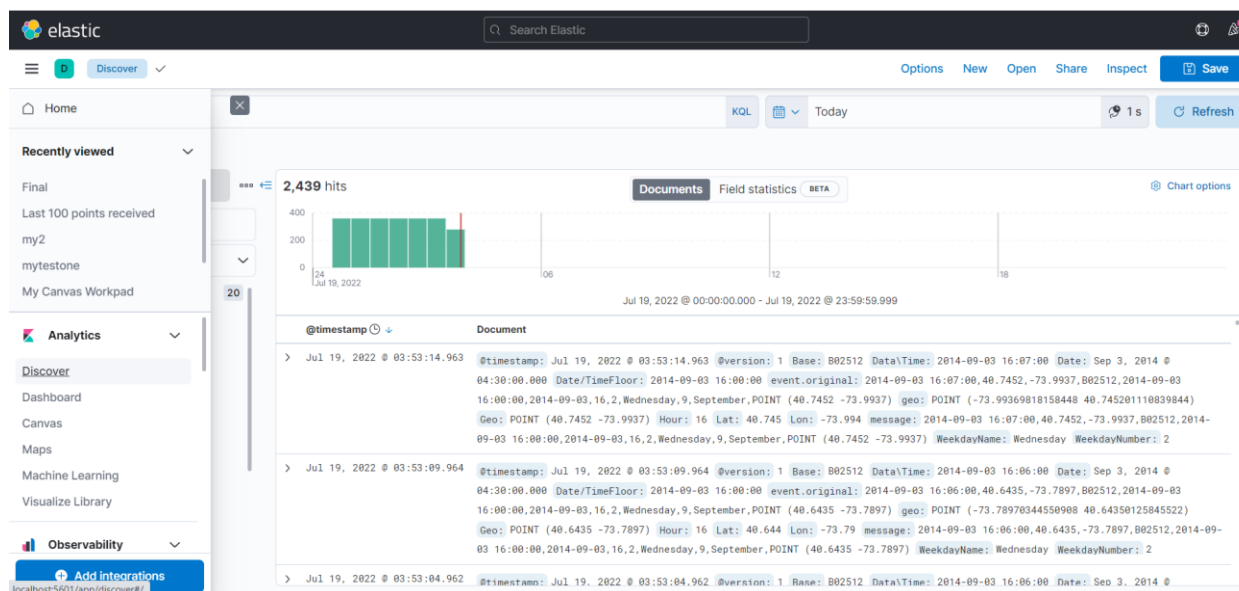
Set your preferred format for displaying the value. Changing the format can affect the value and prevent highlighting in Discover.

Format (Default: `Geo point`)

Geo point

شکل ۱۷- مقداردهی فیلد geo

سپس در Discover می‌توانیم داده‌هایی که در حال دریافت است را مشاهده کنیم. در شکل ۱۸ این مورد را مشاهده می‌کنید.



شکل ۱۸- مشاهده داده‌ها در Discover

در بخش بعد برای کوئری‌های گفته شده، نمودار رسم می‌کنیم.

۵- طراحی داشبورد

در این بخش متناسب با هر کوئری یک نوع نمودار رسم می‌شود. در ادامه نحوه رسم نمودارها توضیح داده می‌شود.

New visualization

Lens
 Create visualizations with our drag and drop editor. Switch between visualization types at any time. *Recommended for most users.*

Maps
 Create and style maps with multiple layers and indices.

TSVB
 Perform advanced analysis of your time series data.

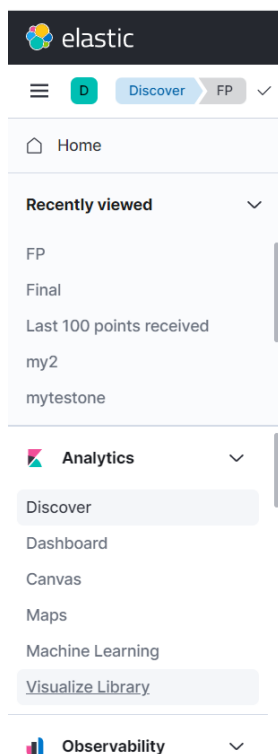
Custom visualization
 Use Vega to create new types of visualizations. *Requires knowledge of Vega syntax.*

Aggregation based
 Use our classic visualize library to create charts based on aggregations.

Tools
Text
 Add text and images to your dashboard.

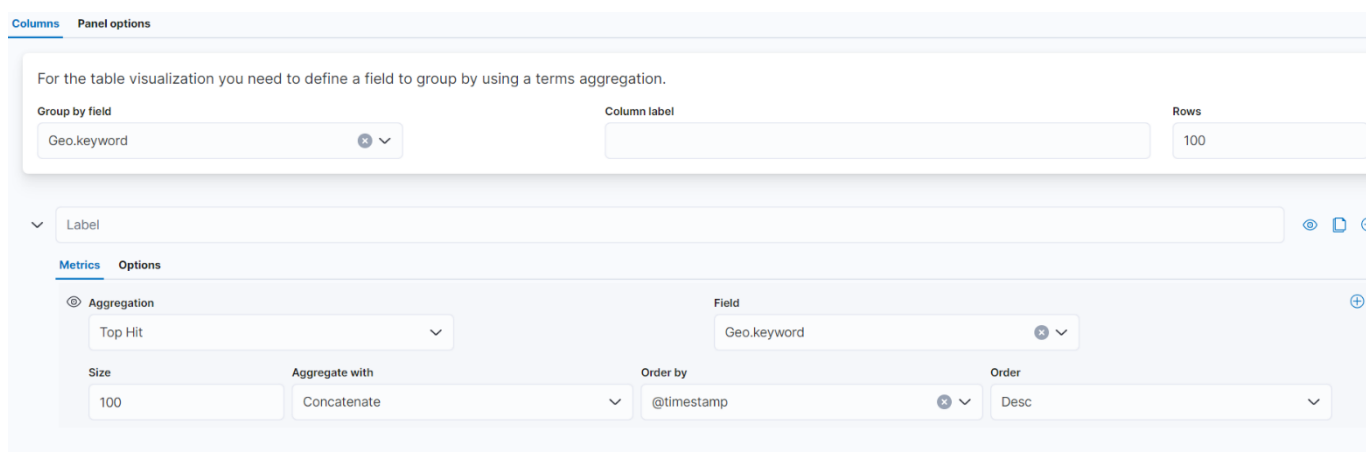
شکل ۱۹- انواع نمودارها

- ۱۰۰ نقطه‌ای آخر دریافتی از استریم: جهت رسم این نمودار ابتدا به بخش Visualiza Library نشان داده شده در شکل ۲۰ رفته و TSVB را انتخاب می‌کنیم.



