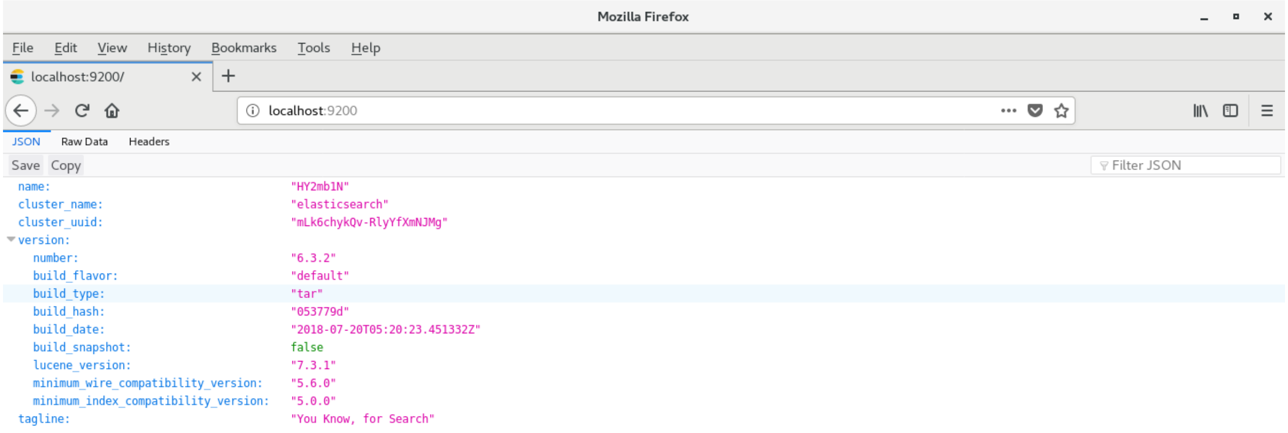
Analysing Cars Data using Elastic Search, Log Stash and Kibana

1. Start the Elastic Search

$ /usr/local/elasticsearch-6.3.2/bin/elasticsearch

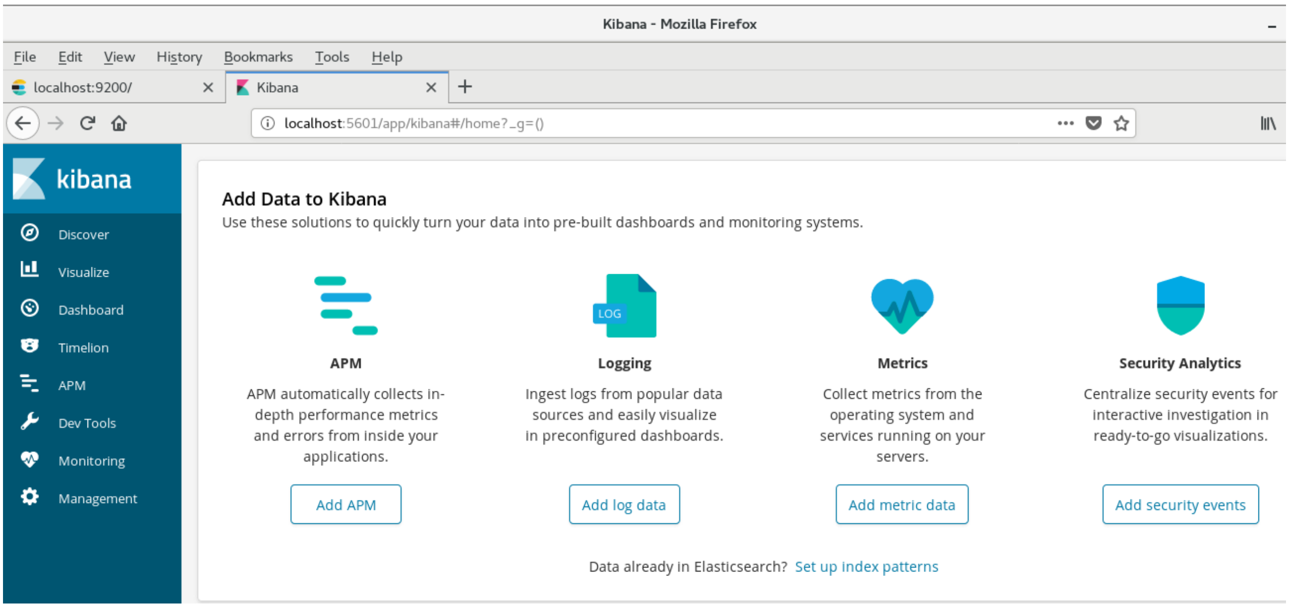


```
name:                                   "HY2mb1N"
cluster_name:                           "elasticsearch"
cluster_uuid:                           "mLk6chykQv-RlyYfXmNJMg"
version:
  number:                               "6.3.2"
  build_flavor:                         "default"
  build_type:                           "tar"
  build_hash:                           "053779d"
  build_date:                           "2018-07-20T05:20:23.451332Z"
  build_snapshot:                       false
  lucene_version:                       "7.3.1"
  minimum_wire_compatibility_version:   "5.6.0"
  minimum_index_compatibility_version:  "5.0.0"
tagline:                                "You Know, for Search"
```

Elastic search is up and running at http://localhost:9200

2. Start the Kibana

$ /usr/local/kibana-6.3.2-linux-x86_64/bin/kibana



Add Data to Kibana

Use these solutions to quickly turn your data into pre-built dashboards and monitoring systems.

APM

APM automatically collects in-depth performance metrics and errors from inside your applications.

Logging

Ingest logs from popular data sources and easily visualize in preconfigured dashboards.

Metrics

Collect metrics from the operating system and services running on your servers.

Security Analytics

Centralize security events for interactive investigation in ready-to-go visualizations.

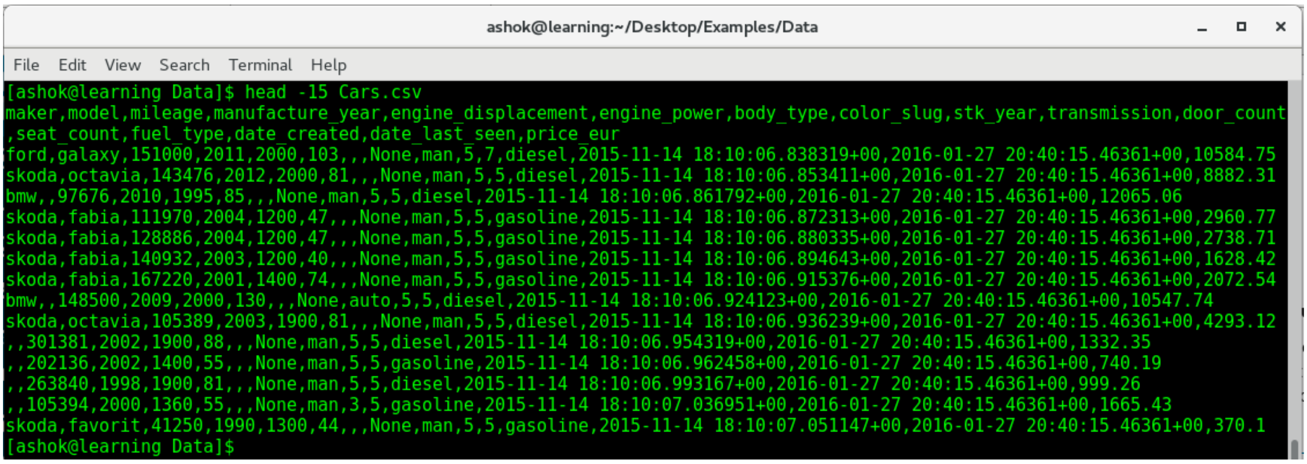
Data already in Elasticsearch? Set up index patterns

Kibana is up and running at http://localhost:5601

3. Input Data:

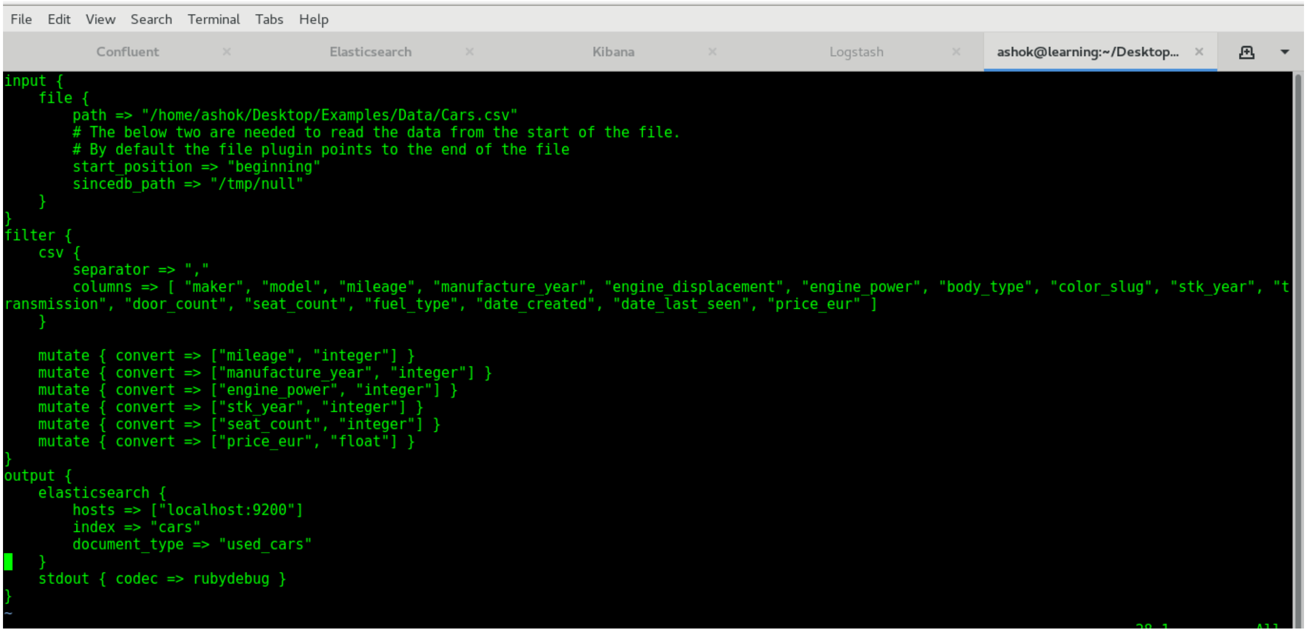
Prepare some data in CSV. For this example data is stored in the following file.

/home/ashok/Desktop/Examples/Data/Cars.csv



```
[ashok@learning Data]$ head -15 Cars.csv
maker,model,mileage,manufacture_year,engine_displacement,engine_power,body_type,color_slug,stk_year,transmission,door_count
,seat_count,fuel_type,date_created,date_last_seen,price_eur
ford,galaxy,151000,2011,2000,103,,,None,man,5,7,diesel,2015-11-14 18:10:06.838319+00,2016-01-27 20:40:15.46361+00,10584.75
skoda,octavia,143476,2012,2000,81,,,None,man,5,5,diesel,2015-11-14 18:10:06.853411+00,2016-01-27 20:40:15.46361+00,8882.31
bmw,,97676,2010,1995,85,,,None,man,5,5,diesel,2015-11-14 18:10:06.861792+00,2016-01-27 20:40:15.46361+00,12065.06
skoda,fabia,111970,2004,1200,47,,,None,man,5,5,gasoline,2015-11-14 18:10:06.872313+00,2016-01-27 20:40:15.46361+00,2960.77
skoda,fabia,128886,2004,1200,47,,,None,man,5,5,gasoline,2015-11-14 18:10:06.880335+00,2016-01-27 20:40:15.46361+00,2738.71
skoda,fabia,140932,2003,1200,40,,,None,man,5,5,gasoline,2015-11-14 18:10:06.894643+00,2016-01-27 20:40:15.46361+00,1628.42
skoda,fabia,167220,2001,1400,74,,,None,man,5,5,gasoline,2015-11-14 18:10:06.915376+00,2016-01-27 20:40:15.46361+00,2072.54
bmw,,148500,2009,2000,130,,,None,auto,5,5,diesel,2015-11-14 18:10:06.924123+00,2016-01-27 20:40:15.46361+00,10547.74
skoda,octavia,105389,2003,1900,81,,,None,man,5,5,diesel,2015-11-14 18:10:06.936239+00,2016-01-27 20:40:15.46361+00,4293.12
,,301381,2002,1900,88,,,None,man,5,5,diesel,2015-11-14 18:10:06.954319+00,2016-01-27 20:40:15.46361+00,1332.35
,,202136,2002,1400,55,,,None,man,5,5,gasoline,2015-11-14 18:10:06.962458+00,2016-01-27 20:40:15.46361+00,740.19
,,263840,1998,1900,81,,,None,man,5,5,diesel,2015-11-14 18:10:06.993167+00,2016-01-27 20:40:15.46361+00,999.26
,,105394,2000,1360,55,,,None,man,3,5,gasoline,2015-11-14 18:10:07.036951+00,2016-01-27 20:40:15.46361+00,1665.43
skoda,favorit,41250,1990,1300,44,,,None,man,5,5,gasoline,2015-11-14 18:10:07.051147+00,2016-01-27 20:40:15.46361+00,370.1
[ashok@learning Data]$
```

4. Write the Log Stash Configuration file.

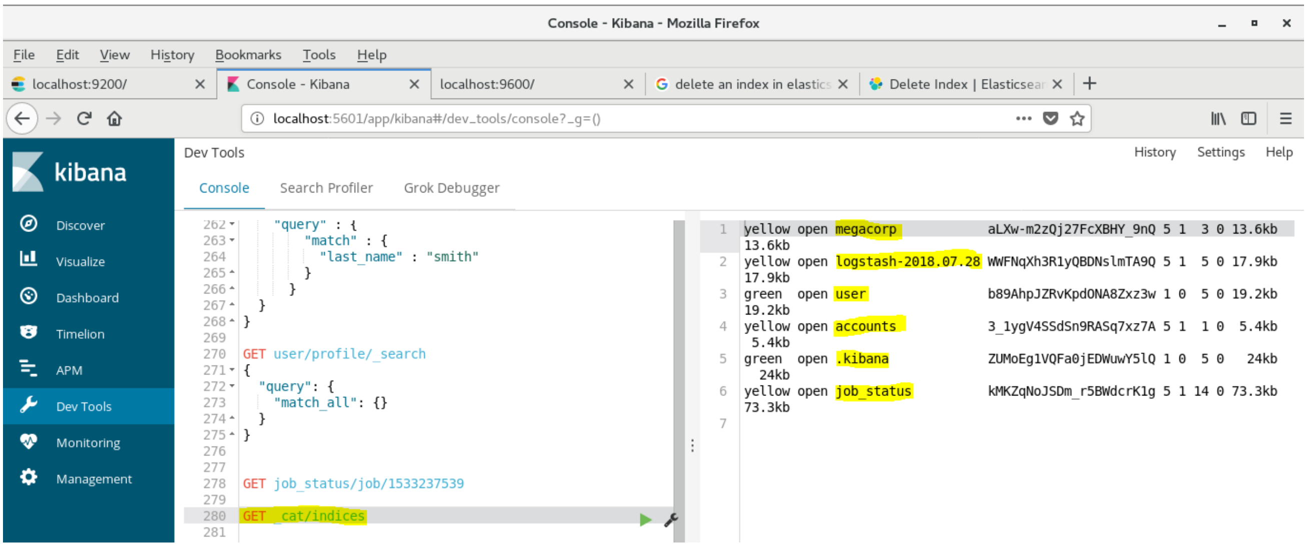


```
input {
    file {
        path => "/home/ashok/Desktop/Examples/Data/Cars.csv"
        # The below two are needed to read the data from the start of the file.
        # By default the file plugin points to the end of the file
        start_position => "beginning"
        sincedb_path => "/tmp/null"
    }
}
filter {
    csv {
        separator => ","
        columns => [ "maker", "model", "mileage", "manufacture_year", "engine_displacement", "engine_power", "body_type", "color_slug", "stk_year", "t
ransmission", "door_count", "seat_count", "fuel_type", "date_created", "date_last_seen", "price_eur" ]
    }

    mutate { convert => ["mileage", "integer"] }
    mutate { convert => ["manufacture_year", "integer"] }
    mutate { convert => ["engine_power", "integer"] }
    mutate { convert => ["stk_year", "integer"] }
    mutate { convert => ["seat_count", "integer"] }
    mutate { convert => ["price_eur", "float"] }
}
output {
    elasticsearch {
        hosts => ["localhost:9200"]
        index => "cars"
        document_type => "used_cars"
    }
    stdout { codec => rubydebug }
}
~
```

