

아파트정보를 이용한 ELK Stack 활용

오근문

System Configuration





SQLite – apt_info.db:table

```
CREATE TABLE "apt_info" (  
    `id`            INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT UNIQUE,  
    `si`           TEXT,  
    `gu`           TEXT,  
    `dong`        TEXT,  
    `aptname`     TEXT,  
    `dong_cnt`    INTEGER,  
    `date`        TEXT,  
    `type`        TEXT,  
    `cnt1`        INTEGER,  
    `cnt2`        INTEGER,  
    `cnt3`        INTEGER,  
    `cnt4`        INTEGER,  
    `cntAll`      INTEGER  
)
```



SQLite – apt_info.db:data

	id	si	gu	dong	aptname	dong_cnt	date	type	cnt1	cnt2	cnt3	cnt4	cntAll
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	서울특별시	강남구	개포동	LG개포자이아파트	4	2004-06-17	계단식	0	0	0	212	212
2	2	서울특별시	강남구	개포동	SH공사대치1단지	8	1991-11-20	복도식	1623	0	0	0	1623
3	3	서울특별시	강남구	개포동	개포2차 현대아파트	13	1986-01-31	계단식	0	0	112	446	558
4	4	서울특별시	강남구	개포동	개포경남아파트	9	1984-11-01	계단식	0	0	198	480	678
5	5	서울특별시	강남구	개포동	개포대치2단지	11	1992-10-20	복도식	1176	577	0	0	1753
6	6	서울특별시	강남구	개포동	개포시영	30	1984-03-20	계단식	1780	190	0	0	1970
7	7	서울특별시	강남구	개포동	개포우성3차	5	1984-12-22	혼합식	0	0	135	270	405
8	8	서울특별시	강남구	개포동	개포우성8차	3	1987-09-21	혼합식	0	0	261	0	261
9	9	서울특별시	강남구	개포동	개포우성9차	2	1991-01-25	계단식	0	0	232	0	232
10	10	서울특별시	강남구	개포동	개포주공1단지	124	1982-06-04	계단식	5040	0	0	0	5040



logstash - plugin

JDBC Input Plugin

- \$ logstash-plugin install logstash-input-jdbc

Elasticsearch Output Plugin

- \$ logstash-plugin install logstash-output-elasticsearch

SQLite Driver download - <http://www.sqlitetutorial.net/sqlite-java/sqlite-jdbc-driver/>

Plugin 정보- <https://www.elastic.co/guide/en/logstash/current/input-plugins.html>



logstash - sqlite_to_ES.conf:input

```
input {
  jdbc {
    jdbc_driver_library => "sqlite-jdbc-3.8.11.2.jar"
    jdbc_driver_class => "org.sqlite.JDBC"
    jdbc_connection_string => "jdbc:sqlite:apt_info.db"
    jdbc_user => "sqlite"
    statement => "select si, gu, dong, aptname, dong_cnt, date, type, cnt1, cnt2, cnt3, cnt4,
cntAll from apt_info"
  }
}
```



logstash - sqlite_to_ES.conf:filter

```
filter {  
  mutate {  
    # field name should be lowercase.  
    remove_field => [ "@version", "@timestamp"  
    rename => { "si" => "시" }  
    rename => { "gu" => "구" }  
    rename => { "dong" => "동" }  
    ....  
    rename => { "cnt4" => "135초과" }  
    rename => { "cntall" => "세대수" }  
  }  
}
```



logstash - output

```
output {  
  elasticsearch {  
    index => "apt_info"  
    document_type => "apt_data"  
  }  
}
```




elasticsearch. - apt_info.json: analysis

```
{
  "settings": {
    "analysis": {
      [CJK Analyzer]
    }
  },
  "mappings": {
    ...
  }
}
```

