

1.1 Setting Up OpenSSL

Before installing OpenSSL a Java Development kit (jdk) has to be installed.

OpenSSL is downloaded from its website and the .exe is clicked to open Open SLL terminal window. After this, the command below entered to create **admin-private-key.pem** file:

```
OpenSSL> req -x509 -newkey rsa:2048 -config 'E:\OpenSSL\openssl-0.9.8e_X64\openssl.cnf' -keyout admin-private-key.pem -out admin-cert.pem -days 365 -subj "/CN=Admin Q. User/C=US/L=Seattle" -nodes
Loading 'screen' into random state - done
Generating a 2048 bit RSA private key
.....+++
.....+++
writing new private key to 'admin-private-key.pem'
```

To create **admin-cert.pem** and **admin-q-user.pfx** :

```
OpenSSL> pkcs12 -inkey admin-private-key.pem -in admin-cert.pem -export -out admin-q-user.pfx -passout pass:"SuperSecret"
Loading 'screen' into random state - done
```

To create keystore, the command below entered:

```
OpenSSL> keytool -genkeypair -alias nifiserver -keyalg RSA -keypass SuperSecret -storepass SuperSecret -keystore server_keystore.jks -dname "CN=Test NiFi Server" -noprompt
```

To add SSL certificate to the KeyStore, the below command is used:

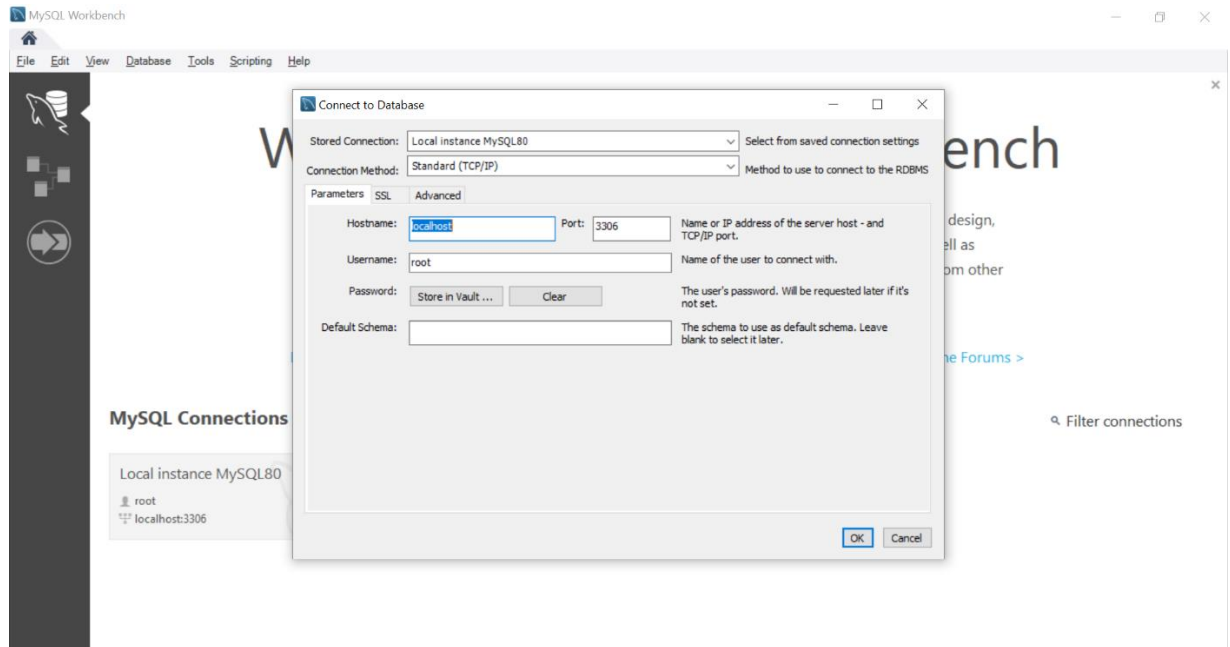
```
C:\Program Files\Java\jdk1.8.0_121\bin>
C:\Program Files\Java\jdk1.8.0_121\bin>keytool -importcert -v -trustcacerts -alias admin -file E:\OpenSSL\openssl-0.9.8e_X64\bin\admin-cert.pem -keystore server_keystore.jks -storepass SuperSecret -noprompt
Certificate was added to keystore
[Storing server_keystore.jks]
```

Normally a trust store is also needed for establishing HTTPS connection but in this project, Java Development Kit's own trust store used.