<u>Logstash_cars.config</u>

```
input {

   file {

      path => "/home/ashok/Desktop/Examples/Data/Cars.csv"

      # The below two are needed to read the data from the start of the file.

      # By default the file plugin points to the end of the file

      start_position => "beginning"

      sincedb_path => "/tmp/null"

   }

}

filter {
   csv {
      separator => ","
      columns => [ "maker", "model", "mileage", "manufacture_year", "engine_displacement", "engine_power", "body_type", "color_slug", "stk_year", "transmission", "door_count", "seat_count", "fuel_type", "date_created", "date_last_seen", "price_eur" ]
   }
   mutate { convert => ["mileage", "integer"] }
   mutate { convert => ["manufacture_year", "integer"] }
   mutate { convert => ["engine_power", "integer"] }
   mutate { convert => ["stk_year", "integer"] }
   mutate { convert => ["seat_count", "integer"] }
   mutate { convert => ["price_eur", "float"] }
}
output {
   elasticsearch {
      hosts => ["localhost:9200"]
      index => "cars"
      document_type => "used_cars"
   }
   stdout { codec => rubydebug }
}
```

5. List of Indices in Elastic Search before running Log Stash



```
    "query" : {
      "match" : {
        "last_name" : "smith"
      }
    }
  }
}

GET user/profile/_search
{
  "query": {
    "match_all": {}
  }
}


GET job_status/job/1533237539

GET _cat/indices
```

```
yellow open megacorp            aLXw-m2zQj27FcXBHY_9nQ 5 1  3 0 13.6kb
13.6kb
yellow open logstash-2018.07.28 WWFNqXh3R1yQBDNslmTA9Q 5 1  5 0 17.9kb
17.9kb
green  open user                b89AhpJZRvKpdONA8Zxz3w 1 0  5 0 19.2kb
19.2kb
yellow open accounts            3_1ygV4SSdSn9RASq7xz7A 5 1  1 0  5.4kb
 5.4kb
green  open .kibana             ZUMoEg1VQFa0jEDWuwY5lQ 1 0  5 0   24kb
  24kb
yellow open job_status          kMKZqNoJSDm_r5BWdcrK1g 5 1 14 0 73.3kb
73.3kb
```

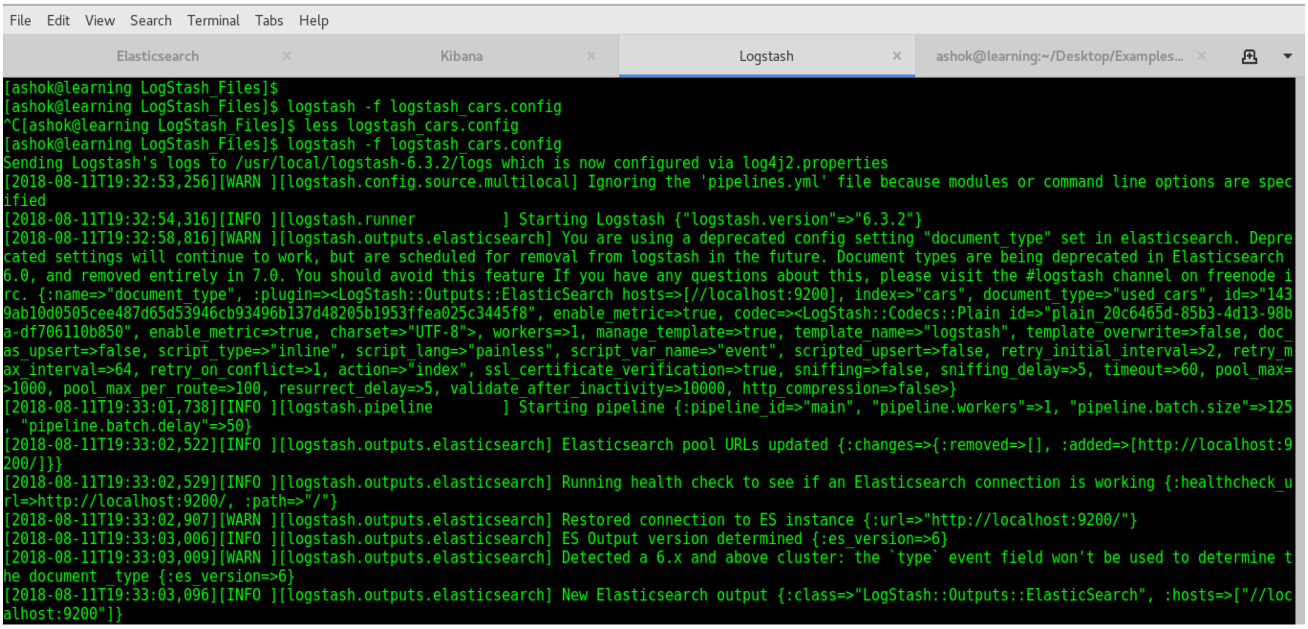
6. Run the Log Stash as follows:

Checking the configuration file:

$ logstash -f logstash_cars.config –config.text_and_exit

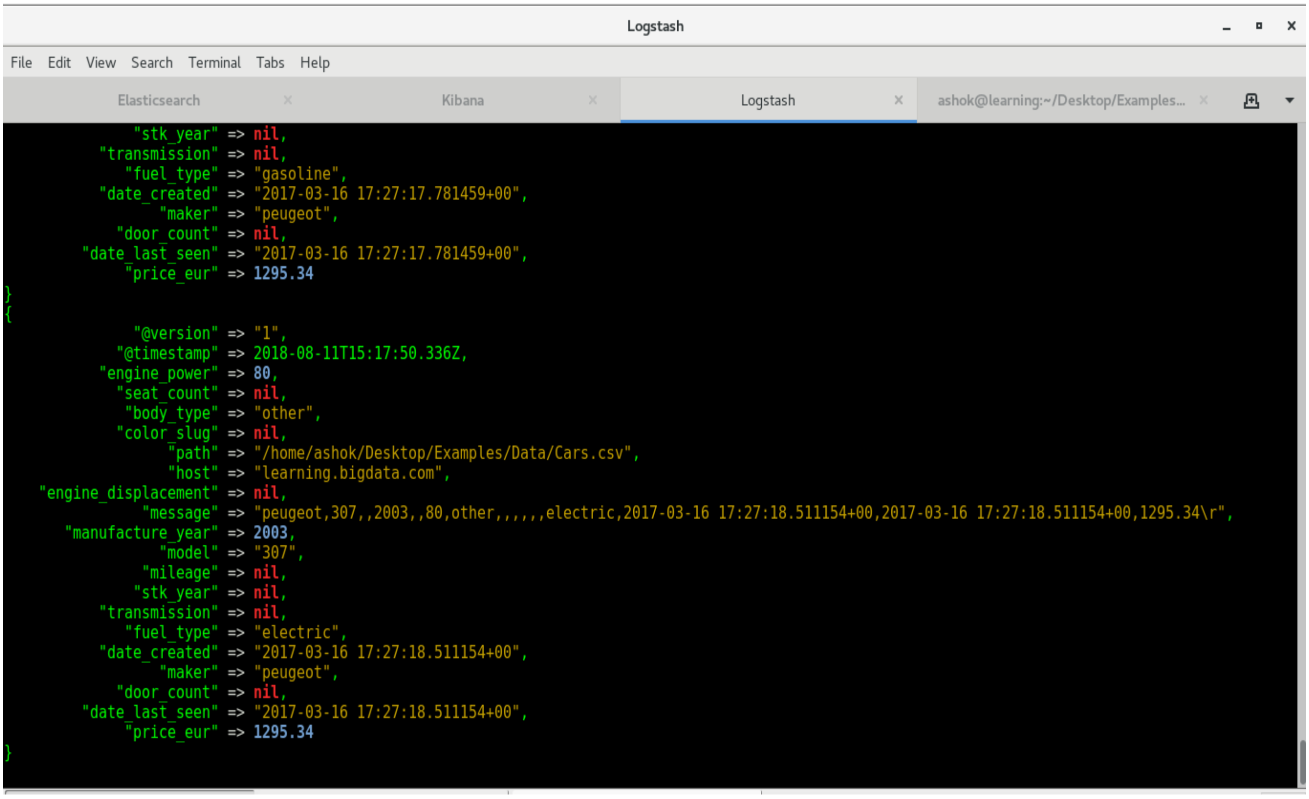
Running:

$ logstash -f logstash_cars.config


File Edit View Search Terminal Tabs Help
Elasticsearch
Kibana
Logstash
ashok@learning:~/Desktop/Examples...

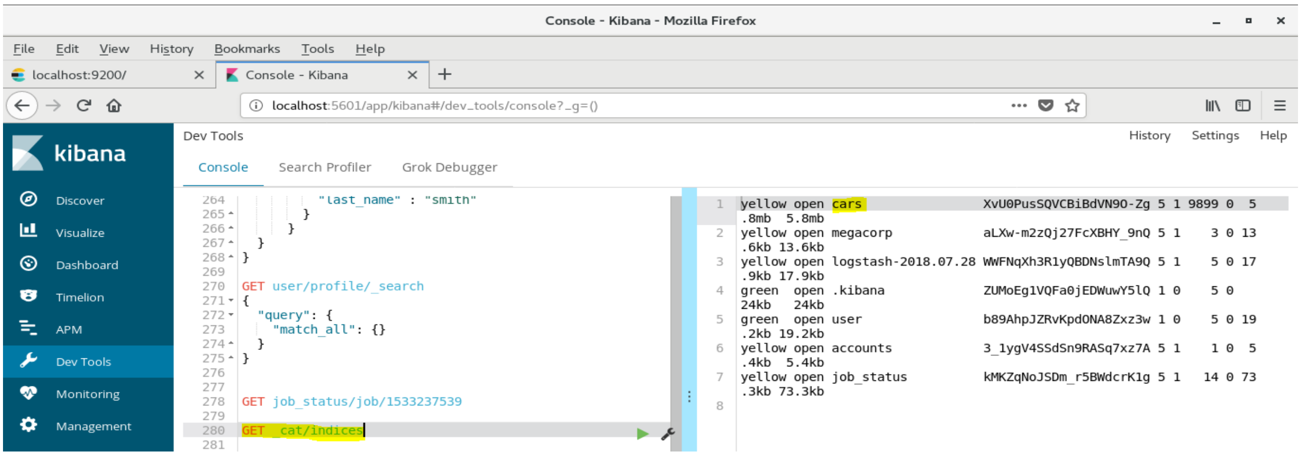
```
[ashok@learning LogStash_Files]$
[ashok@learning LogStash_Files]$ logstash -f logstash_cars.config
^C[ashok@learning LogStash_Files]$ less logstash_cars.config
[ashok@learning LogStash_Files]$ logstash -f logstash_cars.config
Sending Logstash's logs to /usr/local/logstash-6.3.2/logs which is now configured via log4j2.properties
[2018-08-11T19:32:53,256][WARN ][logstash.config.source.multilocal] Ignoring the 'pipelines.yml' file because modules or command line options are spec
ified
[2018-08-11T19:32:54,316][INFO ][logstash.runner          ] Starting Logstash {"logstash.version"=>"6.3.2"}
[2018-08-11T19:32:58,816][WARN ][logstash.outputs.elasticsearch] You are using a deprecated config setting "document_type" set in elasticsearch. Depre
cated settings will continue to work, but are scheduled for removal from logstash in the future. Document types are being deprecated in Elasticsearch
6.0, and removed entirely in 7.0. You should avoid this feature If you have any questions about this, please visit the #logstash channel on freenode i
rc. {:name=>"document_type", :plugin=><LogStash::Outputs::ElasticSearch hosts=>[//localhost:9200], index=>"cars", document_type=>"used_cars", id=>"143
9ab10d0505cee487d65d53946cb93496b137d48205b1953ffea025c3445f8", enable_metric=>true, codec=><LogStash::Codecs::Plain id=>"plain_20c6465d-85b3-4d13-98b
a-df706110b850", enable_metric=>true, charset=>"UTF-8">, workers=>1, manage_template=>true, template_name=>"logstash", template_overwrite=>false, doc_
as_upsert=>false, script_type=>"inline", script_lang=>"painless", script_var_name=>"event", scripted_upsert=>false, retry_initial_interval=>2, retry_m
ax_interval=>64, retry_on_conflict=>1, action=>"index", ssl_certificate_verification=>true, sniffing=>false, sniffing_delay=>5, timeout=>60, pool_max=
>1000, pool_max_per_route=>100, resurrect_delay=>5, validate_after_inactivity=>10000, http_compression=>false>}
[2018-08-11T19:33:01,738][INFO ][logstash.pipeline        ] Starting pipeline {:pipeline_id=>"main", "pipeline.workers"=>1, "pipeline.batch.size"=>125
, "pipeline.batch.delay"=>50}
[2018-08-11T19:33:02,522][INFO ][logstash.outputs.elasticsearch] Elasticsearch pool URLs updated {:changes=>{:removed=>[], :added=>[http://localhost:9
200/]}}
[2018-08-11T19:33:02,529][INFO ][logstash.outputs.elasticsearch] Running health check to see if an Elasticsearch connection is working {:healthcheck_u
rl=>http://localhost:9200/, :path=>"/"}
[2018-08-11T19:33:02,907][WARN ][logstash.outputs.elasticsearch] Restored connection to ES instance {:url=>"http://localhost:9200/"}
[2018-08-11T19:33:03,006][INFO ][logstash.outputs.elasticsearch] ES Output version determined {:es_version=>6}
[2018-08-11T19:33:03,009][WARN ][logstash.outputs.elasticsearch] Detected a 6.x and above cluster: the `type` event field won't be used to determine t
he document _type {:es_version=>6}
[2018-08-11T19:33:03,096][INFO ][logstash.outputs.elasticsearch] New Elasticsearch output {:class=>"LogStash::Outputs::ElasticSearch", :hosts=>["//loc
alhost:9200"]}
```

Sample record loading into Elastic Search:


Logstash
File Edit View Search Terminal Tabs Help
Elasticsearch
Kibana
Logstash
ashok@learning:~/Desktop/Examples...

```
               "stk_year" => nil,
           "transmission" => nil,
              "fuel_type" => "gasoline",
           "date_created" => "2017-03-16 17:27:17.781459+00",
                  "maker" => "peugeot",
             "door_count" => nil,
         "date_last_seen" => "2017-03-16 17:27:17.781459+00",
              "price_eur" => 1295.34
}
{

               "@version" => "1",
             "@timestamp" => 2018-08-11T15:17:50.336Z,
           "engine_power" => 80,
             "seat_count" => nil,
              "body_type" => "other",
             "color_slug" => nil,
                   "path" => "/home/ashok/Desktop/Examples/Data/Cars.csv",
                   "host" => "learning.bigdata.com",
    "engine_displacement" => nil,
                "message" => "peugeot,307,,2003,,80,other,,,,,,electric,2017-03-16 17:27:18.511154+00,2017-03-16 17:27:18.511154+00,1295.34\r",
       "manufacture_year" => 2003,
                  "model" => "307",
                "mileage" => nil,
               "stk_year" => nil,
           "transmission" => nil,
              "fuel_type" => "electric",
           "date_created" => "2017-03-16 17:27:18.511154+00",
                  "maker" => "peugeot",
             "door_count" => nil,
         "date_last_seen" => "2017-03-16 17:27:18.511154+00",
              "price_eur" => 1295.34
}
```

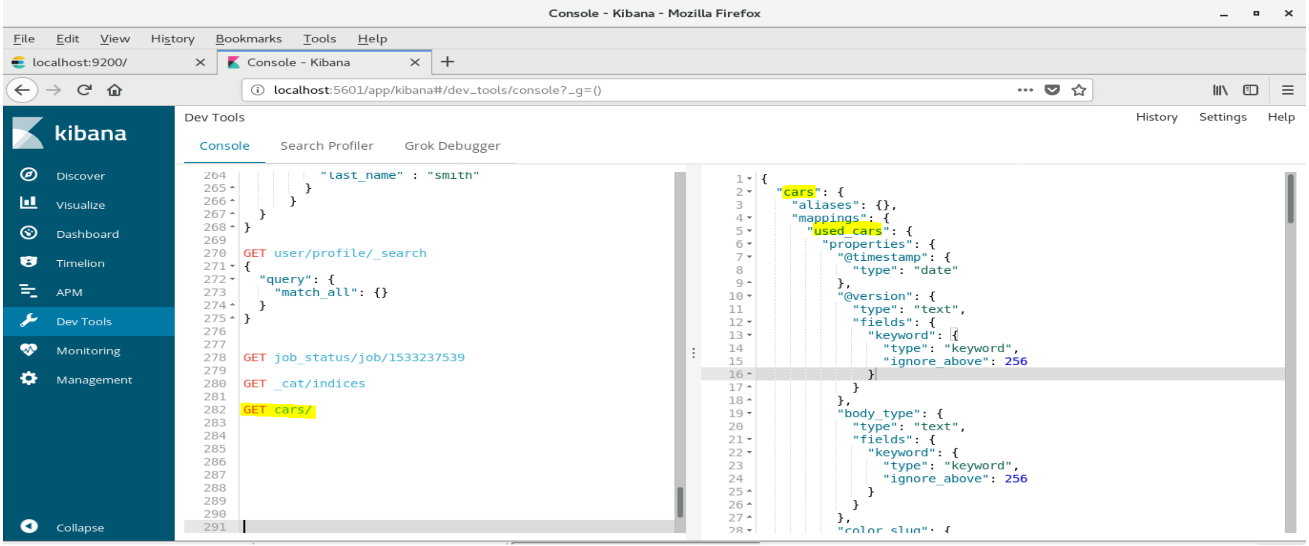
7. After launching Log Stash, a new Index is created in Elastic search