CJK Analyzer - <https://www.elastic.co/guide/en/elasticsearch/reference/current/analysis-lang-analyzer.html#jdk-analyzer>



elasticsearch. - apt_info.json: mapping

```
"mappings": {
  "apt_data": {
    "_source": { "enabled": true },
    "_all": { "enabled": true, "index": "analyzed", "analyzer": "cjk" },
    "include_in_all": false,
    "dynamic": false,
    "properties": {
      "시": { "type": "string", "index": "not_analyzed", "include_in_all": true },
      "구": { "type": "string", "index": "not_analyzed", "include_in_all": true },
      "동": { "type": "string", "index": "not_analyzed", "include_in_all": true },
      "아파트명": { "type": "string", "index": "not_analyzed", "include_in_all": true },
      "동수": { "type": "integer", "index": "not_analyzed" },
      "시공일": { "type": "date", "index": "not_analyzed", "format": "YYYY-MM-dd" },
      "복도유형": { "type": "string", "index": "not_analyzed" },
      "under60": { "type": "integer", "index": "not_analyzed" },
      "60to85": { "type": "integer", "index": "not_analyzed" },
      "85to135": { "type": "integer", "index": "not_analyzed" },
      "over135": { "type": "integer", "index": "not_analyzed" },
      "세대수": { "type": "integer", "index": "not_analyzed" }
    }
  }
}
```



logstash - run

Create Index

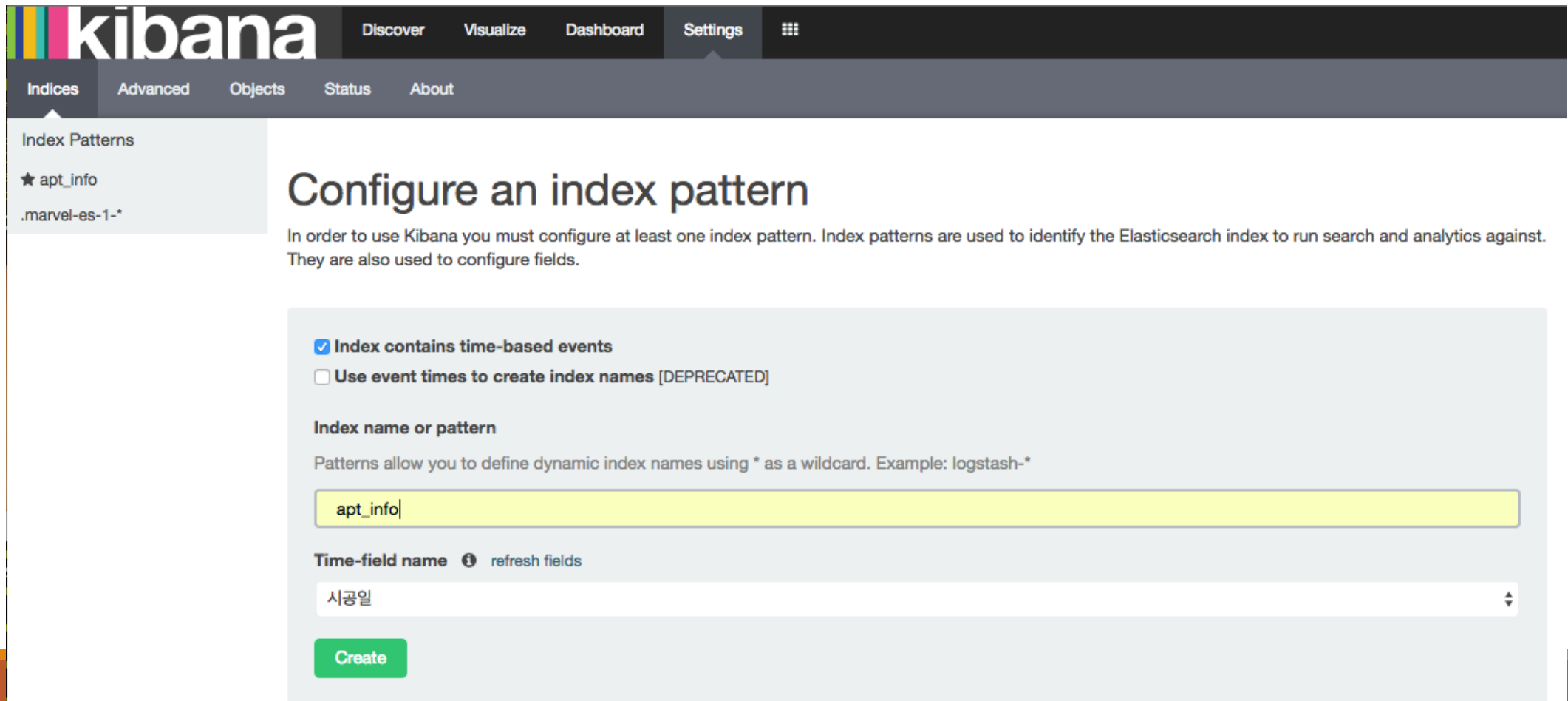
- `$curl -XPOST localhost:9200/apt_info -d @apt_info.json`