شکل ۲۰- رسم نمودار

سپس در سربرگ Table رفته و گزینه‌های مشخص شده در تصویر ۲۱ را تنظیم می‌کنیم.



شکل ۲۱- تنظیم فیلدها

براساس فیلد Geo (فیلدی که در مجموعه داده بود) گروه‌بندی را انجام داده و ۱۰۰ نقطه آخر دریافت شده را براساس timestamp ورودی آن‌ها نمایش می‌دهیم. در شکل ۲۲ مقدار Data View و Last Value تنظیم شده را مشاهده می‌کنید.

Data

Item url

This supports mustache templating. `{{key}}` is set to the term.

Data timerange mode ⓘ

Last value

Data view

test*

Time field

@timestamp

Interval

5m

Examples: auto, 1m, 1d, 7d, 1y, >=1m

Drop last bucket?

☐ Yes ☒ No

Panel filter

Search

KQL

Ignore global filter?

☐ Yes ☒ No

شکل ۲۲- تنظیم فیلدها

در شکل ۲۳، ۱۰۰ نمودار نقطه آخر دریافتی را بعد هر ۵ دقیقه را در داشبورد مشاهده می‌کنید.

Last 100 points received		Last value
Geo.keyword ↑	Top Hit of Geo.keyword	
POINT (40.6448 -73.782)	-	
POINT (40.7739 -73.87260000000001)	-	
POINT (40.6447 -73.7822)	-	
POINT (40.6447 -73.7824)	-	
POINT (40.6448 -73.7821)	-	
POINT (40.6448 -73.7822)	-	
POINT (40.6459 -73.77670000000001)	-	
POINT (40.6878 -74.1816)	-	
POINT (40.688 -74.1828)	-	
POINT (40.695 -74.1782)	-	
POINT (40.7421 -73.9967)	-	

شکل ۲۳- جدول ۱۰۰ نقطه آخر دریافتی

- تعداد درخواست‌ها برای ۱۰ نقطه‌ای پر تردد: جهت رسم نمودار این بخش به Lens می‌رویم. از نمودار میله‌ای افقی استفاده می‌کنیم. در شکل ۲۴ تنظیمات مربوط به محور افقی و عمودی را مشاهده می‌کنید.

Bar horizontal

test*

Vertical axis

Optional

10 busy Point

Horizontal axis

Count of records

Add or drag-and-drop a field

Break down by

Optional

Add or drag-and-drop a field

Vertical axis

Functions

Date histogram

Filters

Intervals

Top values

Fields

Geo.keyword

Add field

Number of values

10

Rank by

Count of records

Rank direction

Ascending

Descending

Close

Horizontal axis

Quick functions

Formula

Functions

Average

Count

Counter rate

Cumulative sum

Differences

Last value

Maximum

Median

Minimum

Moving average

Percentile

Sum

Unique count

Field

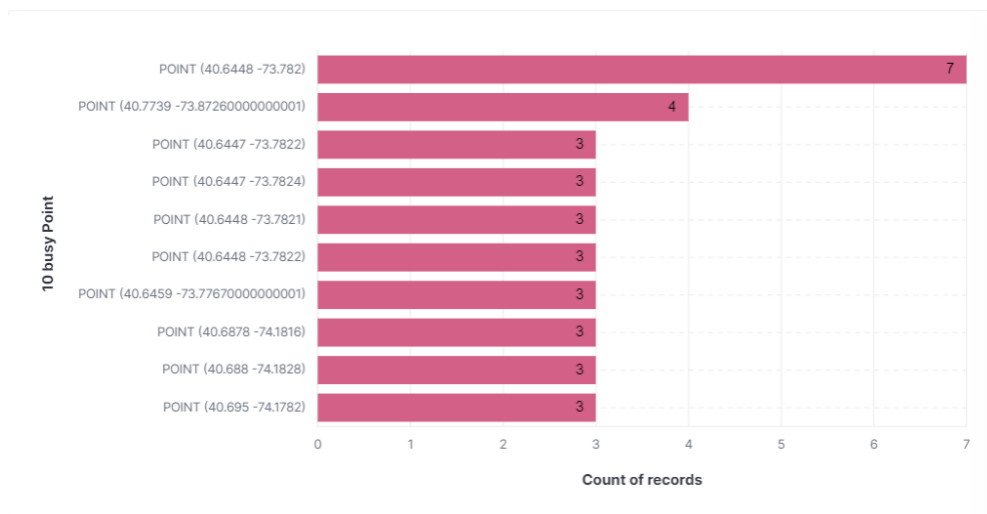
Records

Add advanced options

Close

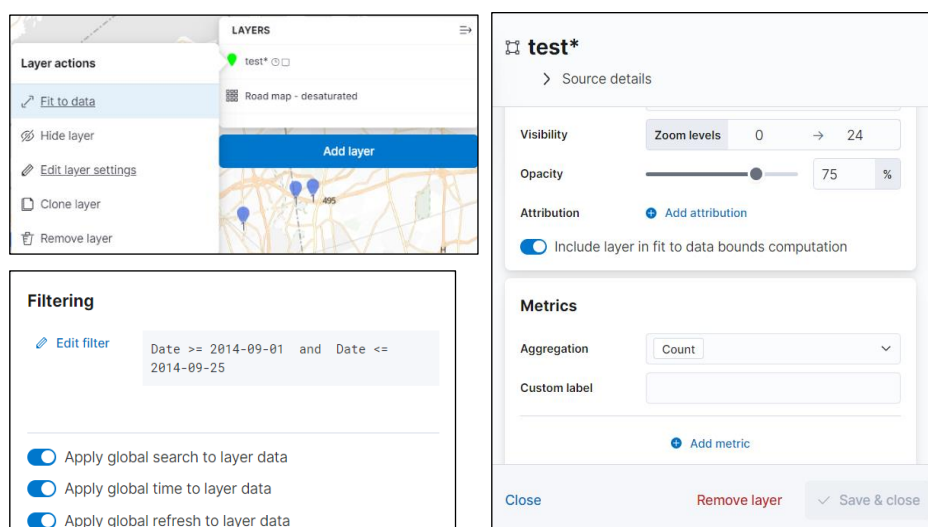
شکل ۲۴- تنظیمات محور افقی و عمودی

براین اساس محور افقی تعداد درخواست هایی را نشان می دهد که ۱۰ نقطه پرتدد دارند و محور عمودی نیز ۱۰ نقطه با بیشترین درخواست را نشان می دهد. در شکل ۲۵ نمودار این بخش را مشاهده می کنید. این نمودار را نیز به داشبورد خود اضافه می کنیم.