Console - Kibana - Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:9200/
Console - Kibana
localhost:5601/app/kibana#/dev_tools/console?_g=()
kibana
Discover
Visualize
Dashboard
Timelion
APM
Dev Tools
Monitoring
Management
Dev Tools
Console Search Profiler Grok Debugger
History Settings Help

```
        "last_name" : "smith"
      }
    }
  }
}

GET user/profile/_search
{
  "query": {
    "match_all": {}
  }
}


GET job_status/job/1533237539

GET _cat/indices
```

```
yellow open cars                XvU0PusSQVCBiBdVN9O-Zg 5 1 9899 0  5
.8mb  5.8mb
yellow open megacorp            aLXw-m2zQj27FcXBHY_9nQ 5 1    3 0 13
.6kb 13.6kb
yellow open logstash-2018.07.28 WWFNqXh3R1yQBDNslmTA9Q 5 1    5 0 17
.9kb 17.9kb
green  open .kibana             ZUMoEg1VQFa0jEDWuwY5lQ 1 0    5 0
24kb   24kb
green  open user                b89AhpJZRvKpdONA8Zxz3w 1 0    5 0 19
.2kb 19.2kb
yellow open accounts            3_1ygV4SSdSn9RASq7xz7A 5 1    1 0  5
.4kb  5.4kb
yellow open job_status          kMKZqNoJSDm_r5BWdcrK1g 5 1   14 0 73
.3kb 73.3kb
```

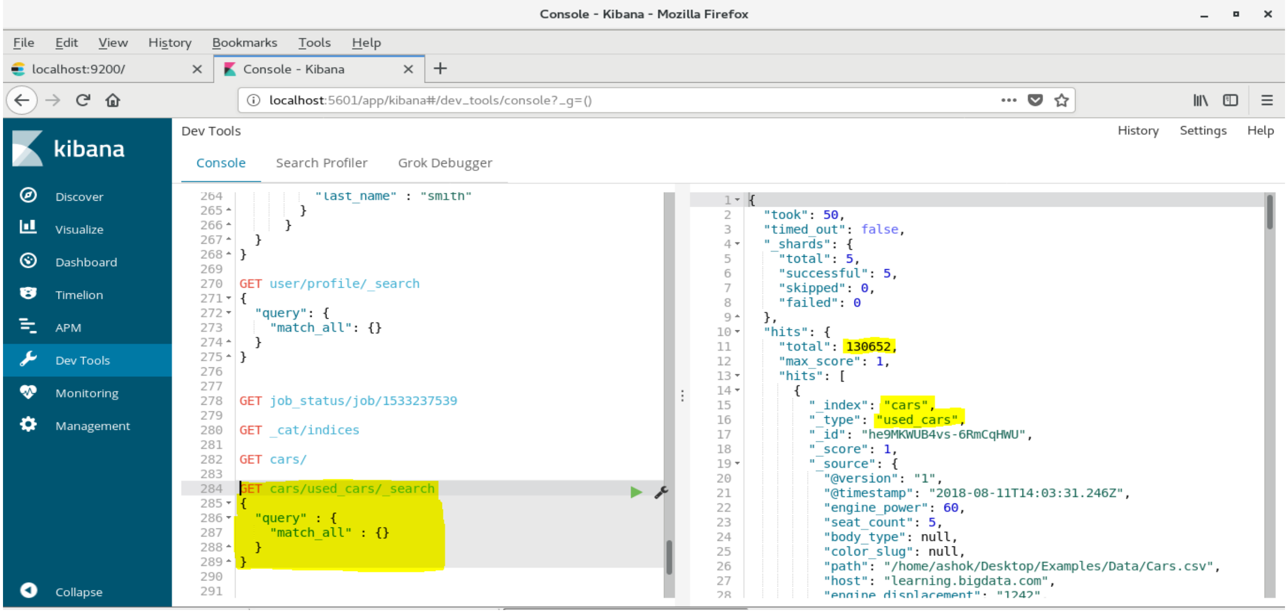
Description of the Index (CARS)


Console - Kibana - Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:9200/
Console - Kibana
localhost:5601/app/kibana#/dev_tools/console?_g=()
kibana
Discover
Visualize
Dashboard
Timelion
APM
Dev Tools
Monitoring
Management
Collapse
Dev Tools
Console Search Profiler Grok Debugger
History Settings Help

```
        "last_name" : "smith"
      }
    }
  }
}

GET user/profile/_search
{
  "query": {
    "match_all": {}
  }
}


GET job_status/job/1533237539

GET _cat/indices

GET cars/
```

```
{
  "cars": {
    "aliases": {},
    "mappings": {
      "used_cars": {
        "properties": {
          "@timestamp": {
            "type": "date"
          },
          "@version": {
            "type": "text",
            "fields": {
              "keyword": {
                "type": "keyword",
                "ignore_above": 256
              }
            }
          },
          "body_type": {
            "type": "text",
            "fields": {
              "keyword": {
                "type": "keyword",
                "ignore_above": 256
              }
            }
          },
          "color_slug": {
```

Able to the see some records in Elastic Search

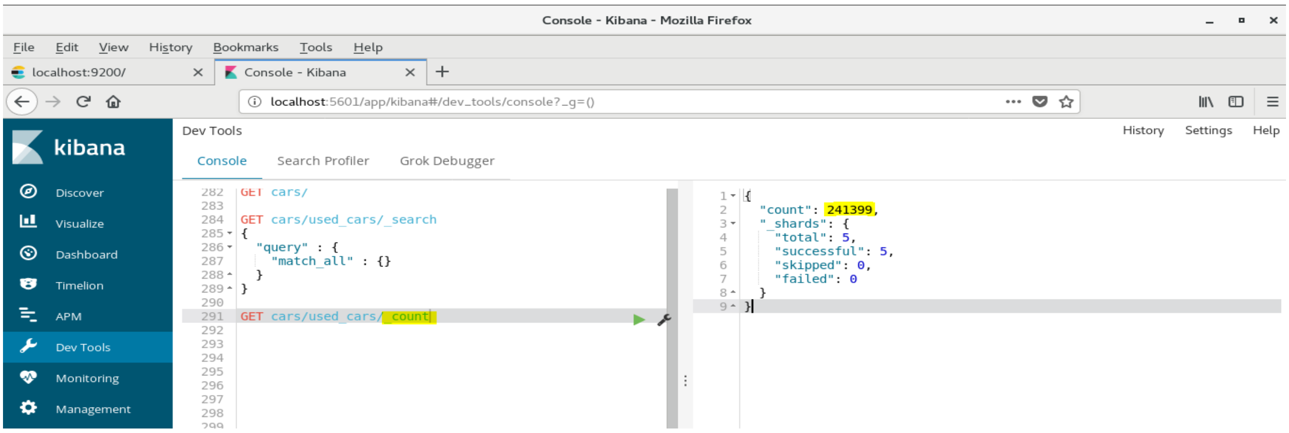

Console - Kibana - Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:9200/
Console - Kibana
localhost:5601/app/kibana#/dev_tools/console?_g=()
kibana
Discover
Visualize
Dashboard
Timelion
APM
Dev Tools
Monitoring
Management
Collapse
Dev Tools
Console Search Profiler Grok Debugger
History Settings Help

```
        "last_name" : "smith"
      }
    }
  }
}

GET user/profile/_search
{
  "query": {
    "match_all": {}
  }
}


GET job_status/job/1533237539

GET _cat/indices

GET cars/

GET cars/used_cars/_search
{
  "query" : {
    "match_all" : {}
  }
}
```

```
{
  "took": 50,
  "timed_out": false,
  "_shards": {
    "total": 5,
    "successful": 5,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": 130652,
    "max_score": 1,
    "hits": [
      {
        "_index": "cars",
        "_type": "used_cars",
        "_id": "he9MKWUB4vs-6RmCqHWU",
        "_score": 1,
        "_source": {
          "@version": "1",
          "@timestamp": "2018-08-11T14:03:31.246Z",
          "engine_power": 60,
          "seat_count": 5,
          "body_type": null,
          "color_slug": null,
          "path": "/home/ashok/Desktop/Examples/Data/Cars.csv",
          "host": "learning.bigdata.com",
          "engine_displacement": "1242",
```

NOTE: Log Stash is still running and loading data into Elastic Search

Count on index CARS:

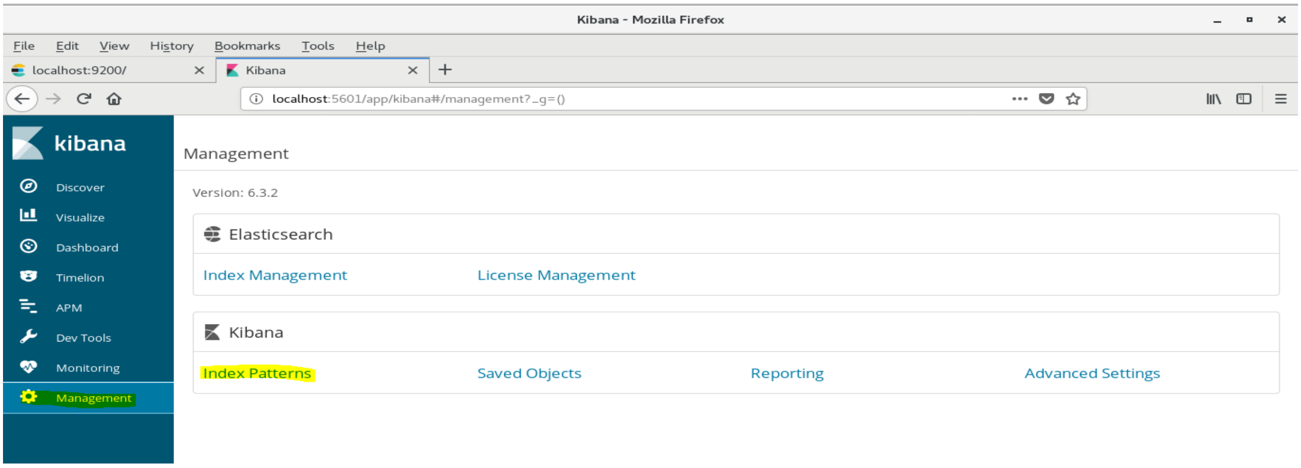


```
GET cars/

GET cars/used_cars/_search
{
  "query" : {
    "match_all" : {}
  }
}

GET cars/used_cars/_count
```

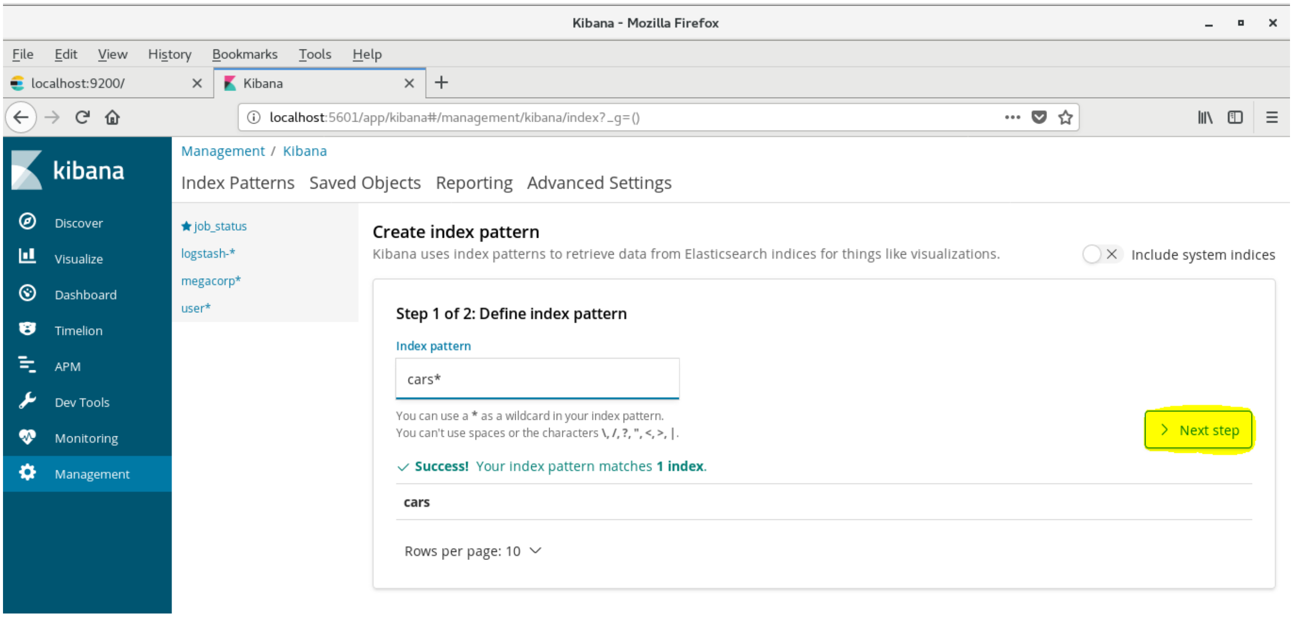
```
{
  "count": 241399,
  "_shards": {
    "total": 5,
    "successful": 5,
    "skipped": 0,
    "failed": 0
  }
}
```

8. Creating an Index Pattern in Kibana:

In Monitoring Section, create the Index Pattern



Creating Index Pattern:



Create index pattern

Kibana uses index patterns to retrieve data from Elasticsearch indices for things like visualizations.

Include system indices

Step 1 of 2: Define index pattern

Index pattern

cars*

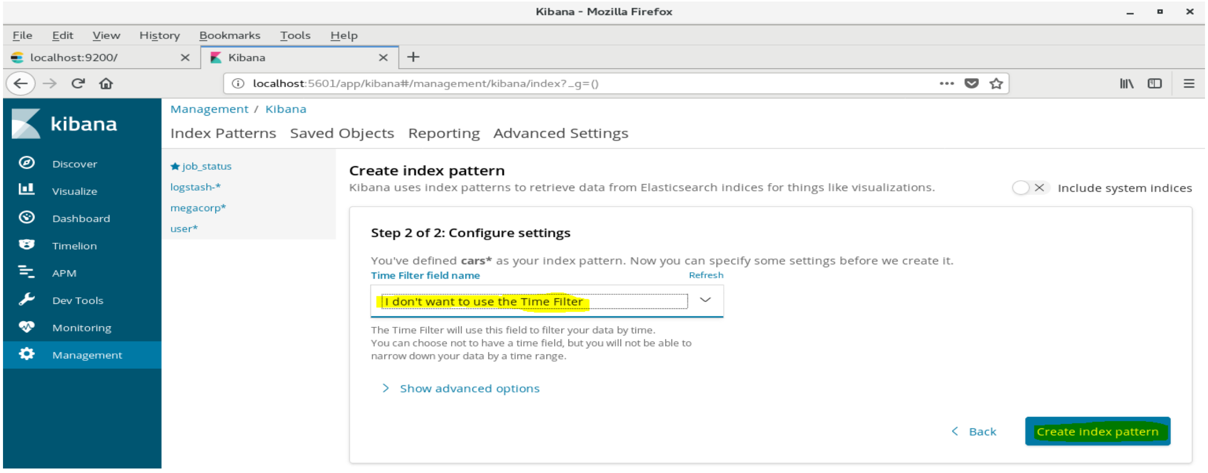
You can use a * as a wildcard in your index pattern.
You can't use spaces or the characters \, /, ?, ", <, >, |.