Run Logstash

- `$logstash agent -f apt_info_transport_to_ES.conf`

kibana - setting

Index name 및 시간 필드 설정



The screenshot shows the Kibana web interface. At the top, there is a navigation bar with the Kibana logo and menu items: Discover, Visualize, Dashboard, Settings, and a hamburger menu icon. Below this is a sub-navigation bar with options: Indices, Advanced, Objects, Status, and About. The 'Indices' menu is expanded, showing a list of index patterns: '★ apt_info' and '.marvel-es-1-*'. The main content area is titled 'Configure an index pattern'. It contains an introductory paragraph: 'In order to use Kibana you must configure at least one index pattern. Index patterns are used to identify the Elasticsearch index to run search and analytics against. They are also used to configure fields.' Below this, there are two checkboxes: 'Index contains time-based events' (checked) and 'Use event times to create index names [DEPRECATED]' (unchecked). The 'Index name or pattern' section has a text input field containing 'apt_info|'. Below that, the 'Time-field name' section has a dropdown menu with '시공일' selected and a 'refresh fields' link. At the bottom of the configuration area is a green 'Create' button.

Configure an index pattern

In order to use Kibana you must configure at least one index pattern. Index patterns are used to identify the Elasticsearch index to run search and analytics against. They are also used to configure fields.

Index contains time-based events

Use event times to create index names [DEPRECATED]

Index name or pattern

Patterns allow you to define dynamic index names using * as a wildcard. Example: logstash-*