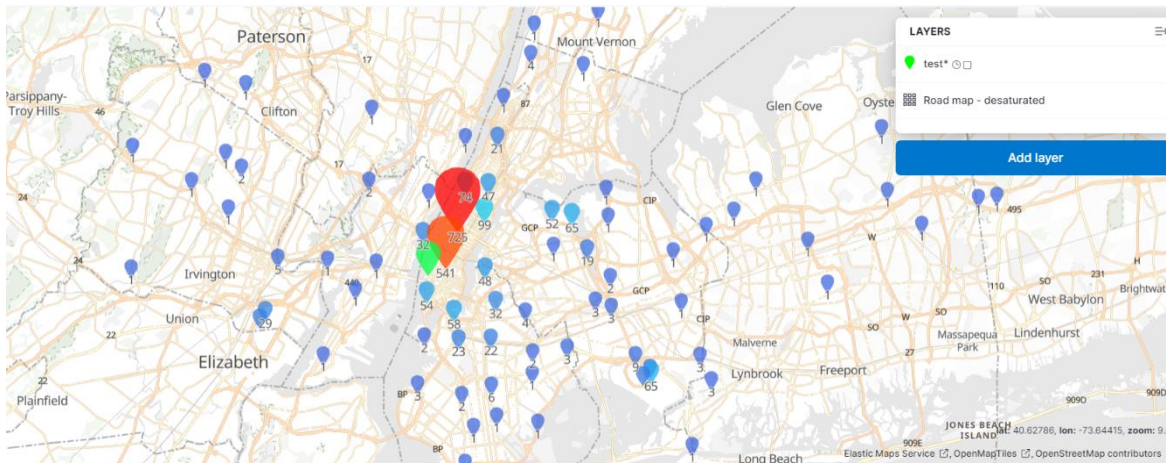
شکل ۲۵- نمودار درخواست ها برای ۱۰ نقطه

- تراکم نقاط شروع پرتدد در یک بازه زمانی خاص: نمودار این بخش با نوع Maps ساخته شده است. ابتدا به این بخش می رویم و یک لایه جدید می سازیم. لایه جدید از Data view ساخته شده یعنی test استفاده می کند. هم چنین برای رسم نقاط از فیلد geo ساخته شده در الستیک استفاده می شود. با استفاده از تنظیماتی که در شکل ۲۶ مشاهده می کنید نقاط درخواست دهنده در نقشه مشخص شده است هم چنین تعداد درخواست ها برای نقاط نیز مشخص می شود.



شکل ۲۶- تنظیمات نمودار تراکم نقاط

جهت نمایش تراکم نقاط در یک بازه زمانی خاص از filtering استفاده می‌کنیم. در اینجا براساس فیلد Date فیلتر را انجام می‌دهیم. بنابراین تراکم نقاط درخواست‌هایی که بین ۱ سپتامبر و ۲۵ سپتامبر هستند را نمایش می‌دهد. در شکل ۲۷ نقشه تراکم نقاط را مشاهده می‌کنید.



شکل ۲۷- نقشه تراکم نقاط

- ۱۰ نقطه‌ای پرتدد در یک بازه زمانی: جهت رسم این نمودار TSVB را انتخاب می‌کنیم. سپس به بخش Table رفته و تنظیمات شکل ۲۸ را جهت پیدا کردن ۱۰ نقطه پرتدد انجام می‌دهیم.

Columns Panel options

For the table visualization you need to define a field to group by using a terms aggregation.

Group by field

Geo.keyword

Column label

Rows

10

Label

Metrics Options

Aggregation

Count

Metrics Options

Data formatter

Default

Template

{{value}}

eg. {{value}}/s

Filter

Date/TimeFloor.keyword >= "2014-09-02 00:00" and Date/TimeFloor.keyword <= "2014-09-03 23:00"

KQL

Show trend arrows?

Yes No

Data

Item url

This supports mustache templating. {{key}} is set to the term.

Data timerange mode

Entire time range

Data view

test*

Time field

@timestamp

Interval

auto

Drop last bucket?