> Next step

✓ Success! Your index pattern matches 1 index.

cars

Rows per page: 10



Create index pattern

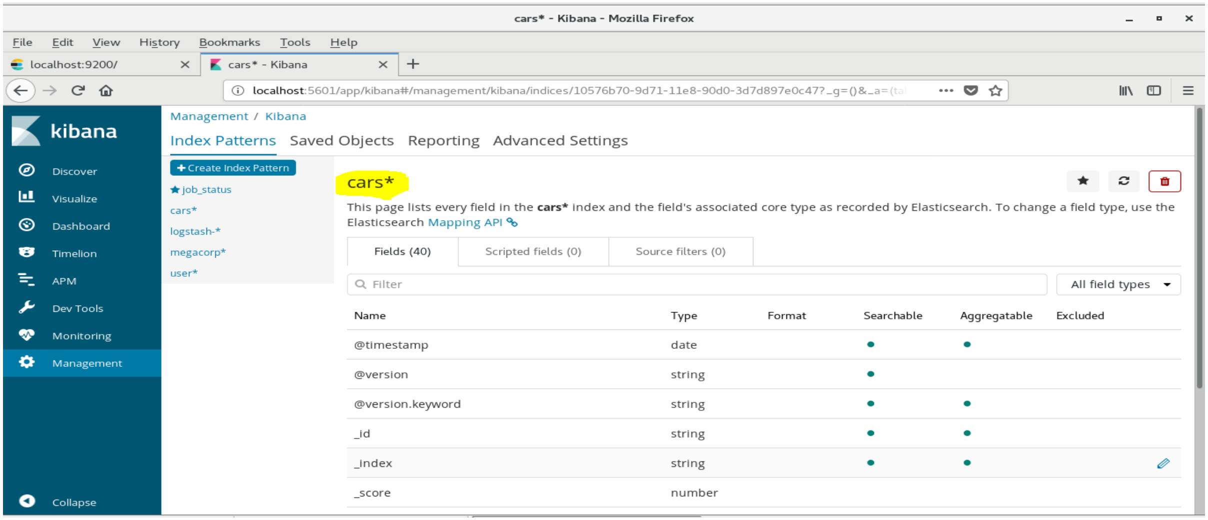
Kibana uses index patterns to retrieve data from Elasticsearch indices for things like visualizations. Include system indices

Step 2 of 2: Configure settings

You've defined **cars*** as your index pattern. Now you can specify some settings before we create it.

Time Filter field name: I don't want to use the Time Filter

The Time Filter will use this field to filter your data by time.
You can choose not to have a time field, but you will not be able to narrow down your data by a time range.


cars* - Kibana - Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:9200/
cars* - Kibana
localhost:5601/app/kibana#/management/kibana/indices/10576b70-9d71-11e8-90d0-3d7d897e0c47?_g=()&_a=(ta
kibana
Discover
Visualize
Dashboard
Timelion
APM
Dev Tools
Monitoring
Management
Collapse
Management / Kibana
Index Patterns Saved Objects Reporting Advanced Settings
Create Index Pattern
job_status
cars*
logstash-*
megacorp*
user*

cars*

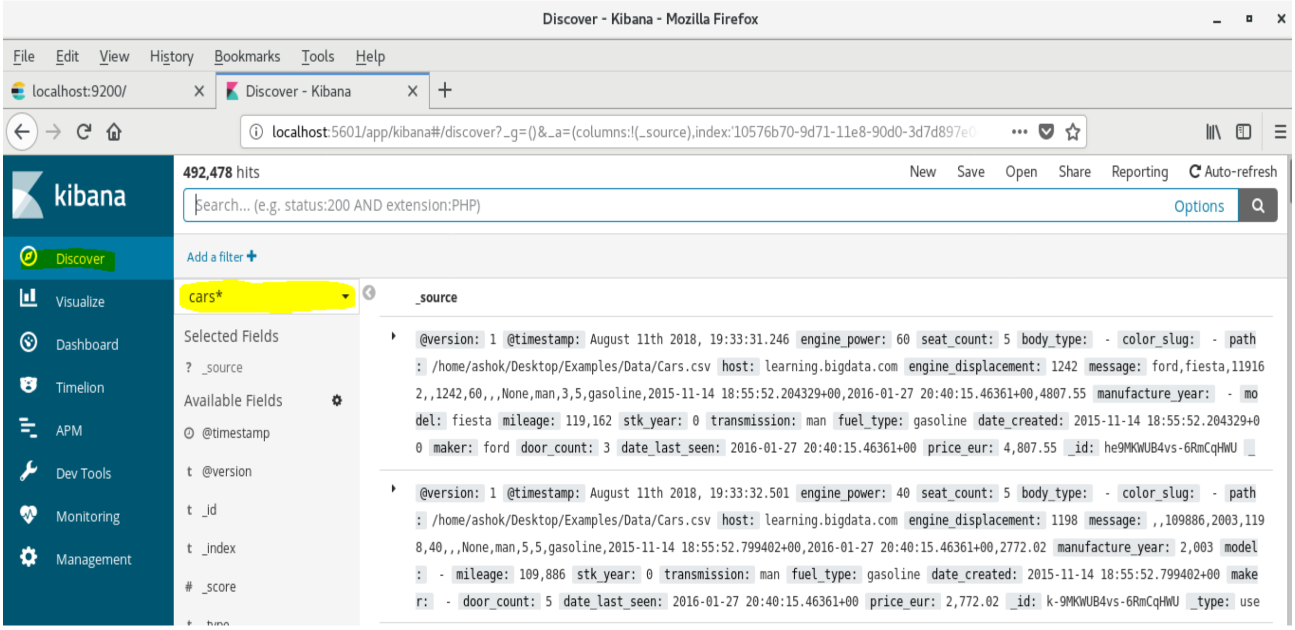
This page lists every field in the **cars*** index and the field's associated core type as recorded by Elasticsearch. To change a field type, use the Elasticsearch Mapping API

Name	Type	Format	Searchable	Aggregatable	Excluded
@timestamp	date		●	●	
@version	string		●		
@version.keyword	string		●	●	
_id	string		●	●	
_index	string		●	●	
_score	number				

CARS Index Pattern is created.

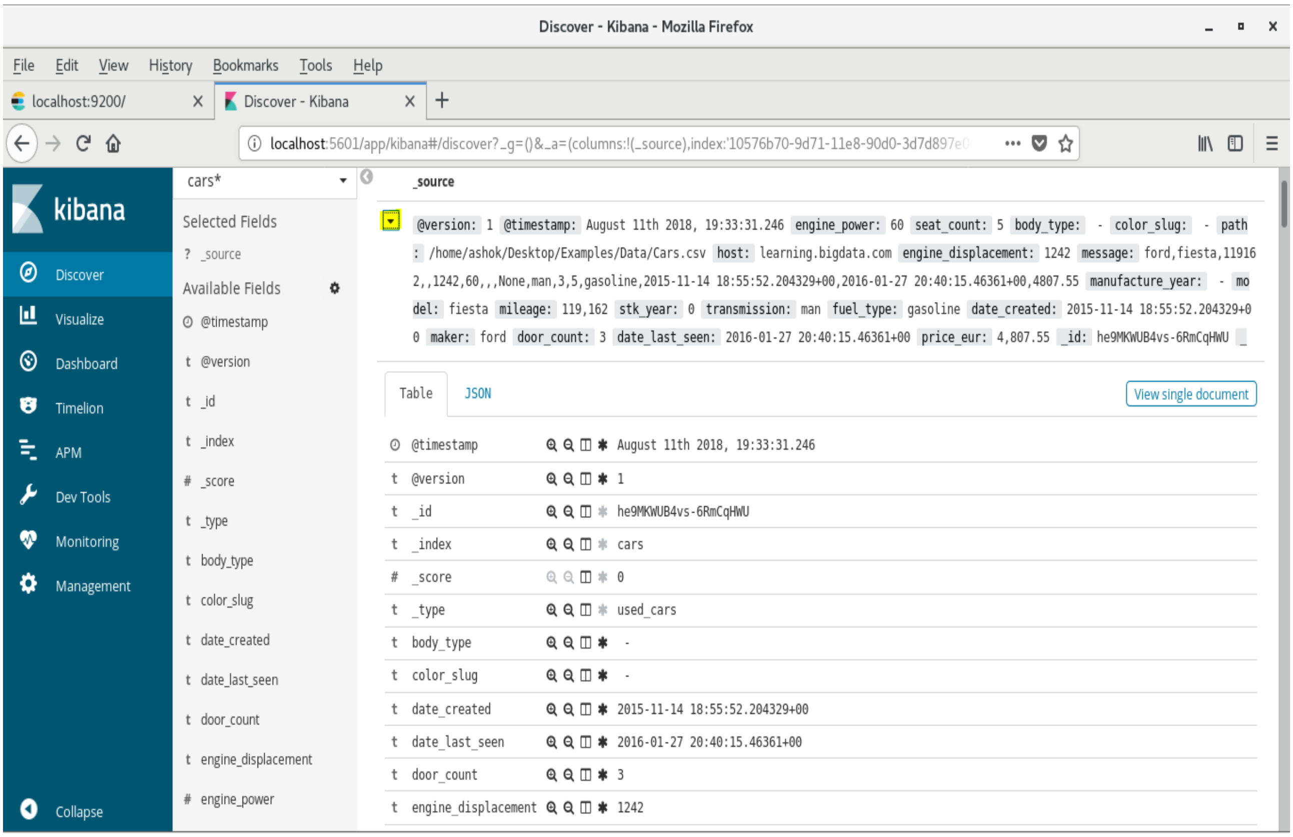
9. Discovering Data in Kibana:

Then the data can be seen in Discover tab by selecting the Index Pattern


Discover - Kibana - Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:9200/
Discover - Kibana
localhost:5601/app/kibana#/discover?_g=()&_a=(columns:!(_source),index:'10576b70-9d71-11e8-90d0-3d7d897e0
kibana
Discover
Visualize
Dashboard
Timelion
APM
Dev Tools
Monitoring
Management
492,478 hits
New Save Open Share Reporting Auto-refresh
Search... (e.g. status:200 AND extension:PHP)
Options
Add a filter +
cars*
Selected Fields
? _source
Available Fields
@timestamp
t @version
t _id
t _index
# _score
t _type

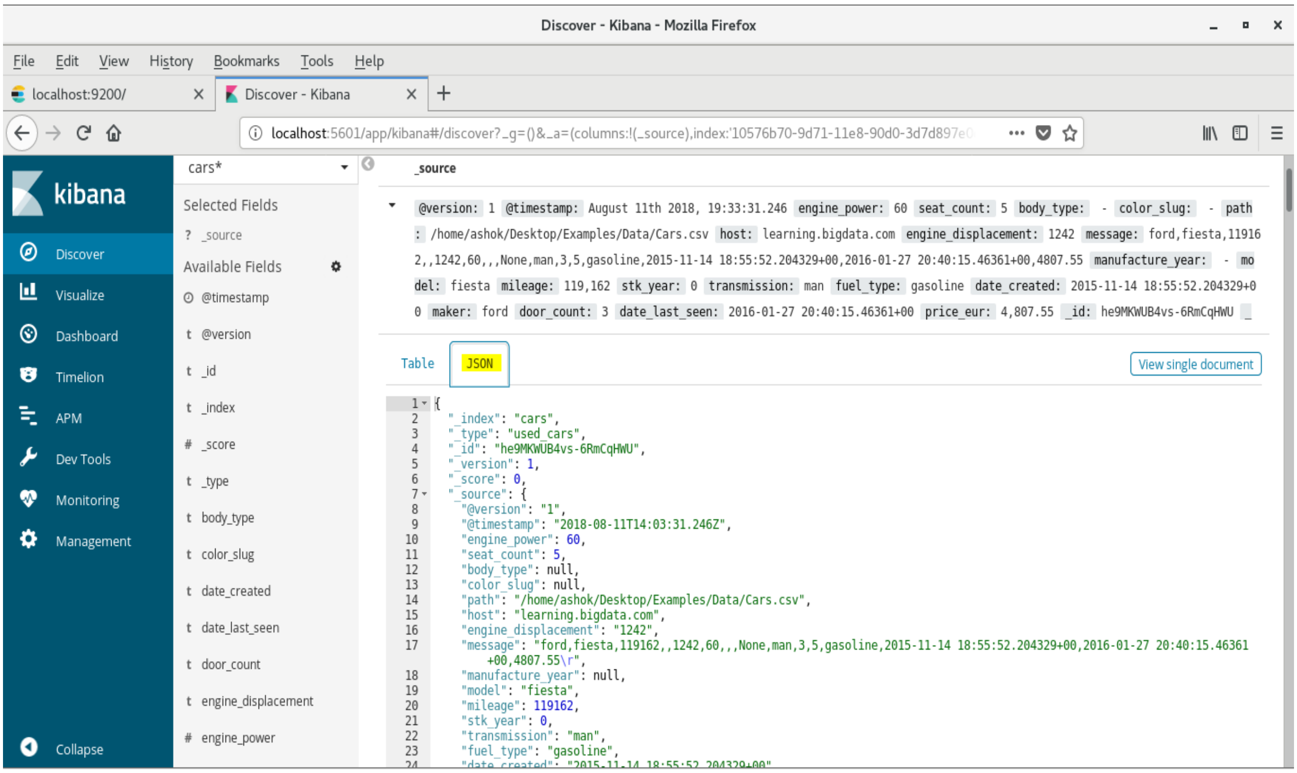
```
_source

@version: 1 @timestamp: August 11th 2018, 19:33:31.246 engine_power: 60 seat_count: 5 body_type:  - color_slug:  - path
: /home/ashok/Desktop/Examples/Data/Cars.csv host: learning.bigdata.com engine_displacement: 1242 message: ford,fiesta,11916
2,,1242,60,,,None,man,3,5,gasoline,2015-11-14 18:55:52.204329+00,2016-01-27 20:40:15.46361+00,4807.55 manufacture_year:  - mo
del: fiesta mileage: 119,162 stk_year: 0 transmission: man fuel_type: gasoline date_created: 2015-11-14 18:55:52.204329+0
0 maker: ford door_count: 3 date_last_seen: 2016-01-27 20:40:15.46361+00 price_eur: 4,807.55 _id: he9MKWUB4vs-6RmCqHWU _

@version: 1 @timestamp: August 11th 2018, 19:33:32.501 engine_power: 40 seat_count: 5 body_type:  - color_slug:  - path
: /home/ashok/Desktop/Examples/Data/Cars.csv host: learning.bigdata.com engine_displacement: 1198 message: ,,109886,2003,119
8,40,,,None,man,5,5,gasoline,2015-11-14 18:55:52.799402+00,2016-01-27 20:40:15.46361+00,2772.02 manufacture_year: 2,003 model
:  - mileage: 109,886 stk_year: 0 transmission: man fuel_type: gasoline date_created: 2015-11-14 18:55:52.799402+00 make
r:  - door_count: 5 date_last_seen: 2016-01-27 20:40:15.46361+00 price_eur: 2,772.02 _id: k-9MKWUB4vs-6RmCqHWU _type: use
```

Can see each record as tabular format or JSON



@version: 1 @timestamp: August 11th 2018, 19:33:31.246 engine_power: 60 seat_count: 5 body_type: - color_slug: - path: /home/ashok/Desktop/Examples/Data/Cars.csv host: learning.bigdata.com engine_displacement: 1242 message: ford,fiesta,119162,,1242,60,,,None,man,3,5,gasoline,2015-11-14 18:55:52.204329+00,2016-01-27 20:40:15.46361+00,4807.55 manufacture_year: - model: fiesta mileage: 119,162 stk_year: 0 transmission: man fuel_type: gasoline date_created: 2015-11-14 18:55:52.204329+00 maker: ford door_count: 3 date_last_seen: 2016-01-27 20:40:15.46361+00 price_eur: 4,807.55 _id: he9MKWUB4vs-6RmCqHWU _

⊙	@timestamp	August 11th 2018, 19:33:31.246
t	@version	1
t	_id	he9MKWUB4vs-6RmCqHWU
t	_index	cars
#	_score	0
t	_type	used_cars
t	body_type	-
t	color_slug	-
t	date_created	2015-11-14 18:55:52.204329+00
t	date_last_seen	2016-01-27 20:40:15.46361+00
t	door_count	3
t	engine_displacement	1242