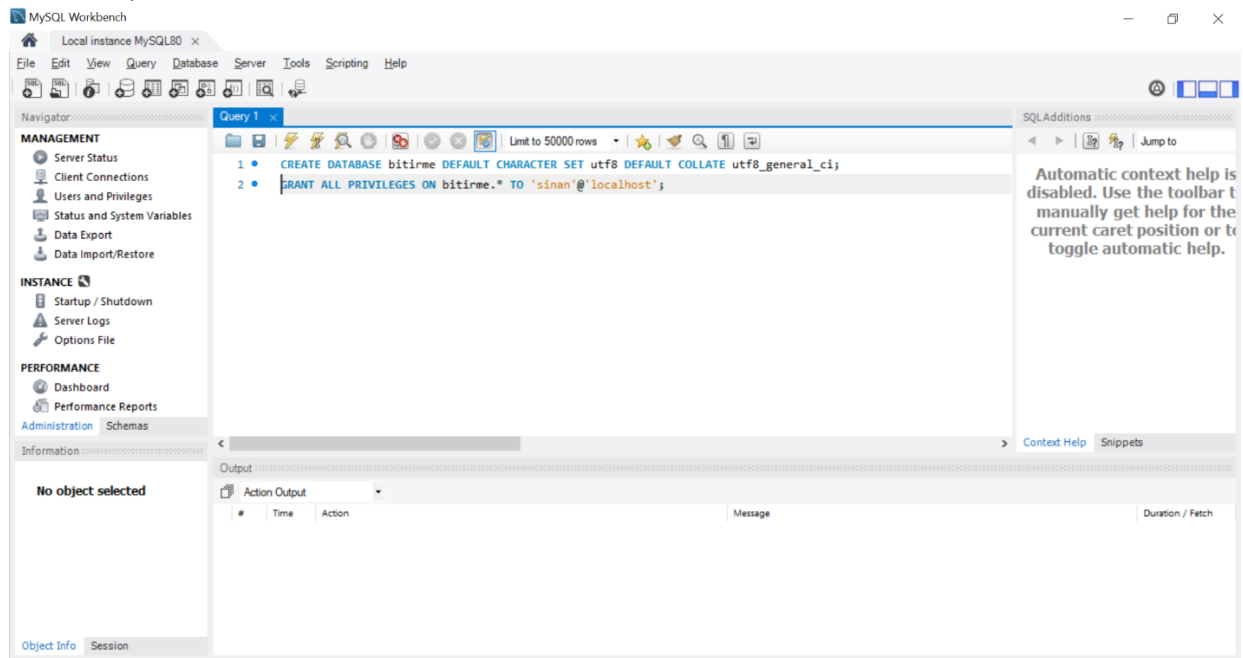
1.2 Creating MySql Database

Creating for MySQL Database, first the set up file downloaded from its official web page.

Community Edition selected for being free of charge. After downloading set-up files, .exe file clicked and a server application, client application, an ide (MySQL Workbench) installed by following directions. A root user and 'sinan' user ,which is for client connection, defined for database administration. After these steps, MySQL Workbench is opened, and a connection established with "root" user:



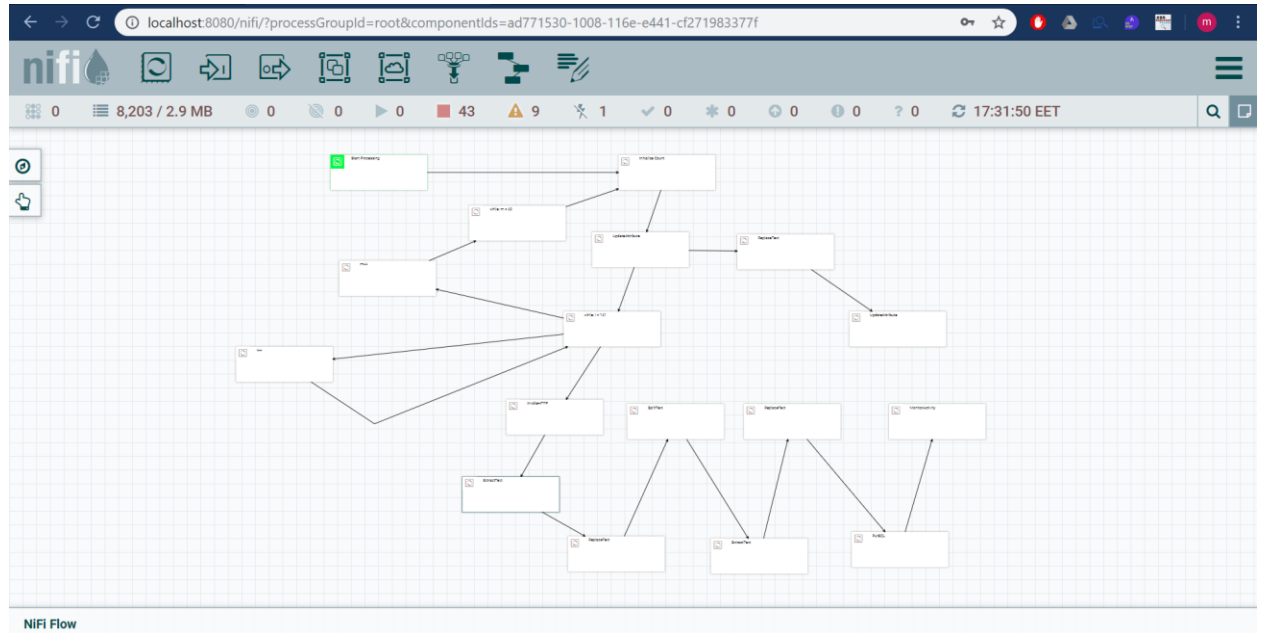
After this, a database created, which is named 'bitirme' and user 'sinan' granted for database created. Default character set defined as 'UTF-8' in order to avoid Turkish character problems:



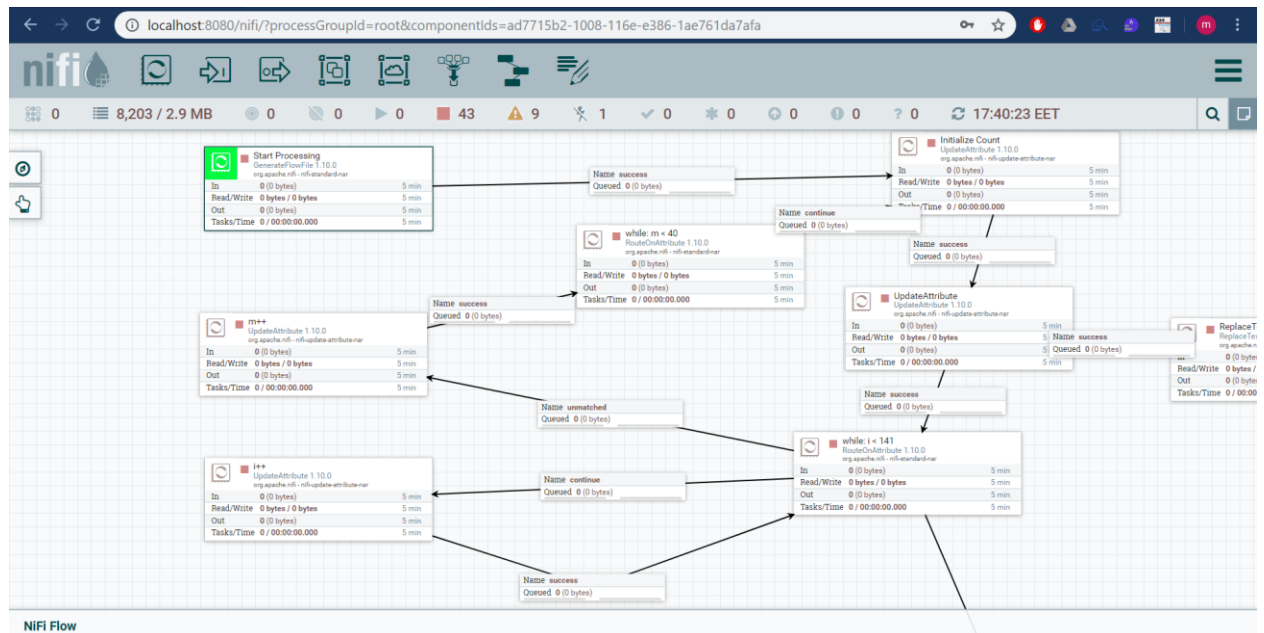
1.3 Setting up NiFi for Web Crawling

Apache NiFi is open source data integration/ETL program. It is downloaded from its official web site. Nifi does not requires any installation process for standalone use. Clicking 'run.bat' starts nifi service in a few mininutes. To open NiFi user interface, 'localhost:8080/nifi' has to

be written on any browser. All processor and their connections for data crawling is shown below:

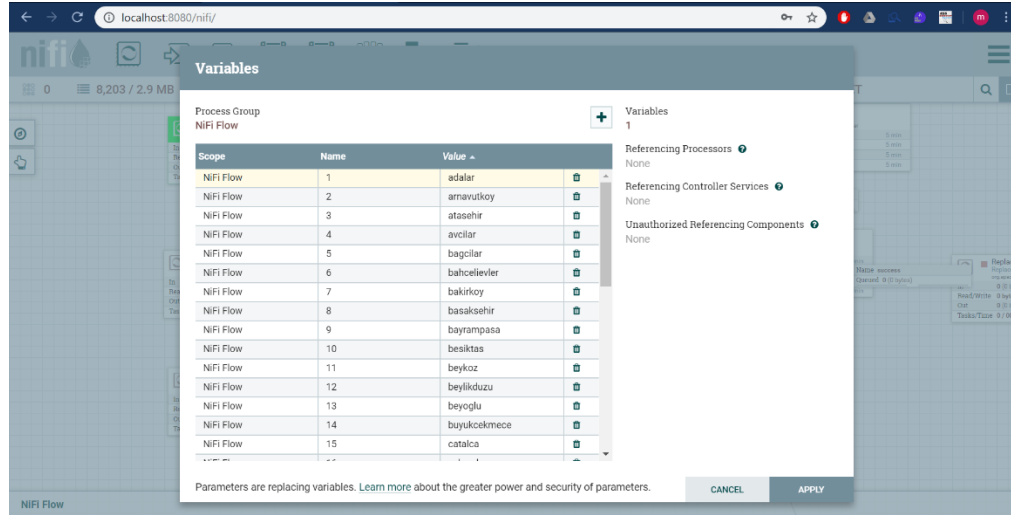


Creating dynamic web link for crawling all data for istanbul is accomplished by these processors:

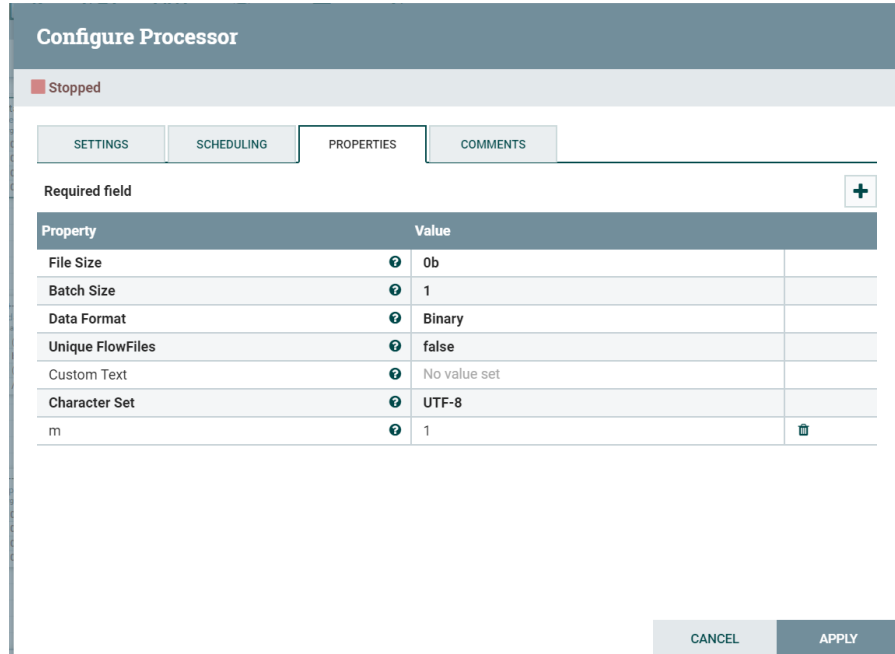


In hurriyetemlak.com, there is page limitation for every selected content as '140' (when showing 50 result per page) So if the link which contains all districts in İstanbul selected and

crawled, only 7000 real estate info could be downloaded despite having 60.000. Because of this dynamic link has to be created from district links. This is all İstanbul link: "<https://www.hurriyetemlak.com/istanbul-satilik?pageSize=50>" and this is the link for Adalar district :"<https://www.hurriyetemlak.com/adalar-satilik?pageSize=50>". To create link automatically and look for all 140 pages for all district, two loop are designed. The inner loop changes page number name and outer loop changes district name. All district names are defined in NiFi as variable first:



After these definitions, first processor added to trigger all process and defining 'm' value. 'm' will be used creating dynamic district name later.



'Initialize Count' is the processor which is used for defining 'ilce' value based on 'm' value and 'l' value which is created for dynamic page number.

Configure Processor

Stopped

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Required field

Property		Value	
Delete Attributes Expression		No value set	
Store State		Do not store state	
Stateful Variables Initial Value		No value set	
Cache Value Lookup Cache Size		100	
i		1	
ilce		\$(m)	

ADVANCED

CANCEL

APPLY

'Ilce' variable created from 'm' variable and dynamic link which includes this variable is created using 'UpdateAttribute' processor.

Configure Processor

Stopped

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Required field

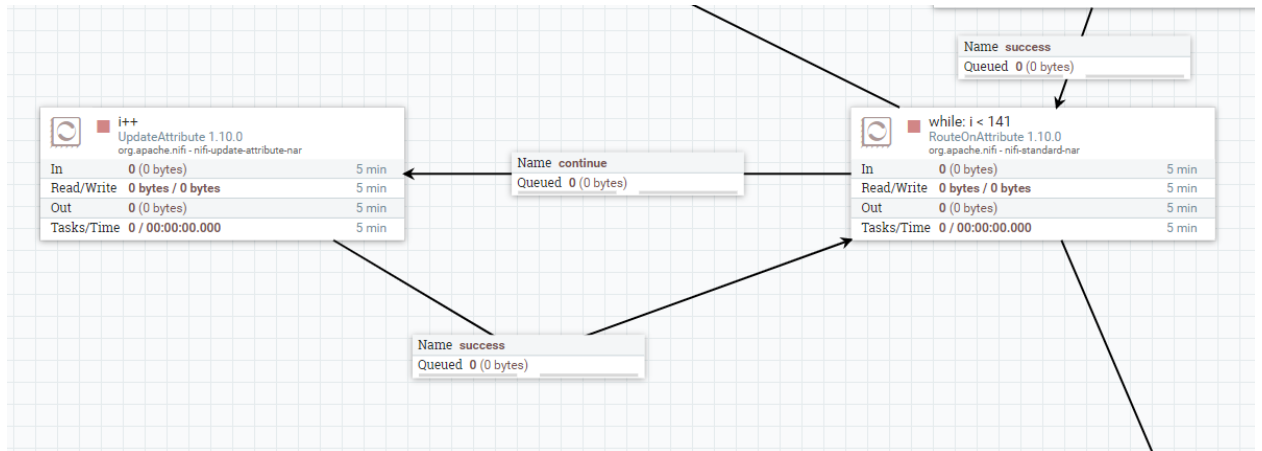
Property		Value	
Delete Attributes Expression		No value set	
Store State		Do not store state	
Stateful Variables Initial Value		No value set	
Cache Value Lookup Cache Size		100	
ilce2		\$(ilce)	