Yes No

Examples: auto, 1m, 1d, 7d, 1y, >=1m

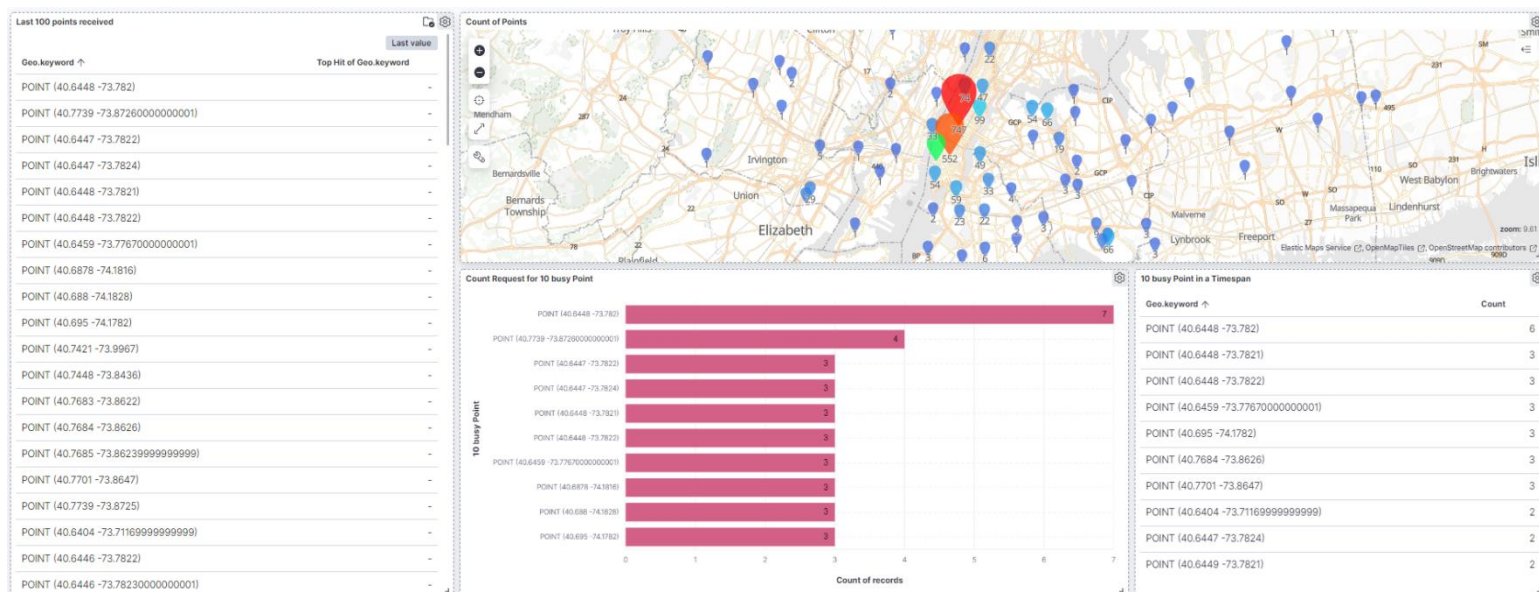
شکل ۲۸- تنظیمات ۱۰ نقطه پرتدد

جهت مشخص کردن بازه زمانی داده‌ها از فیلد Filtering استفاده می‌شود. همچنین از فیلد Date/TimeFloor استفاده می‌کنیم. بنابراین ۱۰ نقطه‌ای که از ۲ سپتامبر ساعت ۱۲ شب تا ۳ سپتامبر ساعت ۱۱ شب پرتدد بوده‌اند را نمایش می‌دهد. جدول این نقاط در شکل ۲۹ نمایش داده شده است.

Geo.keyword ↑	Count
POINT (40.6448 -73.782)	6
POINT (40.6448 -73.7821)	3
POINT (40.6448 -73.7822)	3
POINT (40.6459 -73.77670000000001)	3
POINT (40.695 -74.1782)	3
POINT (40.7684 -73.8626)	3
POINT (40.7701 -73.8647)	3
POINT (40.6404 -73.71169999999999)	2
POINT (40.6447 -73.7824)	2
POINT (40.6449 -73.7821)	2

شکل ۲۹- جدول ۱۰ نقطه پرتدد

در نهایت داشبورد تشکیل شده از کوئری‌های فوق را در شکل ۳۰ مشاهده می‌کنید.



شکل ۳۰- داشبورد

۵- تغییر داده‌ها جهت استفاده در کاساندرا

با استفاده از دستورات شکل زیر به داده‌های مرحله قبل شماره هفته و زمان‌بندی ۱۲ ساعته را اضافه می‌کنیم.

```
import pandas as pd

df = pd.read_csv('myData.csv')

df['Date/Time'] = pd.to_datetime(df['Date/Time'])
df['WeekNumber'] = df['Date/Time'].dt.strftime('%U')
df['Time'] = df['Date/Time'].dt.floor('12H')
df['Time12'] = df['Time'].dt.time

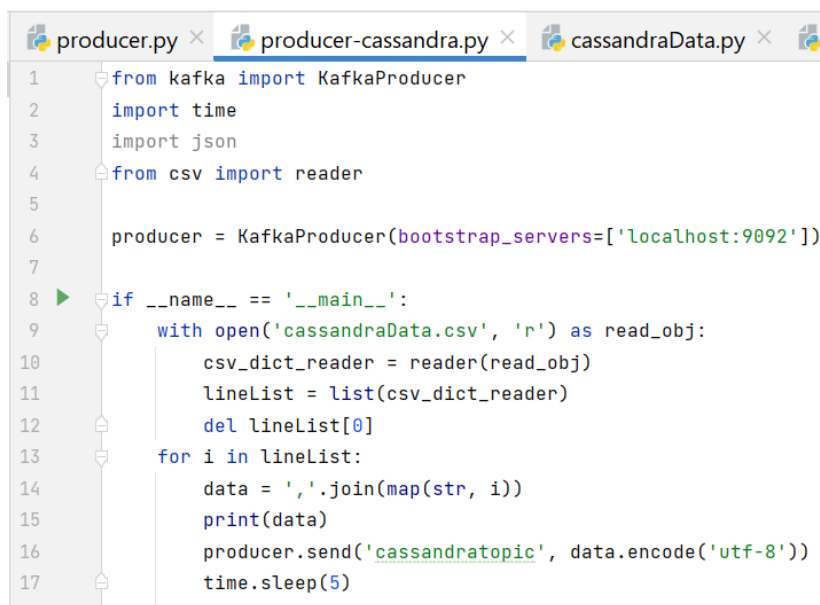
df.to_csv('cassandraData.csv', index=False)
```

۶- انتقال داده از کافکا به کاساندرا

سپس جهت انتقال اطلاعات از کافکا به کاساندرا یک topic جدید ساخته و نام آن را cassandratopic می‌گذاریم.

```
C:\kafka>. \bin\windows\kafka-topics.bat --create --topic cassandratopic --bootstrap-server localhost:9092
Created topic cassandratopic.
```

سپس یک producer جدید به نام producer-cassandra جهت انتقال اطلاعات جدید به کافکا می‌سازیم. کد این برنامه در شکل زیر آمده است.



```
1 from kafka import KafkaProducer
2 import time
3 import json
4 from csv import reader
5
6 producer = KafkaProducer(bootstrap_servers=['localhost:9092'])
7
8 if __name__ == '__main__':
9     with open('cassandraData.csv', 'r') as read_obj:
10         csv_dict_reader = reader(read_obj)
11         lineList = list(csv_dict_reader)
12         del lineList[0]
13         for i in lineList:
14             data = ','.join(map(str, i))
15             print(data)
16             producer.send('cassandratopic', data.encode('utf-8'))
17             time.sleep(5)
```