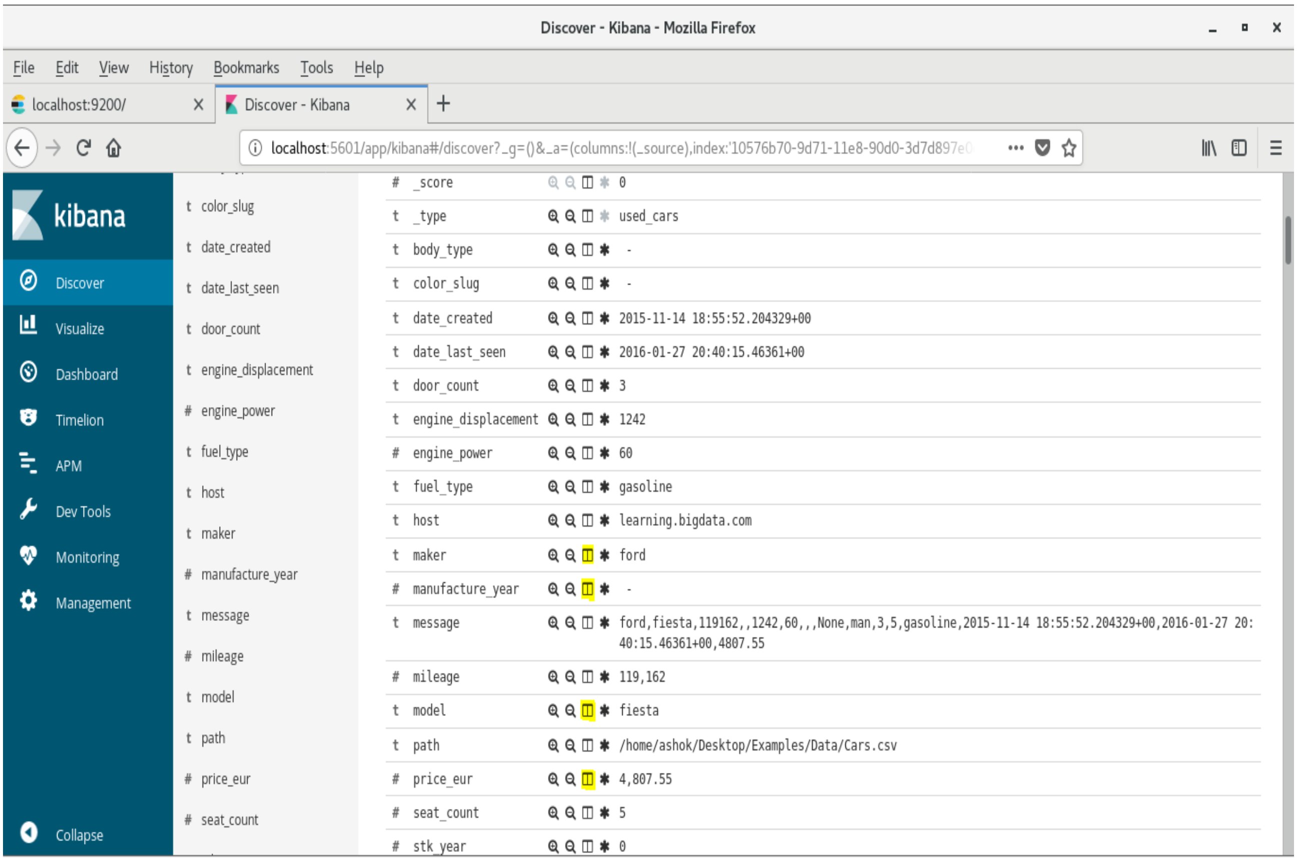


@version: 1 @timestamp: August 11th 2018, 19:33:31.246 engine_power: 60 seat_count: 5 body_type: - color_slug: - path: /home/ashok/Desktop/Examples/Data/Cars.csv host: learning.bigdata.com engine_displacement: 1242 message: ford,fiesta,119162,,1242,60,,,None,man,3,5,gasoline,2015-11-14 18:55:52.204329+00,2016-01-27 20:40:15.46361+00,4807.55 manufacture_year: - model: fiesta mileage: 119,162 stk_year: 0 transmission: man fuel_type: gasoline date_created: 2015-11-14 18:55:52.204329+00 maker: ford door_count: 3 date_last_seen: 2016-01-27 20:40:15.46361+00 price_eur: 4,807.55 _id: he9MKWUB4vs-6RmCqHWU _

```json
{
  "_index": "cars",
  "_type": "used_cars",
  "_id": "he9MKWUB4vs-6RmCqHWU",
  "_version": 1,
  "_score": 0,
  "_source": {
    "@version": "1",
    "@timestamp": "2018-08-11T14:03:31.246Z",
    "engine_power": 60,
    "seat_count": 5,
    "body_type": null,
    "color_slug": null,
    "path": "/home/ashok/Desktop/Examples/Data/Cars.csv",
    "host": "learning.bigdata.com",
    "engine_displacement": "1242",
    "message": "ford,fiesta,119162,,1242,60,,,None,man,3,5,gasoline,2015-11-14 18:55:52.204329+00,2016-01-27 20:40:15.46361
        +00,4807.55\r",
    "manufacture_year": null,
    "model": "fiesta",
    "mileage": 119162,
    "stk_year": 0,
    "transmission": "man",
    "fuel_type": "gasoline",
    "date_created": "2015-11-14 18:55:52.204329+00"
```

Can select only the required columns for all the records as follows:

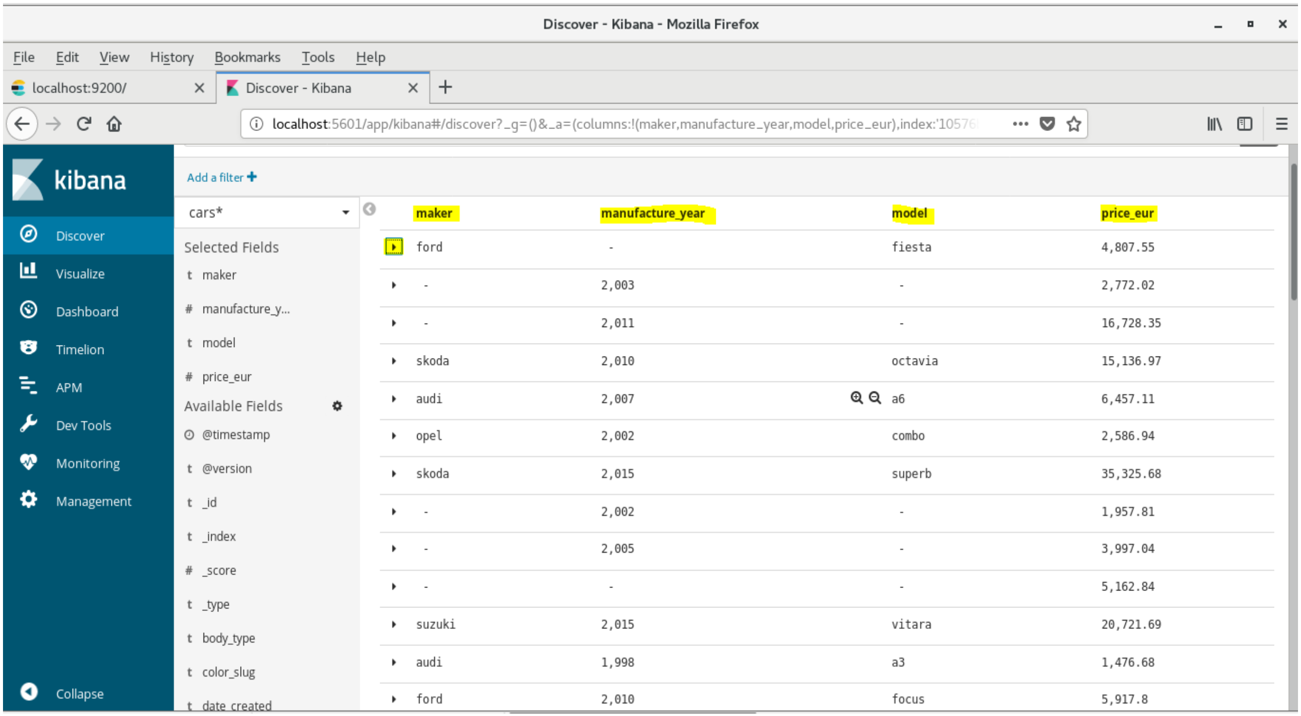
Click on the symbol for the required fields:



localhost:5601/app/kibana#/discover?_g=()&_a=(columns:!(_source),index:'10576b70-9d71-11e8-90d0-3d7d897e0

t color_slug	# _score	0
t date_created	t _type	used_cars
t date_last_seen	t body_type	-
t door_count	t color_slug	-
t engine_displacement	t date_created	2015-11-14 18:55:52.204329+00
# engine_power	t date_last_seen	2016-01-27 20:40:15.46361+00
t fuel_type	t door_count	3
t host	t engine_displacement	1242
t maker	# engine_power	60
# manufacture_year	t fuel_type	gasoline
t message	t host	learning.bigdata.com
# mileage	t maker	ford
t model	# manufacture_year	-
t path	t message	ford,fiesta,119162,,1242,60,,,None,man,3,5,gasoline,2015-11-14 18:55:52.204329+00,2016-01-27 20:40:15.46361+00,4807.55
# price_eur	# mileage	119,162
# seat_count	t model	fiesta
	t path	/home/ashok/Desktop/Examples/Data/Cars.csv
	# price_eur	4,807.55
	# seat_count	5
	# stk_year	0

Then



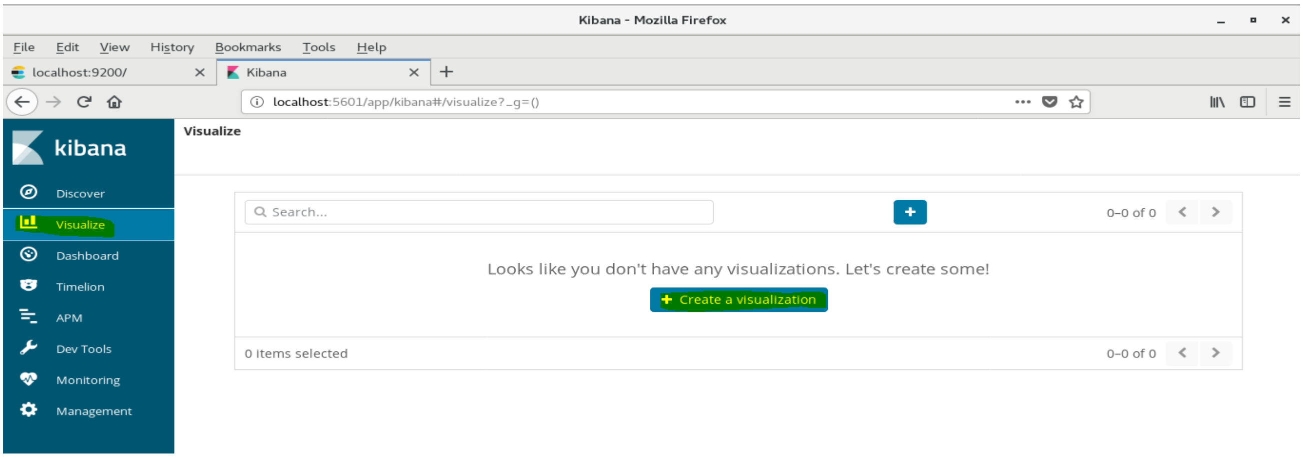
localhost:5601/app/kibana#/discover?_g=()&_a=(columns:!(maker,manufacture_year,model,price_eur),index:'1057

Selected Fields
- t maker
- # manufacture_y...
- t model
- # price_eur

Available Fields
- @timestamp
- t @version
- t _id
- t _index
- # _score
- t _type
- t body_type
- t color_slug
- t date_created

maker	manufacture_year	model	price_eur
ford	-	fiesta	4,807.55
-	2,003	-	2,772.02
-	2,011	-	16,728.35
skoda	2,010	octavia	15,136.97
audi	2,007	a6	6,457.11
opel	2,002	combo	2,586.94
skoda	2,015	superb	35,325.68
-	2,002	-	1,957.81
-	2,005	-	3,997.04
-	-	-	5,162.84
suzuki	2,015	vitara	20,721.69
audi	1,998	a3	1,476.68
ford	2,010	focus	5,917.8

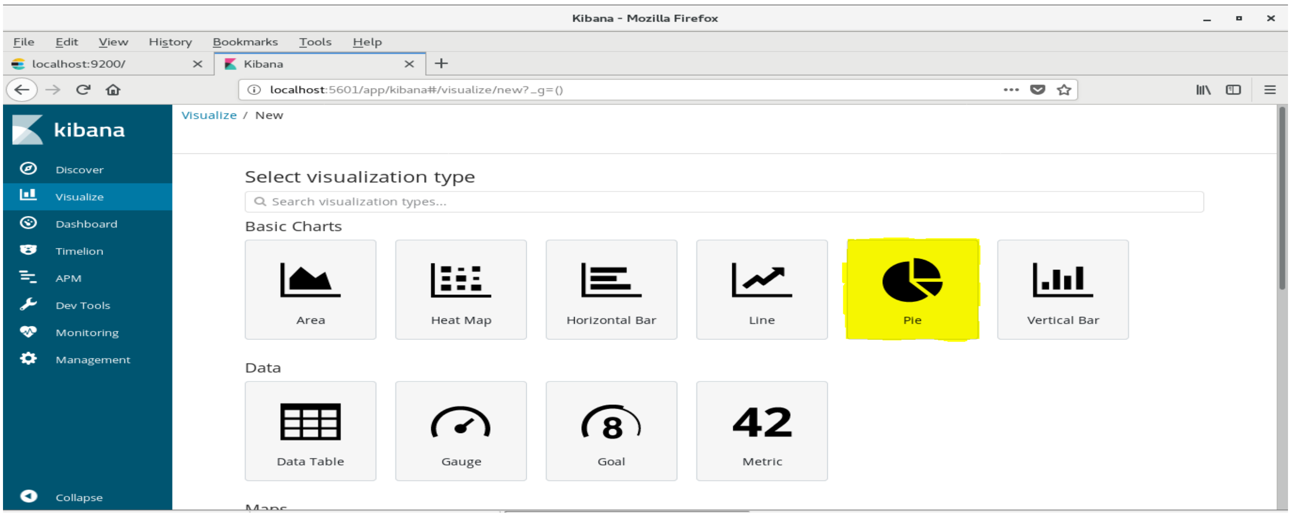
10. Creating Visualization in Kibana:



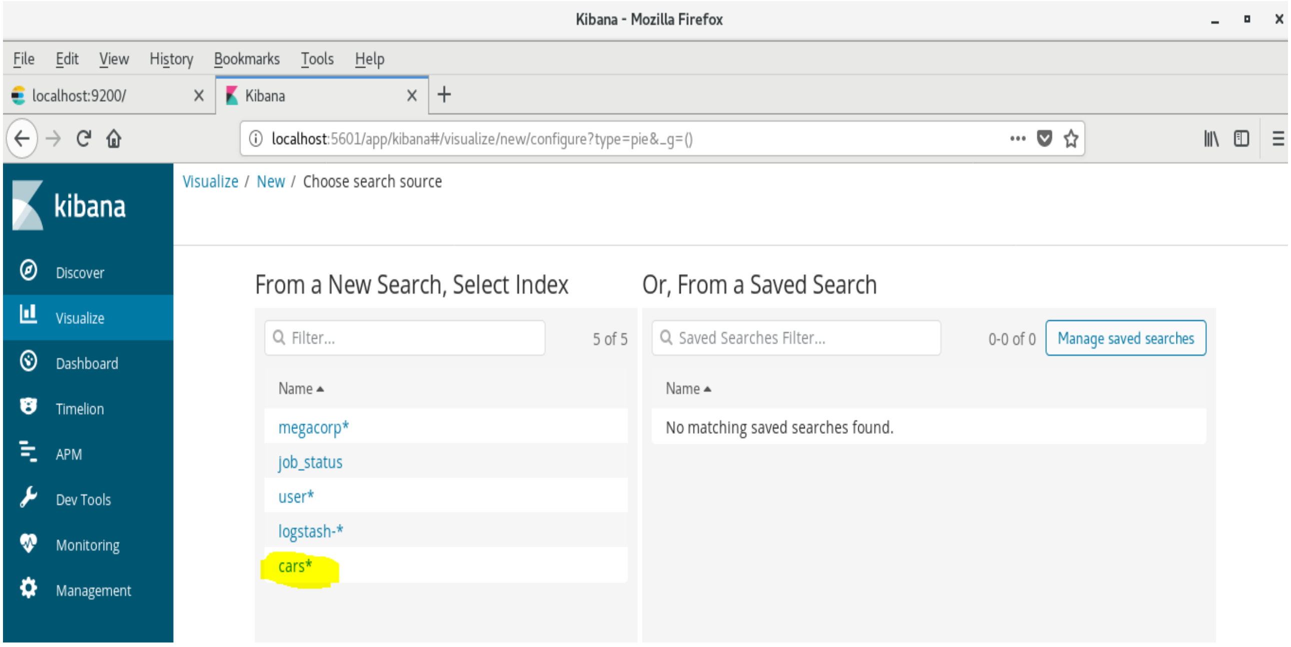
Looks like you don't have any visualizations. Let's create some!