ADVANCED

CANCEL

APPLY

The next processor is 'while: i < 141' it checks the value of 'i' (page number variable) if it is smaller than 141, it routes the flow inner loop to increasing page number variable one by one. If 'i' value is greater than 140 then it routes the flow outer root to update district variables (m,ilce,ilce2). The inner loop:



The processor which is named as 'i++' increases page number variable adding 1 every time it works. 'ilce2' variable is also defined here because of 'InvokeHTTP' processor needs this flow attributes every time it works. This variable just keeps and passed district value which comes from outer loop.

Configure Processor

Stopped

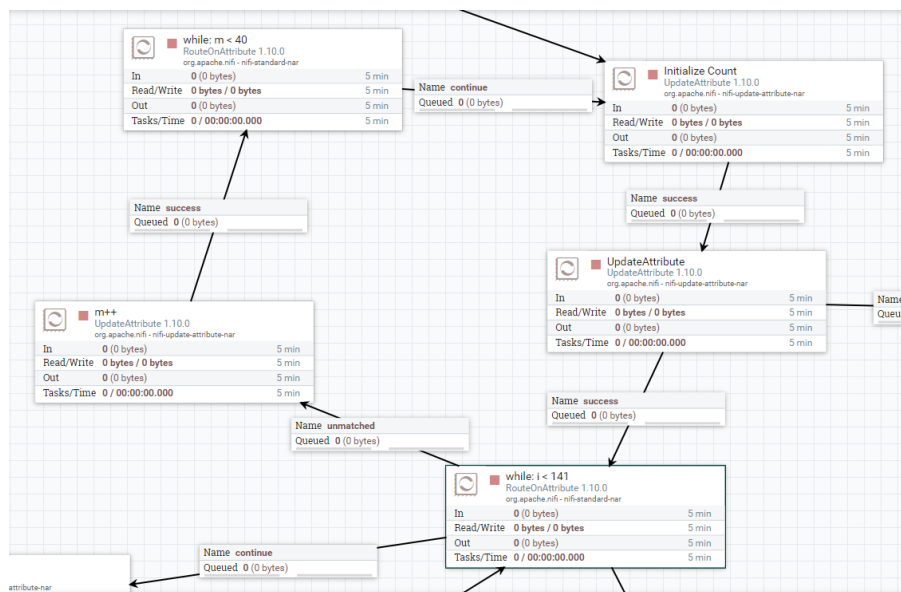
SETTINGS SCHEDULING **PROPERTIES** COMMENTS

Required field +

Property	Value
Delete Attributes Expression	No value set
Store State	Do not store state
Stateful Variables Initial Value	No value set
Cache Value Lookup Cache Size	100
i	$\${i+1}$
ilce2	$\${ilce}$

ADVANCED CANCEL APPLY

The outer loop, which is shown below, increases 'm' variable by adding one and changes district value in the flow. There are 39 districts in Istanbul so when 'm' value reaches '40' it cuts the flow and stops all data flow.



Properties of the 'm++' processor is shown below:

Configure Processor

Stopped

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Required field

Property

Value

Delete Attributes Expression	<div></div> No value set	
Store State	<div></div> Do not store state	
Stateful Variables Initial Value	<div></div> No value set	
Cache Value Lookup Cache Size	<div></div> 100	
m	<div></div> $\{m.plus(1)\}$	<div></div>

ADVANCED

CANCEL

APPLY

Properties of the ‘while: m < 40’ processor is shown below:

Configure Processor

Stopped

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Required field

Property

Value

Routing Strategy	<div></div> Route to Property name	
continue	<div></div> $\{m.lt(40)\}$	<div></div>

CANCEL

APPLY

The next processor on the flow is 'InvokeHTTP' processor. It is selected to download html file of hurriyetemlak.com. It takes flowfile content before itself and uses "get" method to download web content.