پس از ارسال داده‌ها به کافکا با یک برنامه consumer آن‌ها را دریافت کرده و به کاساندرا انتقال می‌دهیم. نام keyspace و جدول مربوطه را برای هر جدول در این برنامه وارد کرد و سپس آن را اجرا می‌کنیم. سپس با استفاده از دستورات زیر ابتدا یک keyspace و سپس جدول‌های مورد نظر را می‌سازیم.

```

producer.py x producer-cassandra.py x cassandraData.py x consumer.py x dataPrepare.py x
1 from kafka import KafkaConsumer
2 from cassandra.cluster import Cluster
3 import time
4
5 cluster = Cluster()
6 session = cluster.connect('finalkey')
7 consumer = KafkaConsumer('cassandratopic', bootstrap_servers='localhost:9092', auto_offset_reset='earliest', consumer_timeout_ms=1000)
8
9 l = []
10 a = 0
11 if __name__ == "__main__":
12     for message in consumer:
13         a = message.value.decode('utf-8')
14         l = a.split(',')
15         l[1] = float(l[1])
16         l[2] = float(l[2])
17         l[6] = int(l[6])
18         l[7] = int(l[7])
19         l[9] = int(l[9])
20         l[12] = int(l[12])
21         session.execute("INSERT INTO timetbl (datetime, lat, lon, Base, datetimestr, date, hour, weekdaynumber, weekdayname, weekmont
22         print("done!")
23         time.sleep(5)

```

CREATE KEYSPACE finalkey WITH replication = {'class' : 'SimpleStrategy',
'replication_factor' : '1'} AND durable_writes = 'true';
USE finalkey

```

C:\apache-cassandra-4.0.4\bin>cqlsh
WARNING: console codepage must be set to cp65001 to support utf-8 encoding on Windows platforms.
If you experience encoding problems, change your console codepage with 'chcp 65001' before starting cqlsh.

Connected to Test Cluster at 127.0.0.1:9042.
[cqlsh 5.0.1 | Cassandra 3.11.4 | CQL spec 3.4.4 | Native protocol v4]
Use HELP for help.
cqlsh> CREATE KEYSPACE finalkey WITH replication = {'class' : 'SimpleStrategy', 'replication_factor' : '1'} AND durable_writes = 'true';
cqlsh> USE finalkey;
cqlsh:finalkey>

```

- جدولی بر اساس کلید زمانی هفته‌ای طراحی کنید.

```
CREATE TABLE weektbl(datetime text, lat float, lon float, base text, datetimefloor text, date
text, hour int, weekdaynumber int, weekdayname text , weekmonthnumber int, weekmonthname
text, geo text, weeknumber int, time text, time12 text,id uuid, PRIMARY KEY((weeknumber,
geo), id));
```

```
mysql:finalkey> CREATE TABLE weektbl(datetime text, lat float, lon float, base text, datetimefloor text, date text, hour int, weekdaynumber int, weekdayname text , weekmonthnumber int, weekmonthname text, geo text, weeknumber int, time text, time12 text,id uuid, PRIMARY KEY((weeknumber, geo), id));
mysql:finalkey> SELECT * FROM weektbl;
```

weeknumber	geo	id	base	date	datetime	datetimefloor	hour	lat	lon	time	time12
2	weekdayname	weekdaynumber	weekmonthname	weekmonthnumber							
35	Monday	POINT (40.7559 -73.9864)	b7426334-0c15-461a-8655-b78e95082152	002512	2014-09-01	2014-09-01 00:03:00	2014-09-01 00:00:00	0	40.7559	-73.9864	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.7471 -73.6472)	e2657849-b10f-4579-ae7-6f399ce41c14	002512	2014-09-01	2014-09-01 00:15:00	2014-09-01 00:00:00	0	40.7471	-73.6472	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.7378 -74.0395)	a4b8ac62-79c3-41d9-b69c-b468c8db88aa	002512	2014-09-01	2014-09-01 00:48:00	2014-09-01 00:00:00	0	40.7378	-74.0395	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.2201 -74.0021)	9f8f96ba-8ebf-4456-bdee-28594002bc0d	002512	2014-09-01	2014-09-01 00:01:00	2014-09-01 00:00:00	0	40.2201	-74.0021	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.4023 -73.98390000000001)	281a1beb-70cf-4542-b228-f1734e2f4209	002512	2014-09-01	2014-09-01 00:39:00	2014-09-01 00:00:00	0	40.4023	-73.9839	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.8145 -73.9444)	41d93586-7b89-4d11-90c3-3d3fc877f9e0	002512	2014-09-01	2014-09-01 00:11:00	2014-09-01 00:00:00	0	40.8145	-73.9444	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.75 -74.0027)	0bcea478-bb6c-4038-b426-86ef6cd00c7d	002512	2014-09-01	2014-09-01 00:01:00	2014-09-01 00:00:00	0	40.75	-74.0027	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.7633 -73.9773)	9de95ddc-64b4-44fb-a519-d69b322be21c	002512	2014-09-01	2014-09-01 00:33:00	2014-09-01 00:00:00	0	40.7633	-73.9773	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.679 -74.0111)	7c61c868-d553-4559-aacf-d2d5b4131985	002512	2014-09-01	2014-09-01 00:38:00	2014-09-01 00:00:00	0	40.679	-74.0111	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.745 -73.9889)	198c8402-c2b6-4f4a-b358-02abda1964a8	002512	2014-09-01	2014-09-01 00:06:00	2014-09-01 00:00:00	0	40.745	-73.9889	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.7467 -73.6131)	fa2172c6-5128-4fbd-9568-019d80e3955e	002512	2014-09-01	2014-09-01 00:33:00	2014-09-01 00:00:00	0	40.7467	-73.6131	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.7214 -73.9884)	4e4dee16-d5de-4c27-abfc-858aa9c3d621	002512	2014-09-01	2014-09-01 00:48:00	2014-09-01 00:00:00	0	40.7214	-73.9884	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.6613 -74.26909999999999)	ac3c8c04-73d1-43c5-afa9-f73af8f8827c	002512	2014-09-01	2014-09-01 00:16:00	2014-09-01 00:00:00	0	40.6613	-74.2691	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.6735 -73.9918)	42beb774-b15f-4b2f-bfec-5b3084129bd2	002512	2014-09-01	2014-09-01 00:12:00	2014-09-01 00:00:00	0	40.6735	-73.9918	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.3745 -73.9999)	d70138f4-f4b1-4000-98b0-d71035220686	002512	2014-09-01	2014-09-01 00:32:00	2014-09-01 00:00:00	0	40.3745	-73.9999	2014-09-01 00:00:00 00:00
35	Monday	POINT (40.8105 -73.95999999999999)	2ed8f78d-0719-4614-a6b3-9b22290b72ee	002512	2014-09-01	2014-09-01 00:37:00	2014-09-01 00:00:00	0	40.8105	-73.96	2014-09-01 00:00:00 00:00