0 items selected

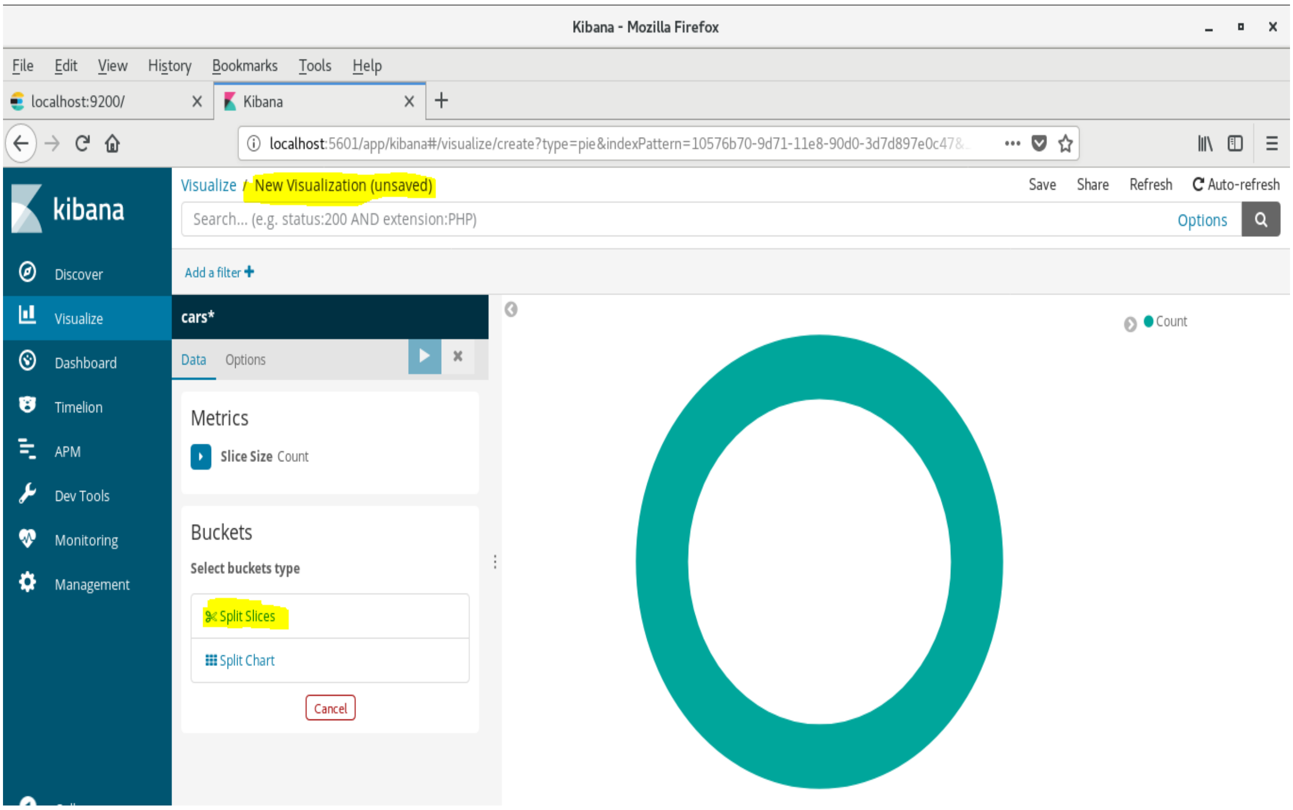
Selecting the PIE Chart:



Select the CARS Index Pattern:



No matching saved searches found.


Kibana - Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:9200/
Kibana
localhost:5601/app/kibana#/visualize/create?type=pie&indexPattern=10576b70-9d71-11e8-90d0-3d7d897e0c47&...
kibana
Discover
Visualize
Dashboard
Timelion
APM
Dev Tools
Monitoring
Management
Visualize / New Visualization (unsaved)
Save Share Refresh Auto-refresh
Search... (e.g. status:200 AND extension:PHP)
Options
Add a filter +
cars*
Data Options
Count

Metrics

Slice Size Count

Buckets

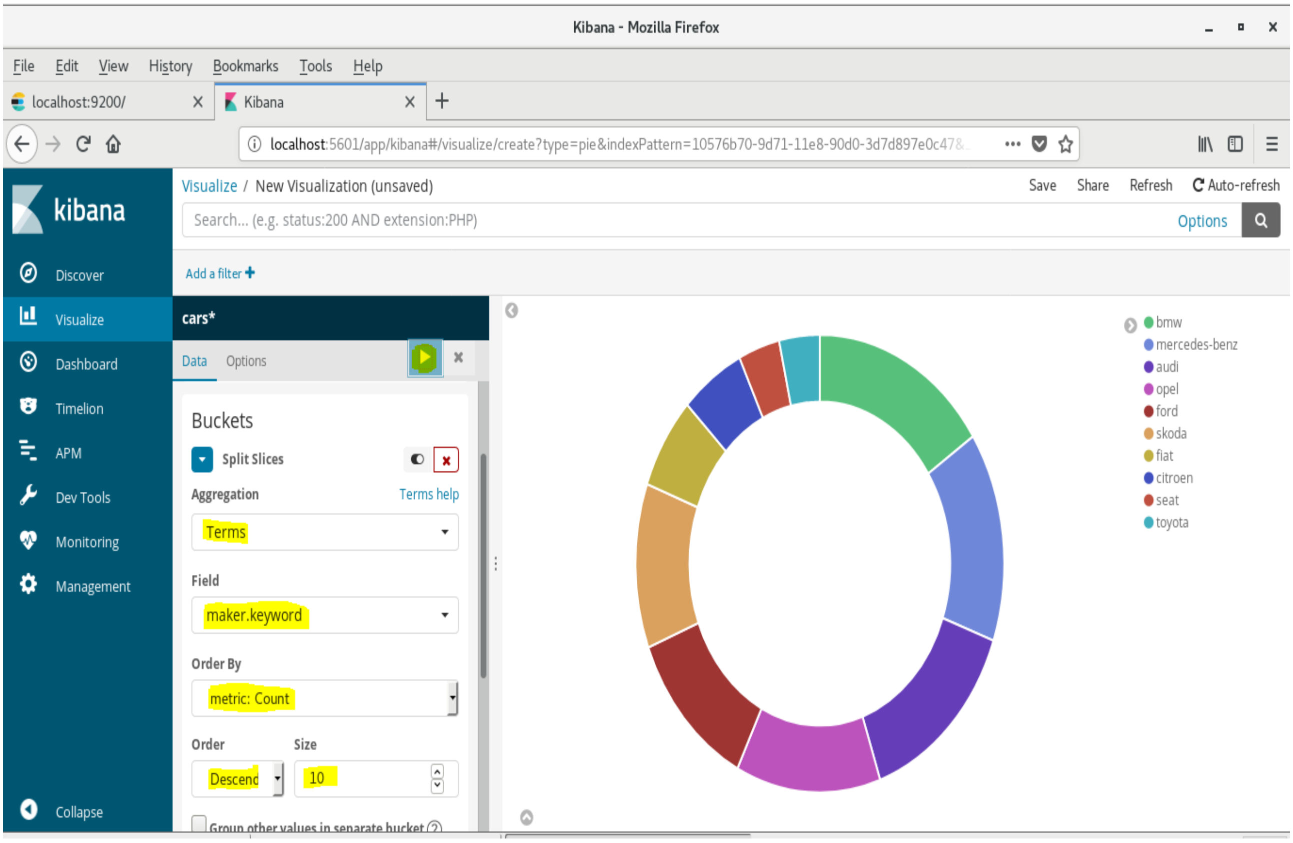
Select buckets type

Split Slices

Split Chart

Cancel

Top 10 Makers of the cars and this is real time, whenever the data inserts into Elastic Search, the PIE chart varies.


Kibana - Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:9200/
Kibana
localhost:5601/app/kibana#/visualize/create?type=pie&indexPattern=10576b70-9d71-11e8-90d0-3d7d897e0c47&...
kibana
Discover
Visualize
Dashboard
Timelion
APM
Dev Tools
Monitoring
Management
Collapse
Visualize / New Visualization (unsaved)
Save Share Refresh Auto-refresh
Search... (e.g. status:200 AND extension:PHP)
Options
Add a filter +
cars*
Data Options
bmw
mercedes-benz
audi
opel
ford
skoda
fiat
citroen
seat
toyota

Buckets

Split Slices

Aggregation: Terms (Terms help)

Field: maker.keyword

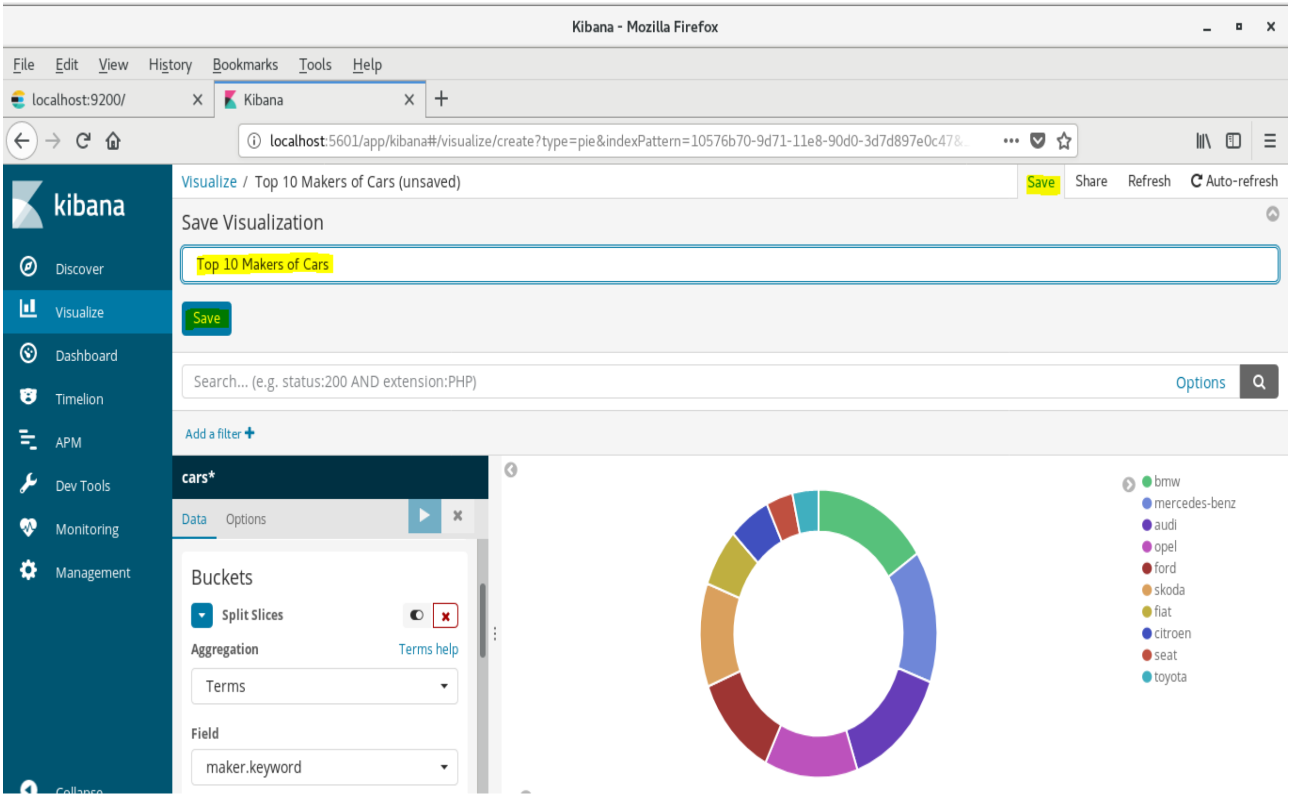
Order By: metric: Count

Order: Descend

Size: 10

□ Group other values in separate bucket

Save the Chart:



Visualize / Top 10 Makers of Cars (unsaved)

Save Visualization

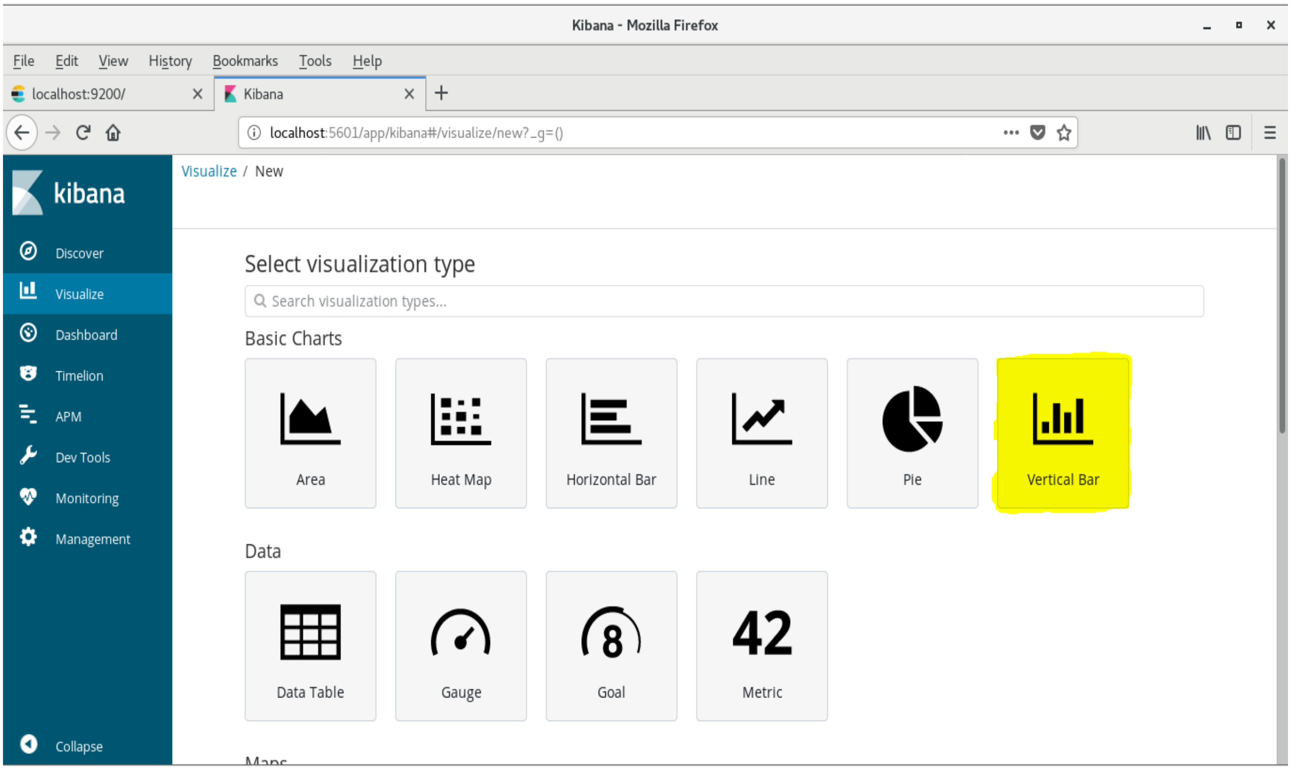
Top 10 Makers of Cars

Save

Aggregation: Terms

Field: maker.keyword

Creating other Visualization Chart:

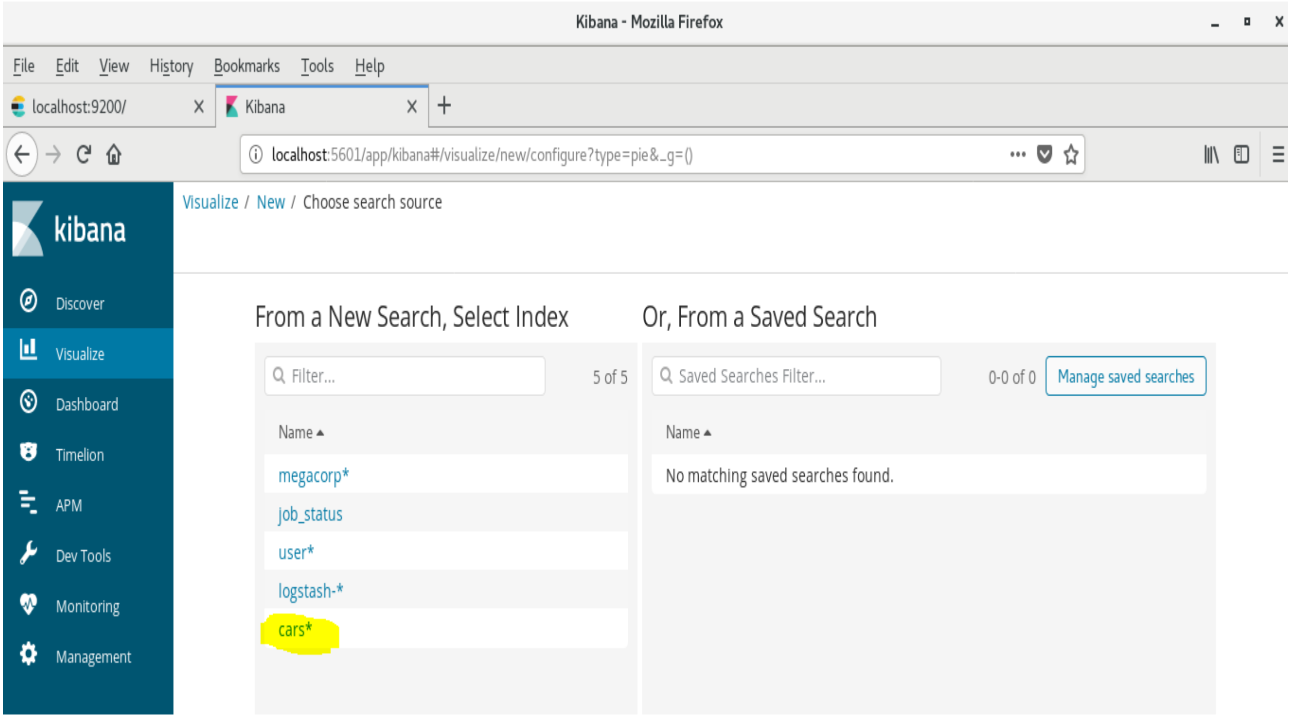


Visualize / New

Select visualization type

Basic Charts

Data


Kibana - Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:9200/
Kibana
localhost:5601/app/kibana#/visualize/new/configure?type=pie&_g=()
kibana
Discover
Visualize
Dashboard
Timelion
APM
Dev Tools
Monitoring
Management
Visualize / New / Choose search source

From a New Search, Select Index

Filter... 5 of 5

Name ▲

- megacorp*
- job_status
- user*
- logstash-*
- cars*

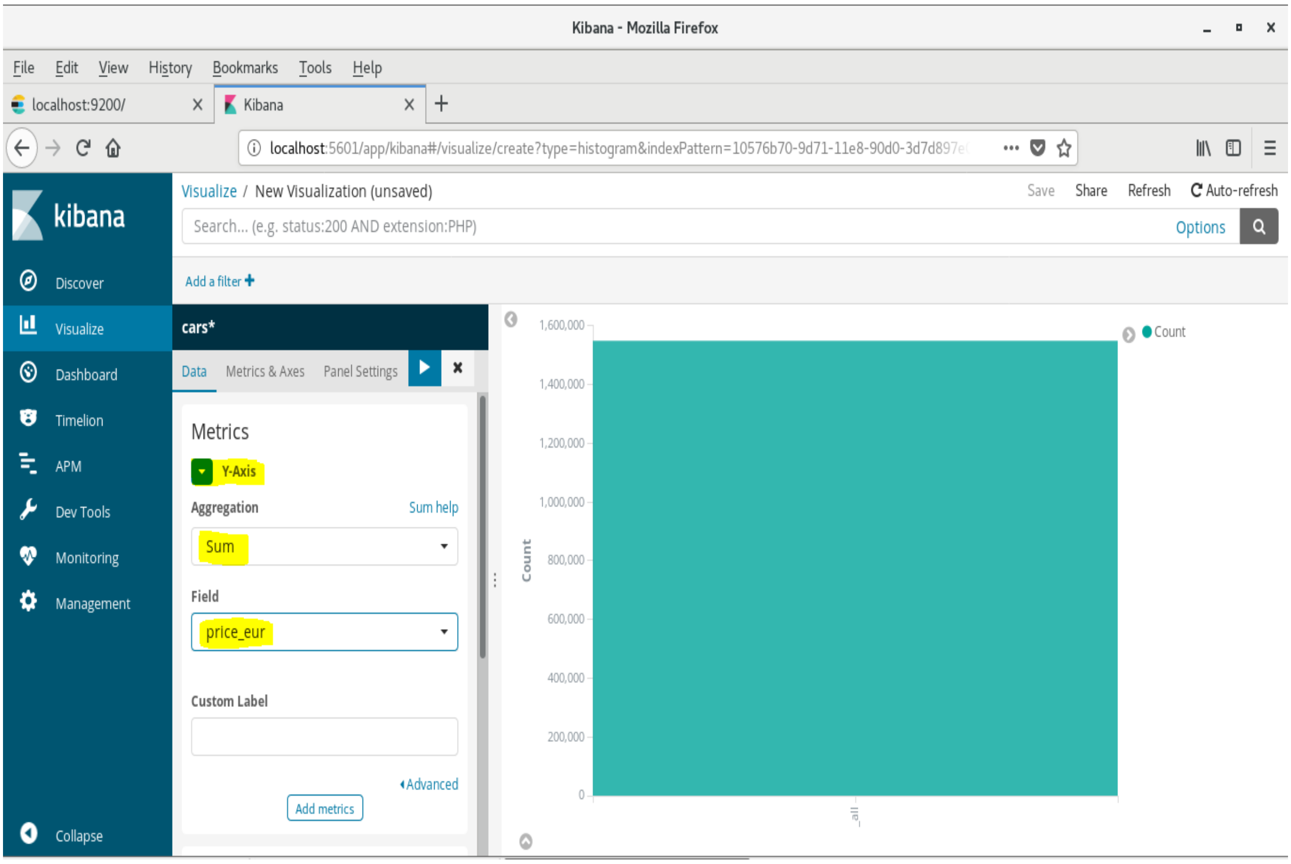
Or, From a Saved Search

Saved Searches Filter... 0-0 of 0 Manage saved searches

Name ▲

No matching saved searches found.

Y-Axis: Sum of Price of the Cars for each Maker



Metrics

Y-Axis

Aggregation: Sum (Sum help)

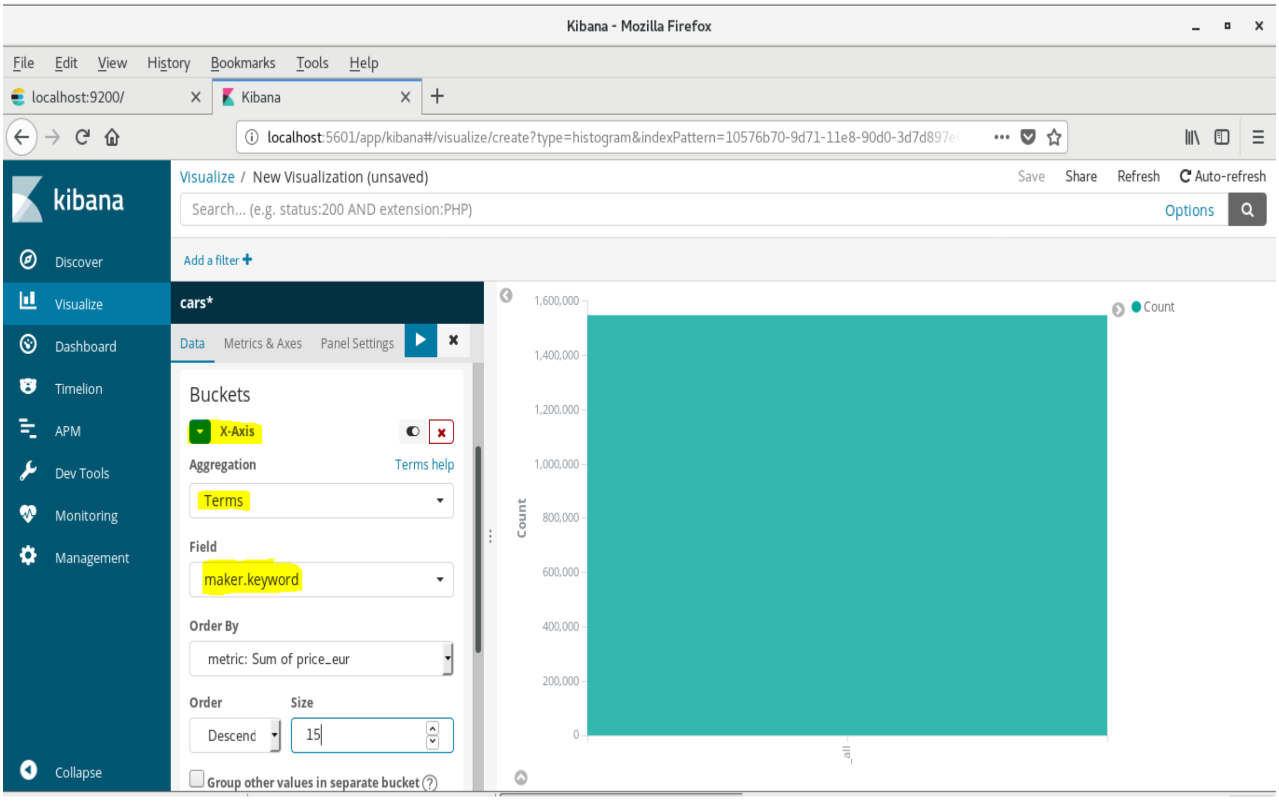
Field: price_eur

Custom Label:

Advanced

Add metrics

X-Axis: Top 15 Makers which make more revenue


Kibana - Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:9200/
Kibana
localhost:5601/app/kibana#/visualize/create?type=histogram&indexPattern=10576b70-9d71-11e8-90d0-3d7d897e
kibana
Discover
Visualize
Dashboard
Timelion
APM
Dev Tools
Monitoring
Management
Collapse
Visualize / New Visualization (unsaved)
Save Share Refresh Auto-refresh
Search... (e.g. status:200 AND extension:PHP)
Options
Add a filter
cars*
Data Metrics & Axes Panel Settings
1,600,000
1,400,000
1,200,000
1,000,000
800,000
600,000
400,000
200,000
0
Count
all
Count