Configure Processor

Stopped

SETTINGS SCHEDULING PROPERTIES COMMENTS

Required field +

Property	Value
HTTP Method	GET
Remote URL	
SSL Context Service	StandardSSLContextService
Content	
Read	
Include	
Follow	
Attributes	
Basic	
Basic	
Proxy	
Proxy	
Proxy	

EL ✓ PARAM ✓

1 https://www.hurriyetemlak.com/\${ilce2}-satilik?pageSize=50&page=\${i}

☐ Set empty string

CANCEL OK APPLY

To use 'invokeHTTP' processor with HTTPS links, 'SSL Content Service' has to be created with OpenSSL values which created before.

Controller Service Details

SETTINGS PROPERTIES COMMENTS

Required field

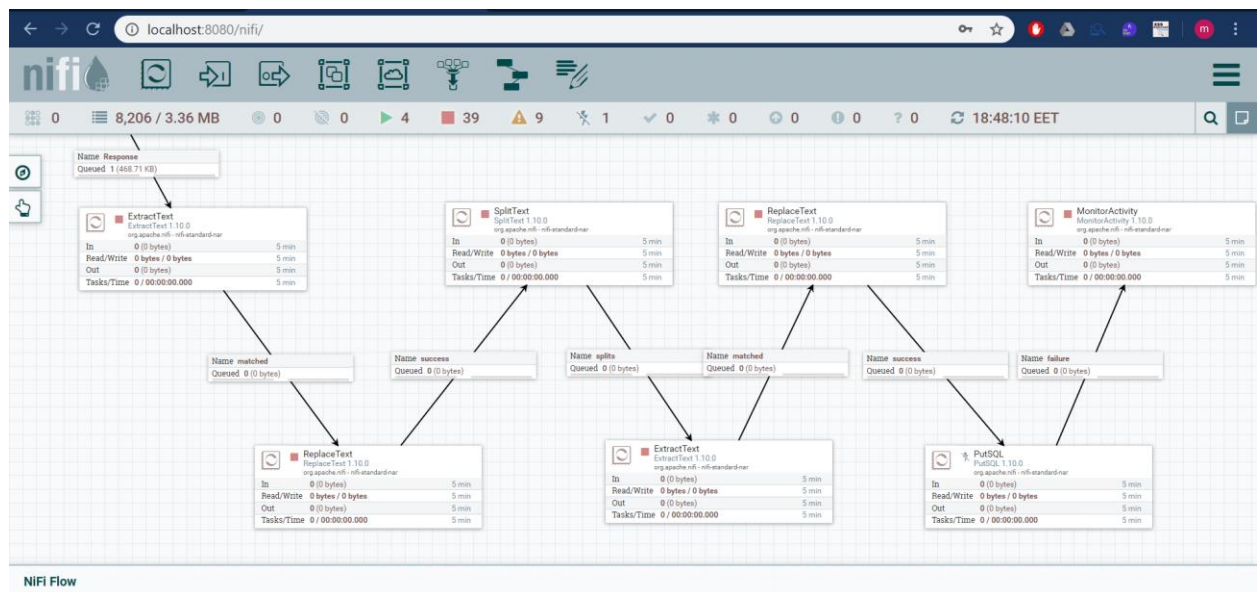
Property	Value
Keystore Filename	C:\Program Files\Java\jdk1.8.0_221\bin\server...
Keystore Password	Sensitive value set
Key Password	Sensitive value set
Keystore Type	JKS
Truststore Filename	C:\Program Files\Java\jdk1.8.0_221\jre\lib\sec...
Truststore Password	Sensitive value set
Truststore Type	JKS
TLS Protocol	SSL