(16 rows)

- جدولی بر اساس مختصات نقطه‌ای شروع طراحی کنید.

```
CREATE TABLE latlontbl(datetime text, lat float, lon float, base text, datetimefloor text, date
text, hour int, weekdaynumber int, weekdayname text , weekmonthnumber int, weekmonthname
text, geo text, weeknumber int, time text, time12 text,id uuid, PRIMARY KEY((lat, lon), id));
```



```
cqlsh:finalkey> CREATE TABLE latlontbl(datetime text, lat float, lon float, base text, datetimetype floor text, date text, hour int, weekdaynumber int, weekdayname text , weekmonthnumber int, weekmonthname text, geo text, weeknumber int, time text, time12 text,id uuid, PRIMARY KEY((lat, lon), id));
cqlsh:finalkey> SELECT * FROM latlontbl;
```

lat	lon	id	base	date	datetime	datetimetype floor	geo	hour	time	time12	weekday
name weekdaynumber weekmonthname weekmonthnumber weeknumber											
40.8145	-73.9444	fe8a0782-95d0-4942-9d38-03a57062ea0a	B02512	2014-09-01	2014-09-01 00:11:00	2014-09-01 00:00:00	POINT (40.8145 -73.9444)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											
40.679	-74.0111	1b05b1b8-421e-438c-a9b3-266c49b8a73f	B02512	2014-09-01	2014-09-01 00:38:00	2014-09-01 00:00:00	POINT (40.679 -74.0111)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											
40.7633	-73.9773	24ca4887-d8b6-402e-9650-649eb8bd8cc4	B02512	2014-09-01	2014-09-01 00:33:00	2014-09-01 00:00:00	POINT (40.7633 -73.9773)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											
40.2201	-74.0021	3a0373b6-019a-4752-8d3f-a730e4f1f1b6	B02512	2014-09-01	2014-09-01 00:01:00	2014-09-01 00:00:00	POINT (40.2201 -74.0021)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											
40.7471	-73.6472	7b4a7a37-91e4-447d-a782-474a82e18cea	B02512	2014-09-01	2014-09-01 00:15:00	2014-09-01 00:00:00	POINT (40.7471 -73.6472)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											
40.75	-74.0027	fa2dc168-c794-def4-a5c0-9ad0a2530a7c	B02512	2014-09-01	2014-09-01 00:01:00	2014-09-01 00:00:00	POINT (40.75 -74.0027)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											
40.8105	-73.96	da731953-a67b-4418-915b-0221ec9af4c4e	B02512	2014-09-01	2014-09-01 00:37:00	2014-09-01 00:00:00	POINT (40.8105 -73.95999999999999)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											
40.4023	-73.9839	f474e54f-d096-4658-a494-b6b165ec5747	B02512	2014-09-01	2014-09-01 00:39:00	2014-09-01 00:00:00	POINT (40.4023 -73.9839000000000001)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											
40.7559	-73.9864	ca4cded7-aab4-4ad6-925f-4ab2d5f744c4	B02512	2014-09-01	2014-09-01 00:03:00	2014-09-01 00:00:00	POINT (40.7559 -73.9864)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											
40.7467	-73.6131	d709efc5-d529-4205-bb5a-7cec576e672a	B02512	2014-09-01	2014-09-01 00:33:00	2014-09-01 00:00:00	POINT (40.7467 -73.6131)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											
40.745	-73.9889	fe0c1b06-e8a2-4399-a087-5addcfeddd7d	B02512	2014-09-01	2014-09-01 00:06:00	2014-09-01 00:00:00	POINT (40.745 -73.9889)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											
40.3745	-73.9999	f1e723da-709c-4ab6-a6fc-618e8de80915	B02512	2014-09-01	2014-09-01 00:32:00	2014-09-01 00:00:00	POINT (40.3745 -73.9999)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											
40.6735	-73.9918	56e447e8-ce18-4a1e-9d04-839b51f23fb3	B02512	2014-09-01	2014-09-01 00:12:00	2014-09-01 00:00:00	POINT (40.6735 -73.9918)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											
40.6613	-74.2691	6a543c25-242d-4db5-a2c2-f8b82f62ef0b	B02512	2014-09-01	2014-09-01 00:16:00	2014-09-01 00:00:00	POINT (40.6613 -74.26909999999999)	0	2014-09-01 00:00:00	00:00:00	Mo
nday 0 September 9 35											

(14 rows)