Buckets

X-Axis

Aggregation: Terms

Terms help

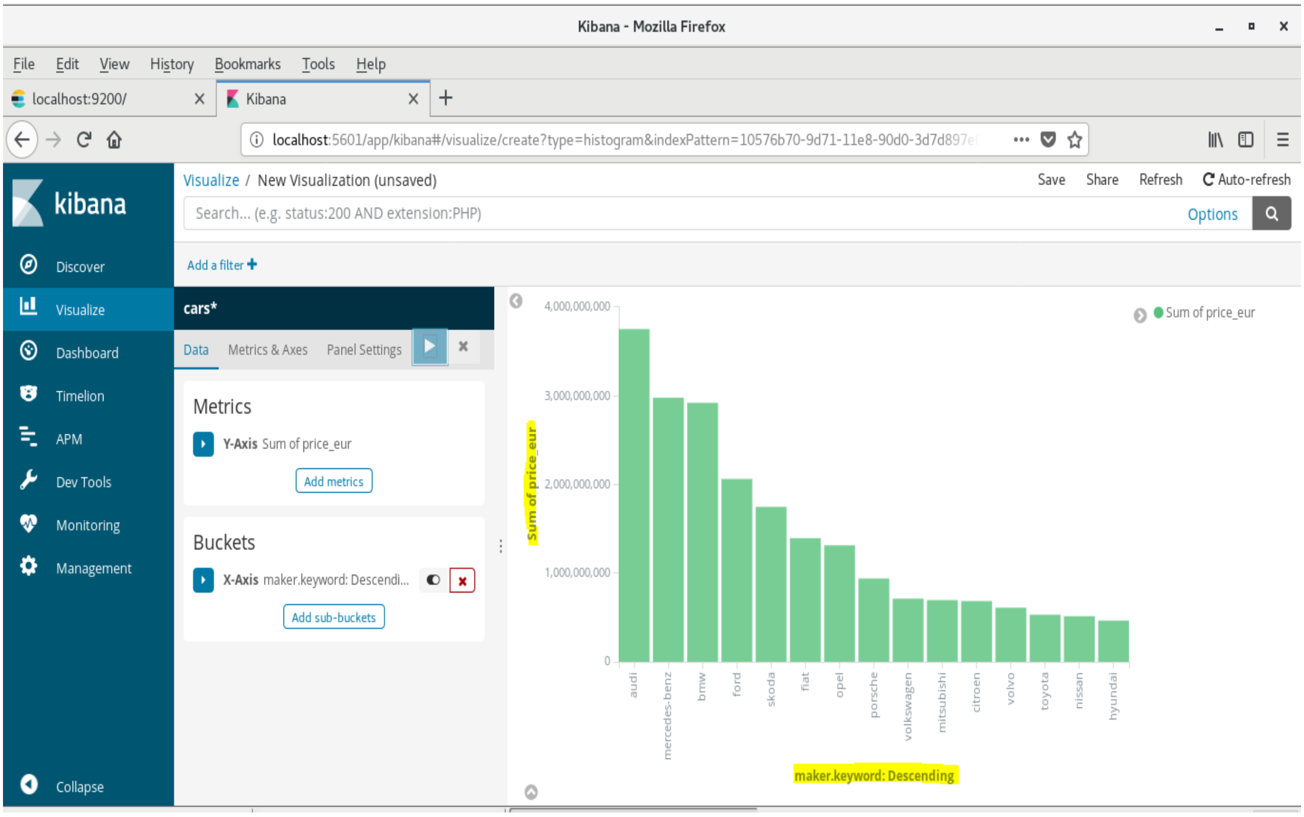
Field: maker.keyword

Order By: metric: Sum of price_eur

Order: Descend

Size: 15

□ Group other values in separate bucket



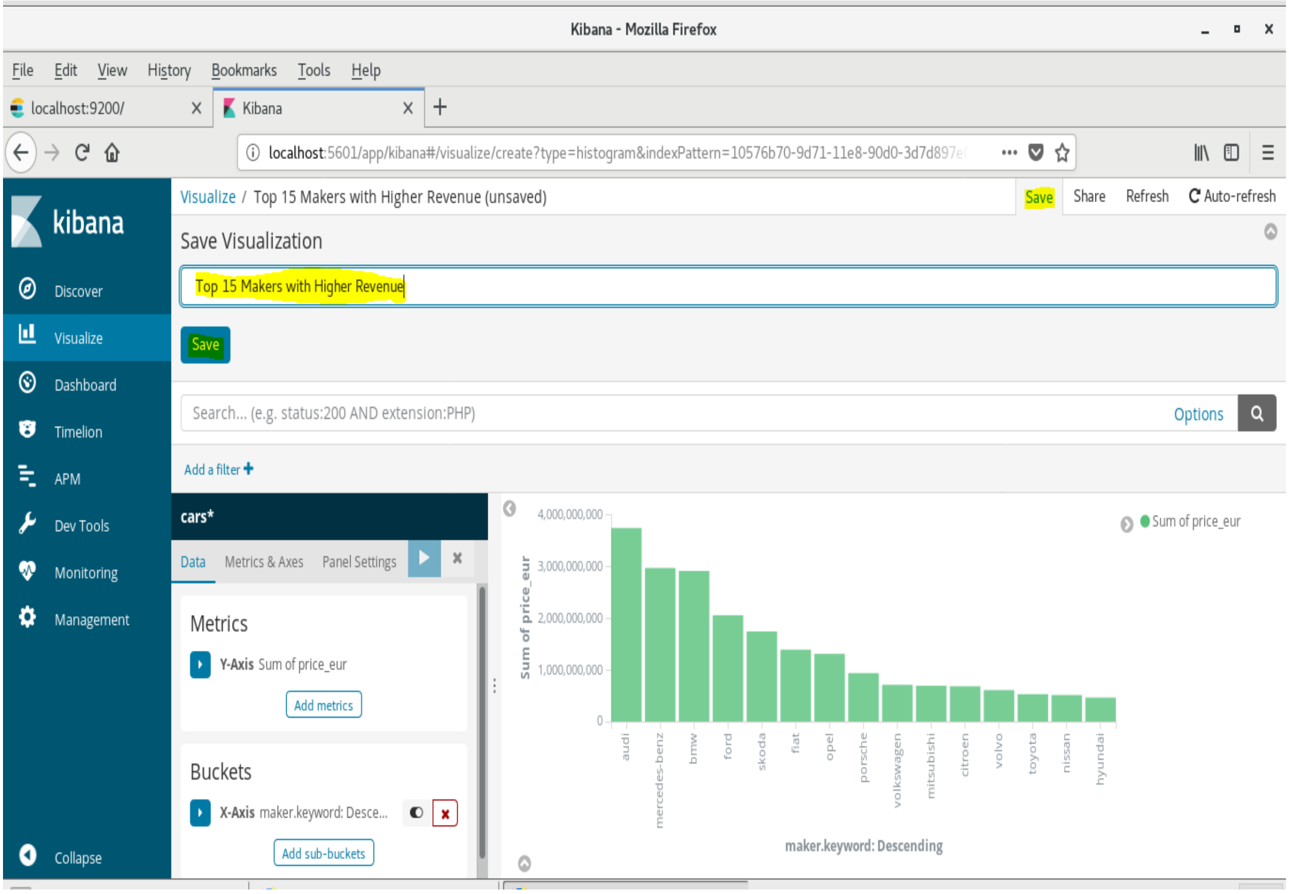
Metrics

Y-Axis Sum of price_eur

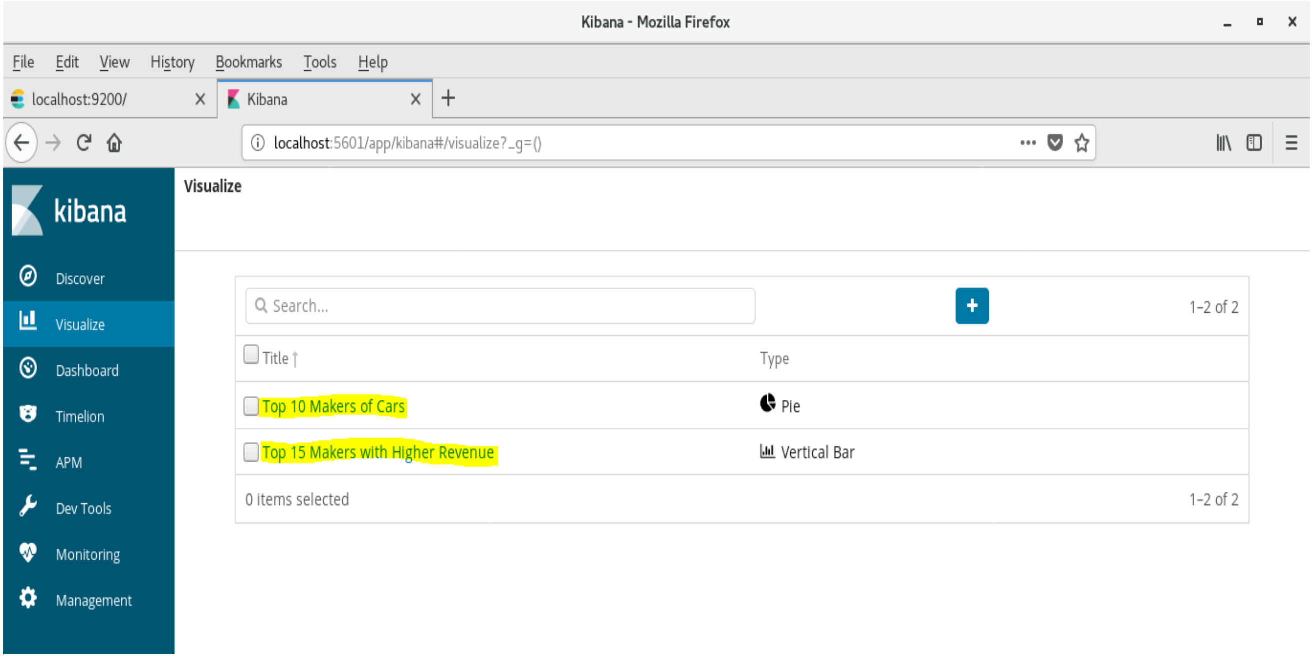
Buckets

X-Axis maker.keyword: Descendi...

Save bar plot.



The list of Visualizations can be seen in the Visualize tab.

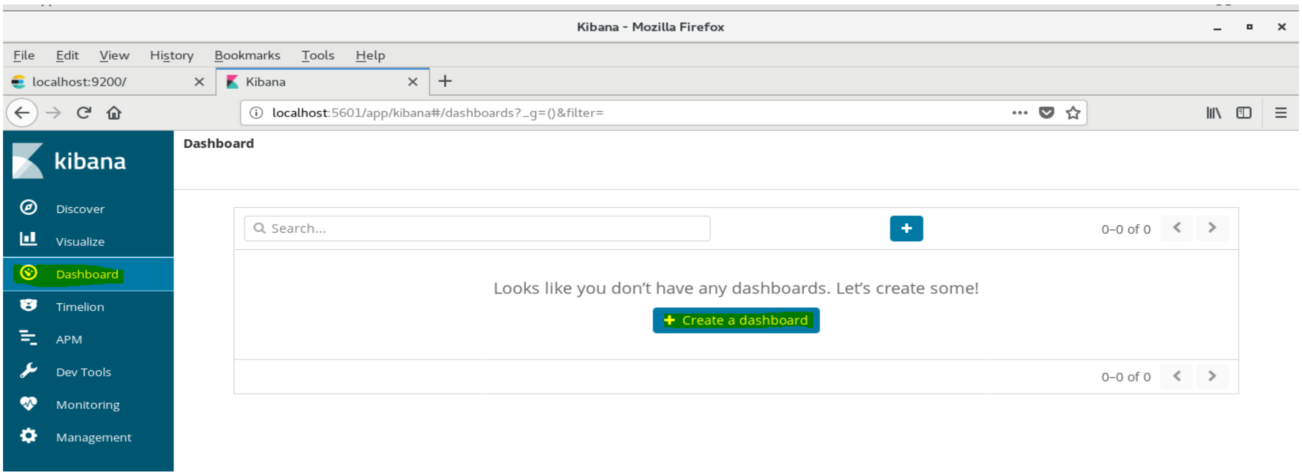


Title ↑	Type
□ Top 10 Makers of Cars	Pie
□ Top 15 Makers with Higher Revenue	Vertical Bar

Select the Visualizations to create the Dash Board.

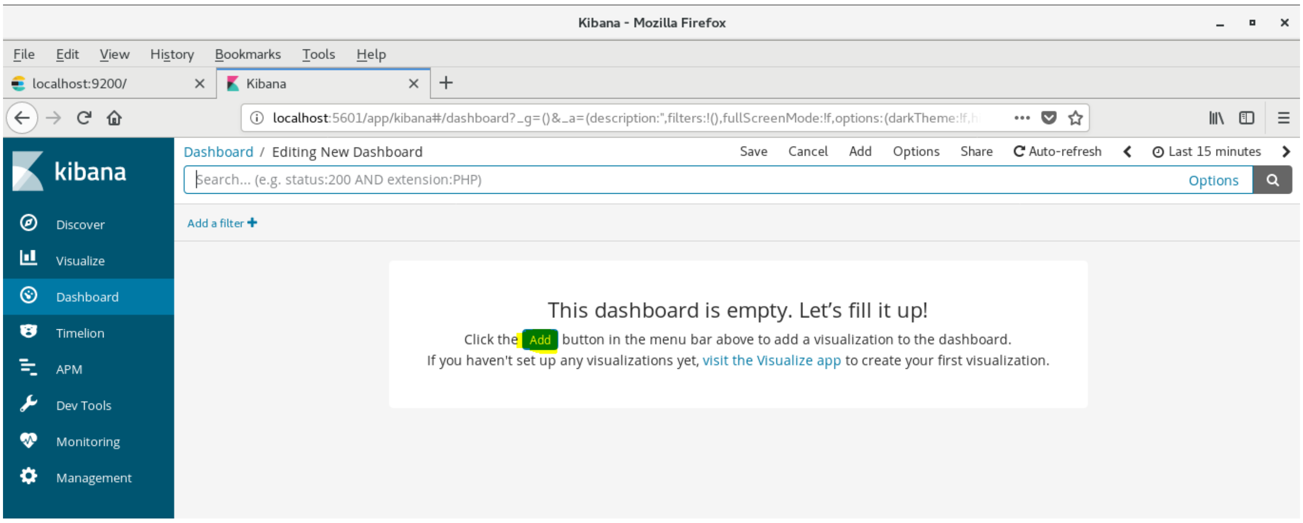
11. Creating Dashboard in Kibana:

Dashboards can be created from the Dashboard tab.



Looks like you don't have any dashboards. Let's create some!

Add the Visualization.

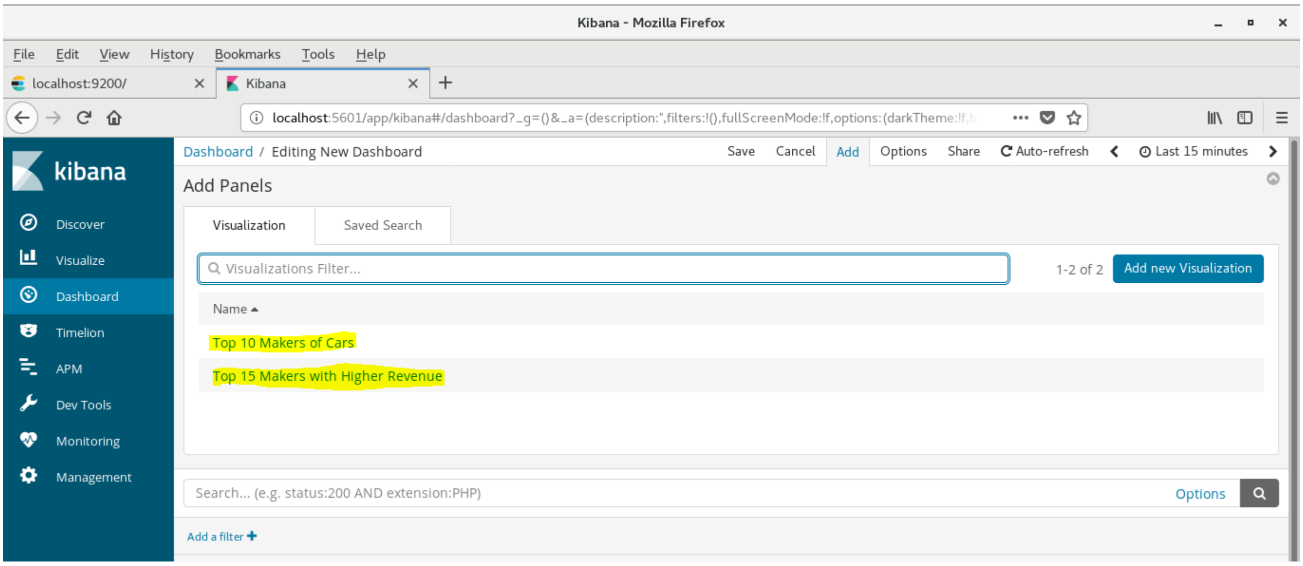

Kibana - Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:9200/
Kibana
localhost:5601/app/kibana#/dashboard?_g=()&_a=(description:'',filters:!(),fullScreenMode:!f,options:(darkTheme:!f,h
kibana
Discover
Visualize
Dashboard
Timelion
APM
Dev Tools
Monitoring
Management
Dashboard / Editing New Dashboard
Save
Cancel
Add
Options
Share
Auto-refresh
Last 15 minutes
Search... (e.g. status:200 AND extension:PHP)
Options
Add a filter +

This dashboard is empty. Let's fill it up!

Click the Add button in the menu bar above to add a visualization to the dashboard.

If you haven't set up any visualizations yet, visit the Visualize app to create your first visualization.

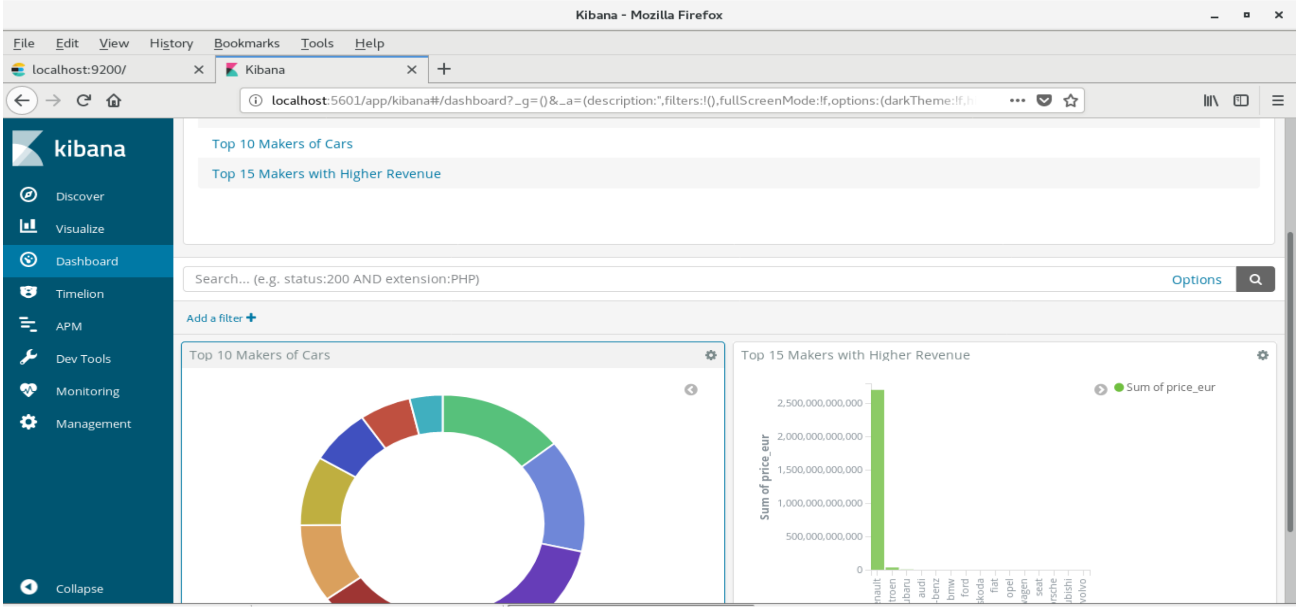
Select the Visualizations:

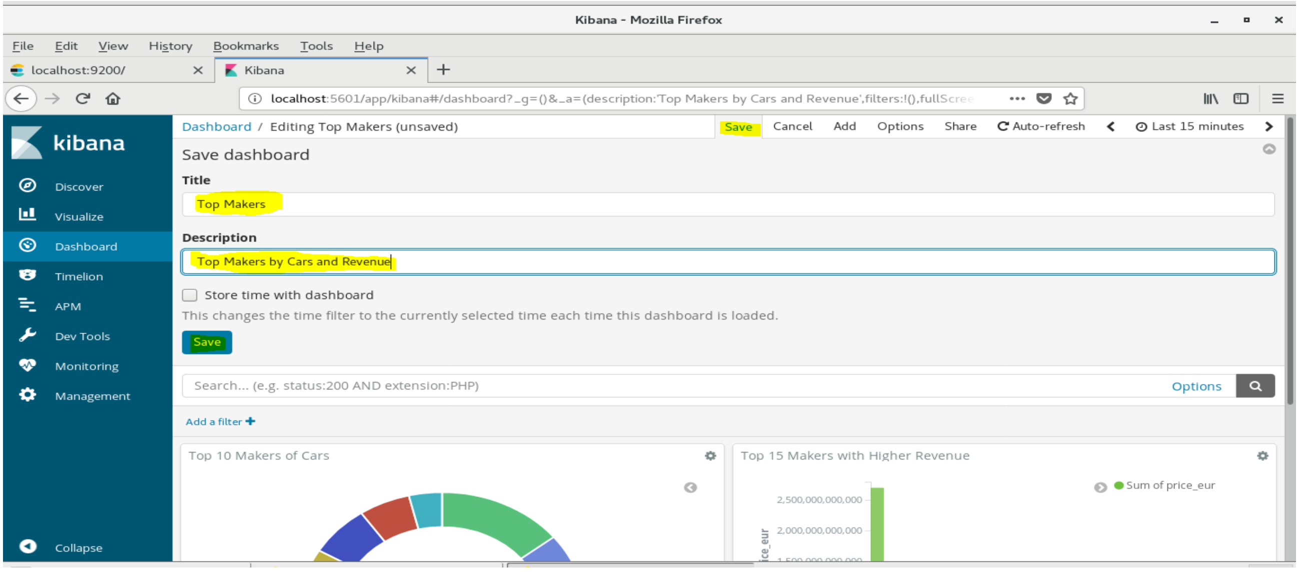


Top 10 Makers of Cars

Top 15 Makers with Higher Revenue

After Selecting Visualizations, can view the Visualizations as follows:




Kibana - Mozilla Firefox
File Edit View History Bookmarks Tools Help
localhost:9200/
Kibana
localhost:5601/app/kibana#/dashboard?_g=()&_a=(description:'Top Makers by Cars and Revenue',filters:!(),fullScre
kibana
Discover
Visualize
Dashboard
Timelion
APM
Dev Tools
Monitoring
Management
Collapse
Dashboard / Editing Top Makers (unsaved)
Save
Cancel
Add
Options
Share
Auto-refresh
Last 15 minutes

Save dashboard

Title: Top Makers

Description: Top Makers by Cars and Revenue

☐ Store time with dashboard

This changes the time filter to the currently selected time each time this dashboard is loaded.

End of the tutorial.