OK

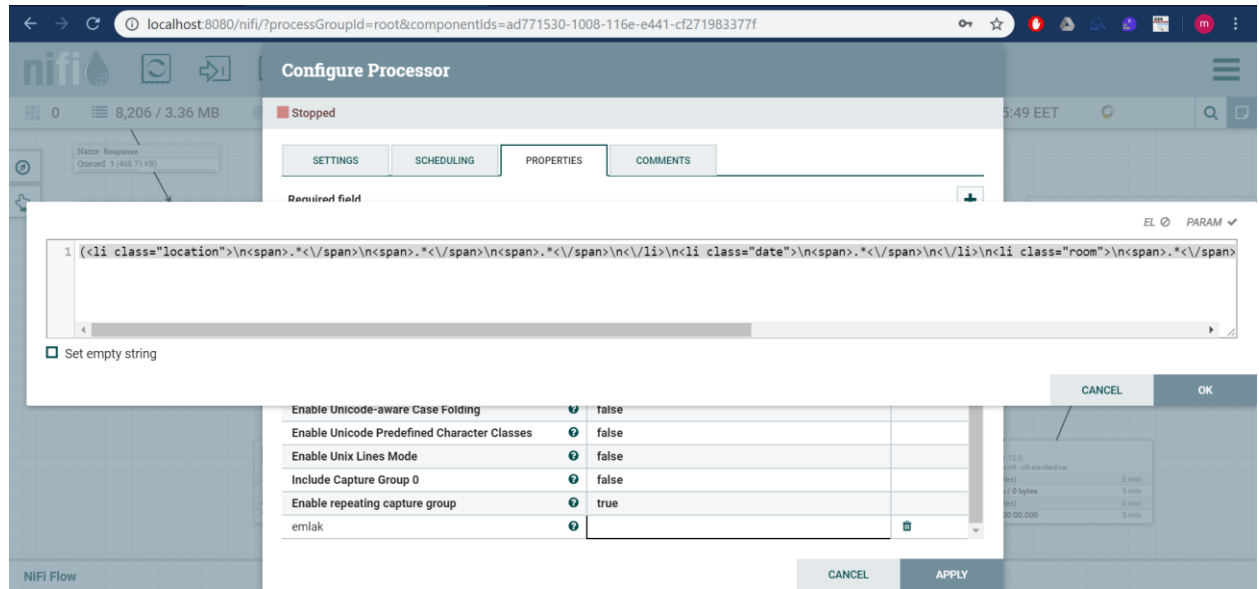
When the processors above started, a html file derived from hurriyetemlak.com as shown below.

```
1730 <div class="list-item timeshare clearfix" data-reatly-firm-id="2827" data-reatly-firm-user-id="2827" data-reatly-id="24654848">
1731 <a title="BÜYÜKADA\0027DA MUKEMMEL DENİZ MANZ.SATILIK TRİPLEKS DAİRE" class="overlay-link" href="/istanbul-adalar-buyukada-maden-satilik/daire/2827-519" data-ad-type="Standart" da
1732 
1733 <a href="javascript:void(0);" rel="nofollow" data-select-name="listing-mini-at" data-listing-id="2827-519" class="listing-mini-at isBankRide" target="_blank">Krediye Bagvur
1735 <div class="oell-1">
1736 <div class="photo">
1737 30</span>
1739 </div>
1740 <span class="tag "></span>
1741 </div>
1742 <div class="oell-2">
1743 <span class="title">
1744 B4#220;Y4#220;KADA#39;DA M4#220;KEMMEL DENİZ MANZ.SATILIK TRİPLEKS DAİRE
1745 </span>
1746 <input type="hidden" class="listEachPrice" id="" value="664000,00" />
1747 <div class="badges-wrapper ">
1748 </div>
1749 </div>
1750 <ul class="oell-3 list-features">
1751 <li class="feature">
1752 <span>İstanbul</span>
1753 <span>Adalar</span>
1754 <span>B4#220;Y4#220;KADA-Maden</span>
1755 </li>
1756 <li class="date">
1757 <span>01.12.2019</span>
1758 </li>
1759 <li class="room">
1760 <span>5+2</span>
1761 </li>
1762 <li class="square">
1763 <span>210 m²</span>
1764 </li>
1765 <li class="price">
1766 <span>830.000 TL</span>
1767 </li>
1768 </ul>
1769 </div>
1770 </div>
1771 </div>
1772
```

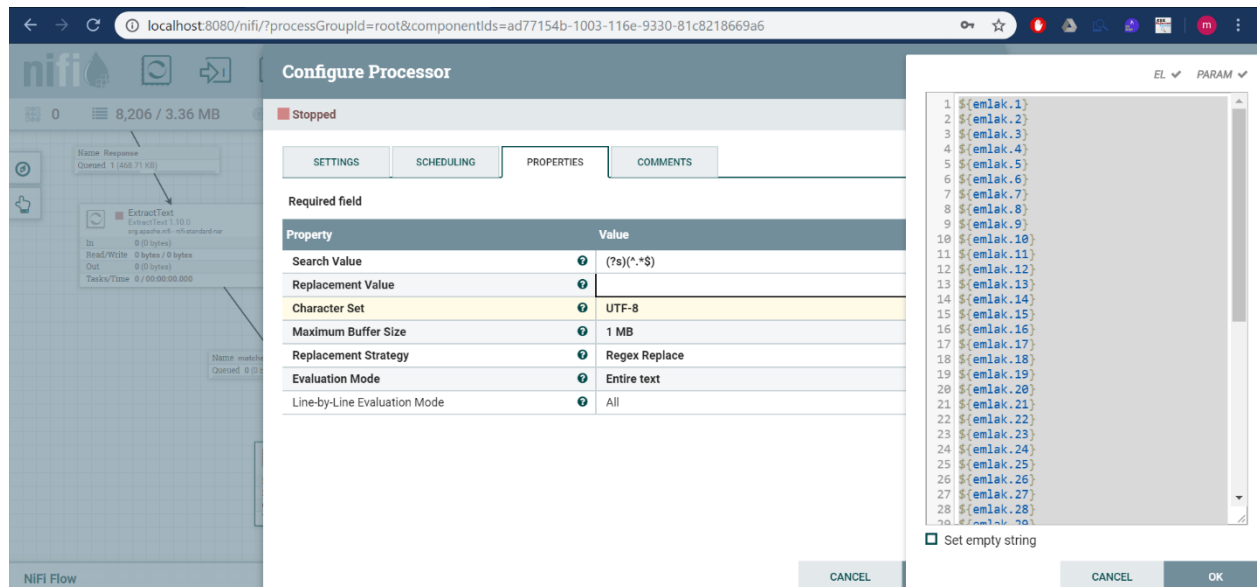
After getting this html file, data tranfomation process begins to extact real estate information from HTML file. The processor, which are doing data transformation, are shown below:



First processor in data transformation side is 'ExtractText' processor. It matches text pattern using regular expression and defines it a flowfile attribute which is manually created. For this case, manually created flowfile attribute is 'emlak'.