- جدولی بر اساس زمانبندی ۱۲ ساعته سفرها طراحی کنید.

```
CREATE TABLE timetbl(datetime text, lat float, lon float, base text, datetimetype floor text, date text, hour int, weekdaynumber int, weekdayname text , weekmonthnumber int, weekmonthname text, geo text, weeknumber int, time text, time12 text,id uuid, PRIMARY KEY(time12, id));
cqlsh:finalkey> SELECT * FROM timetbl;
```

time12	id	base	date	datetime	datetimetype floor	geo	hour	lat	lon	time	weekday
name weekdaynumber weekmonthname weekmonthnumber weeknumber											
00:00:00	00b94293-9658-47c3-be87-4f5ad199041e	B02512	2014-09-01	2014-09-01 00:48:00	2014-09-01 00:00:00	POINT (40.7214 -73.9884)	0	40.7214	-73.9884	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	0db04b5a-09e0-4789-8c83-f1a18c436d8	B02512	2014-09-01	2014-09-01 00:39:00	2014-09-01 00:00:00	POINT (40.4023 -73.9839000000000001)	0	40.4023	-73.9839	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	1dd3af9b-95b7-44f7-8881-e2cecd4d5b38	B02512	2014-09-01	2014-09-01 01:08:00	2014-09-01 01:00:00	POINT (40.7398 -74.0061)	1	40.7398	-74.0061	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	3ea424d2-197b-436f-b927-8fb33ebdbdb9b	B02512	2014-09-01	2014-09-01 00:11:00	2014-09-01 00:00:00	POINT (40.8145 -73.9444)	0	40.8145	-73.9444	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	496bcad1-d4d6-4350-b6ac-7e9767530a2c	B02512	2014-09-01	2014-09-01 00:33:00	2014-09-01 00:00:00	POINT (40.7467 -73.6131)	0	40.7467	-73.6131	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	546b12d5-1df0-47f9-bbe6-8cd962eef6c6	B02512	2014-09-01	2014-09-01 00:48:00	2014-09-01 00:00:00	POINT (40.7378 -74.0395)	0	40.7378	-74.0395	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	60c94bb8-d456-4734-aa97-e0be51e6f61a	B02512	2014-09-01	2014-09-01 00:32:00	2014-09-01 00:00:00	POINT (40.3745 -73.9999)	0	40.3745	-73.9999	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	65332d5e-869f-4682-bc6d-2e598c598e84	B02512	2014-09-01	2014-09-01 00:12:00	2014-09-01 00:00:00	POINT (40.6735 -73.9918)	0	40.6735	-73.9918	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	698b777c-b2a2-4c81-bc84-f3400a86663f	B02512	2014-09-01	2014-09-01 00:37:00	2014-09-01 00:00:00	POINT (40.8105 -73.95999999999999)	0	40.8105	-73.96	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	8a9b30b1-3a38-4d7d-ba76-cb3aeff6e6ab	B02512	2014-09-01	2014-09-01 00:49:00	2014-09-01 00:00:00	POINT (40.8646 -73.9081)	0	40.8646	-73.9081	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	8fc5d2f3-18ae-437c-bbcc-f32a0733fee8	B02512	2014-09-01	2014-09-01 00:01:00	2014-09-01 00:00:00	POINT (40.75 -74.0027)	0	40.75	-74.0027	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	93bde59d-47be-4f9e-a003-b3576d7a4e8	B02512	2014-09-01	2014-09-01 00:01:00	2014-09-01 00:00:00	POINT (40.2201 -74.0021)	0	40.2201	-74.0021	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	99505240-c364-4354-bfbf-633158733840	B02512	2014-09-01	2014-09-01 01:17:00	2014-09-01 01:00:00	POINT (40.6793 -74.0116)	1	40.6793	-74.0116	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	c167e04d-1b41-4812-8a12-a1cae8f7296f	B02512	2014-09-01	2014-09-01 00:06:00	2014-09-01 00:00:00	POINT (40.745 -73.9889)	0	40.745	-73.9889	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	ccdcf4a0-07ea-4808-9d58-fa90eccb4b48	B02512	2014-09-01	2014-09-01 00:38:00	2014-09-01 00:00:00	POINT (40.679 -74.0111)	0	40.679	-74.0111	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	d6b5b2ed-9f9d-41a9-a83e-764b25e3aa32	B02512	2014-09-01	2014-09-01 00:33:00	2014-09-01 00:00:00	POINT (40.7633 -73.9773)	0	40.7633	-73.9773	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	d9bdc5b2-df50-4437-aab-5fc395b646f1	B02512	2014-09-01	2014-09-01 00:03:00	2014-09-01 00:00:00	POINT (40.7559 -73.9864)	0	40.7559	-73.9864	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	f32ce583-f4c2-4aff-b61c-76711cf9a8fa	B02512	2014-09-01	2014-09-01 00:16:00	2014-09-01 00:00:00	POINT (40.6613 -74.26909999999999)	0	40.6613	-74.2691	2014-09-01 00:00:00	Mo
nday 0 September 9 35											
00:00:00	fc7d52dd-9fc0-4906-951d-901d36c78325	B02512	2014-09-01	2014-09-01 00:15:00	2014-09-01 00:00:00	POINT (40.7471 -73.6472)	0	40.7471	-73.6472	2014-09-01 00:00:00	Mo
nday 0 September 9 35											

(19 rows)

با استفاده از دستورات فوق جداول ساخته می‌شوند.

دستورات زیر کوئری‌های خواسته شده را پیاده‌سازی می‌کنند.

- بازیابی سفرهای ۱ روز اخیر، ۶ ساعت اخیر

SELECT id, datetime, geo, base, weeknumber, weekdayname FROM timetbl WHERE time12 = '00:00:00';

```
cqlsh:finalkey> SELECT id, datetime, geo, base, weeknumber, weekdayname FROM timetbl WHERE time12 = '00:00:00';
```