apt_info|

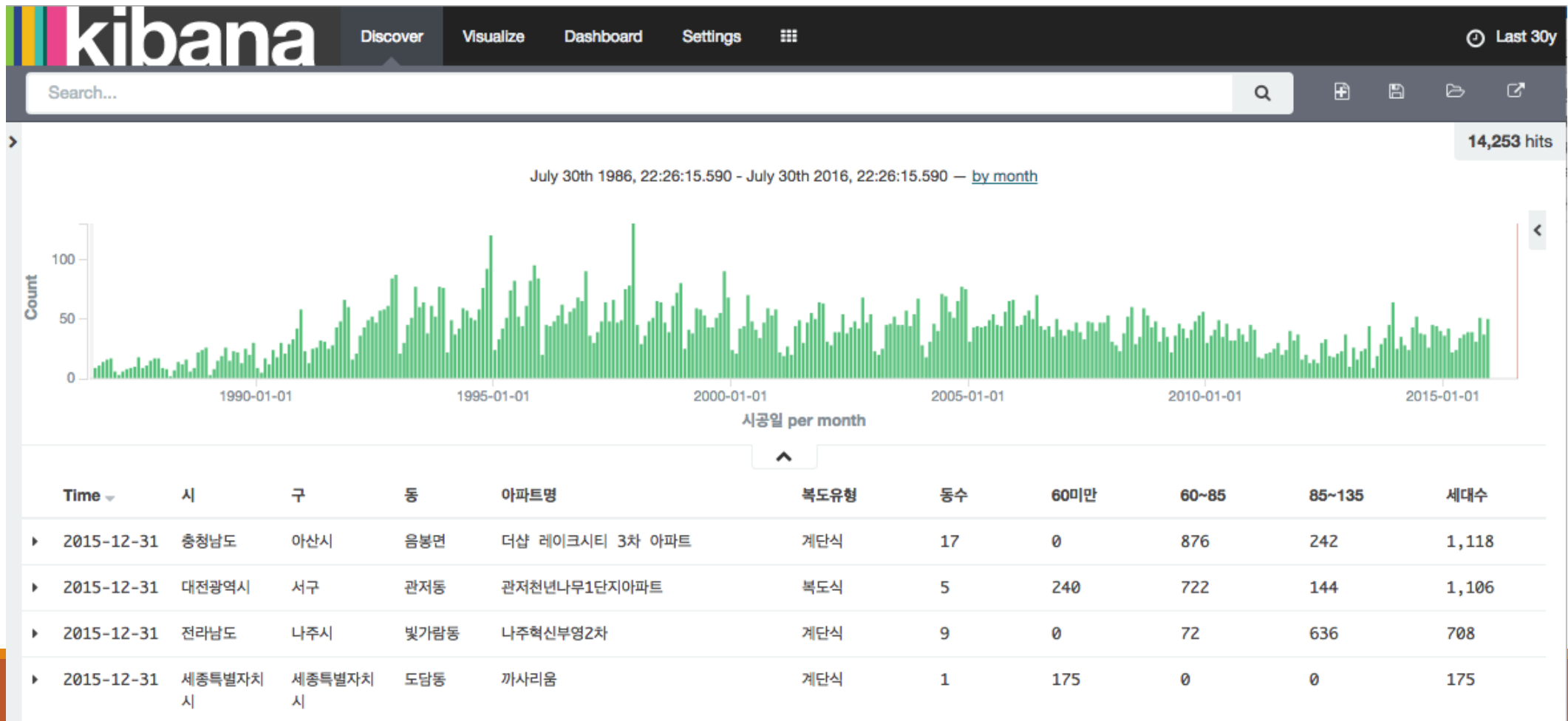
Time-field name ⓘ refresh fields

시공일

Create

kibana - discover

Discover 화면



kibana - discover

검색 방식:

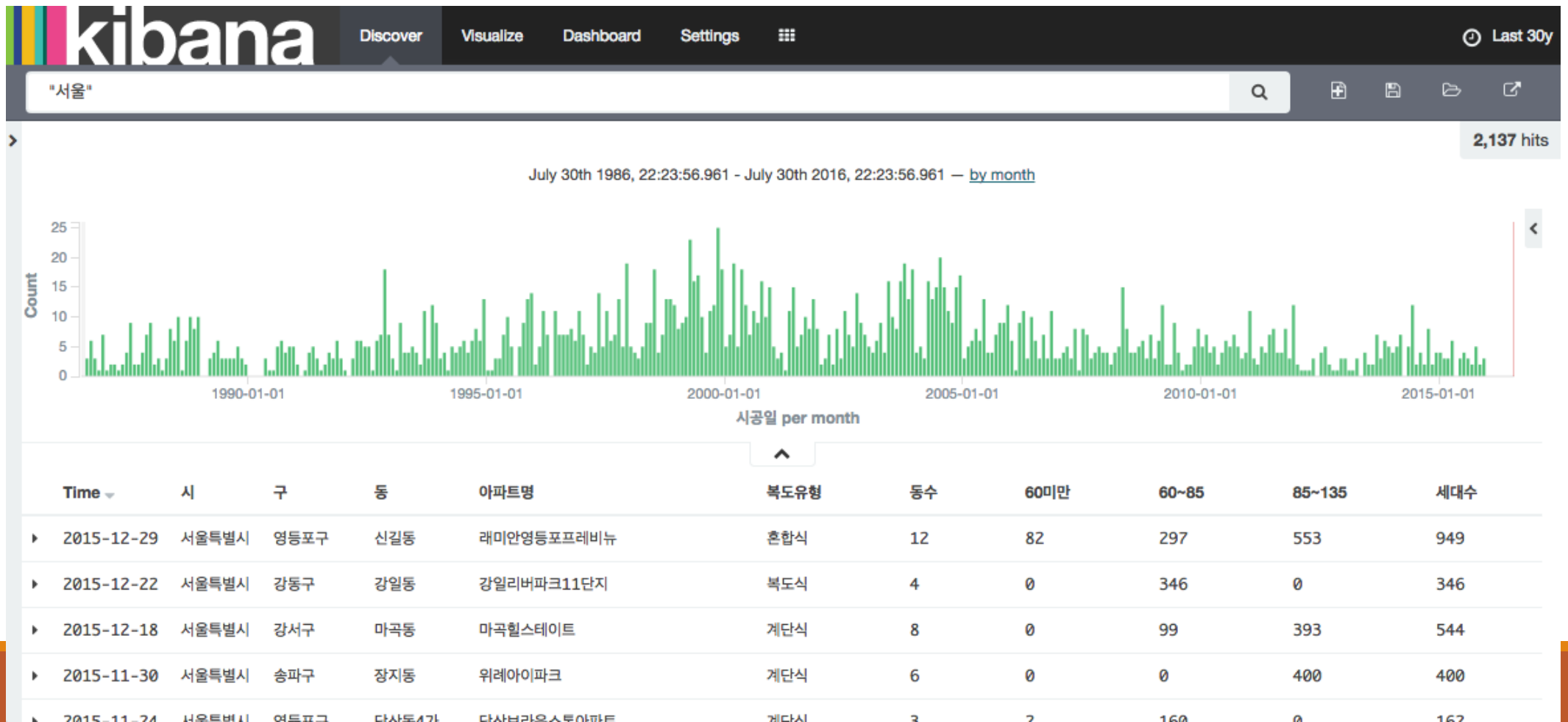
- ES의 query_string

```
{
  "query_string": {
    "query": "(content:this OR name:this) AND (content:that OR name:that)"
  }
}
```