id	datetime	geo	base	weeknumber	weekdayname
00b94293-9658-47c3-be87-4f5ad199041e	2014-09-01 00:48:00	POINT (40.7214 -73.9884)	B02512	35	Monday
0db04b5a-09e0-4789-8c83-8f1a18c436d8	2014-09-01 00:39:00	POINT (40.4023 -73.98390000000001)	B02512	35	Monday
0ecc29d7-0976-4be7-adc7-2e406c967b70	2014-09-01 01:39:00	POINT (40.6743 -73.93340000000001)	B02512	35	Monday
11520152-d606-4f4f-906f-823436433ae2	2014-09-01 05:55:00	POINT (40.7769 -73.9862)	B02512	35	Monday
1ad5e518-62a9-4604-83f4-b6314b934bd3	2014-09-01 04:06:00	POINT (40.7983 -73.9628)	B02512	35	Monday
1dd3af9b-95b7-44f7-8881-e2cec4d5b538	2014-09-01 01:08:00	POINT (40.7398 -74.0061)	B02512	35	Monday
23b33a34-0484-453e-8e4b-cbb8b6f8c5aa	2014-09-01 06:14:00	POINT (40.7674 -73.9841)	B02512	35	Monday
29a67bb9-e8df-420a-b0fc-427bfb63b7ab	2014-09-01 06:35:00	POINT (40.7685 -73.8626)	B02512	35	Monday
3ea424d2-197b-436f-b927-8fb33ebddb9b	2014-09-01 00:11:00	POINT (40.8145 -73.9444)	B02512	35	Monday
496bcadf-d4d6-4350-b6ac-7e9767530a2c	2014-09-01 00:33:00	POINT (40.7467 -73.6131)	B02512	35	Monday
52a996a6-bab6-45ec-a61d-fa943fb7f6e5	2014-09-01 03:18:00	POINT (40.7529 -74.004)	B02512	35	Monday
534cb7db-b2fb-45ed-a3e1-656a79bc9b5d	2014-09-01 02:41:00	POINT (40.7494 -73.99460000000001)	B02512	35	Monday
546b12d5-1df0-47f9-8be6-8cd962eef6c6	2014-09-01 00:48:00	POINT (40.7378 -74.0395)	B02512	35	Monday
60c94bb8-d456-4734-aa97-e6be51e6f61a	2014-09-01 00:32:00	POINT (40.3745 -73.9999)	B02512	35	Monday
6462e3c9-1ae0-4f1f-a406-c041e39aa2a1	2014-09-01 01:19:00	POINT (40.7328 -73.9875)	B02512	35	Monday
65332d5e-869f-4682-bc6d-2e598c598e84	2014-09-01 00:12:00	POINT (40.6735 -73.9918)	B02512	35	Monday
698b777c-b2a2-4c81-bc84-f3400a86663f	2014-09-01 00:37:00	POINT (40.8105 -73.95999999999999)	B02512	35	Monday
6ebcd035-9997-4d46-b302-9d0c4abc6615	2014-09-01 05:02:00	POINT (40.7193 -73.99979999999999)	B02512	35	Monday
709cc062-5688-4bfb-8a85-4c07f478f290	2014-09-01 04:34:00	POINT (40.707 -74.00449999999999)	B02512	35	Monday
72afd82e-67cf-4045-808d-4c372651360e	2014-09-01 05:45:00	POINT (40.7501 -74.0025)	B02512	35	Monday
73f5b778-e16d-4d59-a5e7-ac63c6b0756f	2014-09-01 05:12:00	POINT (40.8293 -73.9432)	B02512	35	Monday
7aa05f2f-3d39-4cb8-a7a8-b2bd355abd56	2014-09-01 04:01:00	POINT (41.0319 -74.18129999999999)	B02512	35	Monday
876ec9d0-0779-41e1-ae43-5b76101a9a48	2014-09-01 04:20:00	POINT (40.6932 -73.943)	B02512	35	Monday
8a9b30b1-3a38-47d7-ba76-cb3aeff6e6ab	2014-09-01 00:49:00	POINT (40.8646 -73.9081)	B02512	35	Monday
8fc5d2f3-18ae-437c-bbcc-f32a0733fee8	2014-09-01 00:01:00	POINT (40.75 -74.0027)	B02512	35	Monday
93dbe59d-47be-4f9e-a003-b3576d7a4eb8	2014-09-01 00:01:00	POINT (40.2201 -74.0021)	B02512	35	Monday
99505240-c364-4354-bfbf-633158733840	2014-09-01 01:17:00	POINT (40.6793 -74.0116)	B02512	35	Monday
9bd3435e-0fc2-4eda-bd97-72ce39228f14	2014-09-01 02:03:00	POINT (40.7638 -73.9811)	B02512	35	Monday
a4573a19-7349-4f6a-9d6d-72ea171702a6	2014-09-01 06:43:00	POINT (40.8298 -73.97190000000001)	B02512	35	Monday
af0c6b3e-3619-4e10-b92d-1913622542d0	2014-09-01 01:41:00	POINT (40.7638 -73.9962)	B02512	35	Monday
c167e04d-1b41-4812-8a12-a1cae8ff2967	2014-09-01 00:06:00	POINT (40.745 -73.9889)	B02512	35	Monday
c10270e4-b2e4-40e0-0b05-3140f00b513e	2014-09-01 02:20:00	POINT (40.6614 -73.9400)	B02512	35	Monday

- بازیابی سفرها برای یک مختصات خاص

SELECT id, datetime, geo, base, weeknumber, weekdayname FROM latlontbl WHERE lat = 40.7559 AND lon = -73.9864;

```
cqlsh:finalkey> SELECT id, datetime, geo, base, weeknumber, weekdayname FROM latlontbl WHERE lat = 40.7559 AND lon = -73.9864;
```

id	datetime	geo	base	weeknumber	weekdayname
ca4cded7-aab4-4ad6-925f-4ab2d5f744c4	2014-09-01 00:03:00	POINT (40.7559 -73.9864)	B02512	35	Monday

(1 rows)

- بازیابی زمان‌های در خواست برای یک مختصات در تایم یک هفته

```
SELECT id, datetime, geo, base, weeknumber, weekdayname FROM weektbl WHERE
weeknumber = 35 AND geo = 'POINT (40.745 -73.9889)';
```

```
cqlsh:finalkey> SELECT id, datetime, geo, base, weeknumber, weekdayname FROM weektbl WHERE weeknumber = 35 AND geo = 'POINT (40.7471 -73.6472)';
```

id	datetime	geo	base	weeknumber	weekdayname
e2657849-b10f-4579-ae7-6f399ce41c14	2014-09-01 00:15:00	POINT (40.7471 -73.6472)	B02512	35	Monday

```
(1 rows)
cqlsh:finalkey> SELECT id, datetime, geo, base, weeknumber, weekdayname FROM weektbl WHERE weeknumber = 35 AND geo = 'POINT (40.745 -73.9889)';
```

id	datetime	geo	base	weeknumber	weekdayname
198c8402-c2b6-4f4a-b358-02abda1964a8	2014-09-01 00:06:00	POINT (40.745 -73.9889)	B02512	35	Monday

```
(1 rows)
```

لینک گیت‌هاب: <https://github.com/AtiyehNikbakht/FinalPro.git>