<https://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl-query-string-query.html>

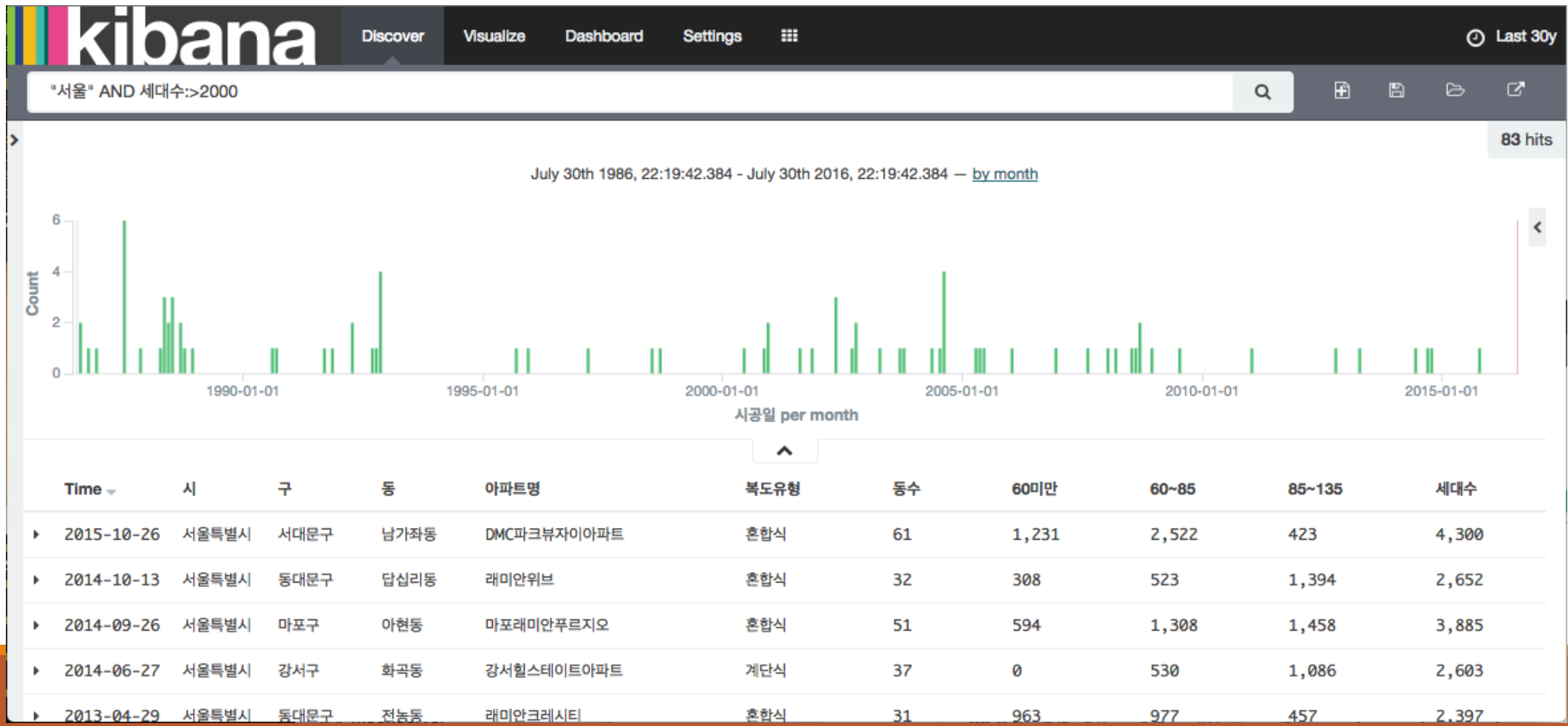
kibana - discover: 서울 지역의 아파트

서울 지역의 아파트 검색



kibana - discover: 서울 지역의 아파트

서울 지역의 세대수가 2000세대 이상



kibana - dashboard