The next processor is 'ReplaceText' processor. It is used to replace all flow file content with 'emlak' content. All emlak content has an index which represents the capturing group of the regular expression. There are 50 real estate data on every page so there are 50 capturing groups.



After first 'ReplaceText' processor, html data tranforms this form:

```
1 <li class="location">
2 <span>Istanbul</span>
3 <span>Adalar</span>
4 <span>Bis252,y6#252;kada-Maden</span>
5 </li>
6 <li class="date">
7 <span>01.12.2019</span>
8 </li>
9 <li class="room">
10 <span>5+2</span>
11 </li>
12 <li class="square">
13 <span>210 <i>m</i></span>
14 </li>
15 <li class="price">
16 <span>830.000 TL</span>
17 </li>
18 <li class="location">
19 <span>Istanbul</span>
20 <span>Adalar</span>
21 <span>Bis252,y6#252;kada-Nizam</span>
22 </li>
23 <li class="date">
24 <span>01.12.2019</span>
25 </li>
26 <li class="room">
27 <span>3+2</span>
28 </li>
29 <li class="square">
30 <span>290 <i>m</i></span>
31 </li>
32 <li class="price">
33 <span>1.985.000 TL</span>
34 </li>
35 <li class="location">
36 <span>Istanbul</span>
37 <span>Adalar</span>
38 <span>Bis252,y6#252;kada-Maden</span>
39 </li>
40 <li class="date">
41 <span>01.12.2019</span>
42 </li>
43 <li class="room">
44 <span>2+1</span>
```

'SplitText' processor splits text by line numbers and separates code blocks to process every real estate data individually. Thanks to this processor, not only processing code block one by one decreases CPU usage, but also it will be very handy while creating insert script on next processes.

Configure Processor

Stopped

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Required field

Property		Value
Line Split Count		17
Maximum Fragment Size		No value set
Header Line Count		0
Header Line Marker Characters		No value set
Remove Trailing Newlines		true

CANCEL

APPLY

After this process, flow file data is divided 50 different code blocks. Example code block:

```
1 <li class="location">
2 <span>Istanbul</span>
3 <span>Adalar</span>
4 <span>B&#252;ry&#252;kada-Maden</span>
5 </li>
6 <li class="date">
7 <span>01.12.2019</span>
8 </li>
9 <li class="room">
10 <span>5+2</span>
11 </li>
12 <li class="square">
13 <span>210 <i>m'</i></span>
14 </li>
15 <li class="price">
16 <span>#30.000 TL</span>
17 </li>
```

The next processor is 'ExtractText' processor again. This time it is used to extract attributes from code blocks. To match patterns, regular expressions are used. The new manual flow file attribute name is 'emlak2'.

Configure Processor

Stopped

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Required field

+

Property		Value	
Enable Canonical Equivalence	?	false	
Enable Case-insensitive Matching	?	false	
Permit Whitespace and Comments in Pattern	?	false	
Enable DOTALL Mode	?	false	
Enable Literal Parsing of the Pattern	?	false	
Enable Multiline Mode	?	false	
Enable Unicode-aware Case Folding	?	false	
Enable Unicode Predefined Character Classes	?	false	
Enable Unix Lines Mode	?	false	
Include Capture Group 0	?	false	
Enable repeating capture group	?	true	
emlak2	?	(>.*<)	

CANCEL

APPLY

After that, 'ReplaceText' processor used again. Using regular expression capture groups, a sql insert script created from each code blocks. In addition to this, Turkish characters matched with Unicode values and replaced English characters. In fact, NiFi is set to 'UTF-8' format bu 'ReplaceText' processor has an bug an it spoils character set.

Configure Processor

Stopped

SETTINGS

SCHEDULING

PROPERTIES

COMMENTS

Required field

Property	Value
Search Value	(?s)(^.*\$)
Replacement Value	

EL ✓

PARAM ✓

1

INSERT INTO bitirme.hurriyetemlak_v2 (provience, district, neighborhood, publish_date, r

☐ Set empty string

CANCEL

OK

CANCEL

APPLY

Full code of replacement string:

```

1  INSERT INTO bitirme.hurriyetemlak_v2 (provience, district, neighborhood, publish_date, room, square, price)
2  VALUES (${emlak2.1:replaceAll("<|>",""):replace("#304","I"):replace("#286","G"):replace("#350","S"):
3  replace("#199","C"):replace("#220","U"):replace("#214","O"):replace("#305","i"):replace("#287","g"):
4  replace("#351","s"):replace("#231","c"):replace("#252","u"):replace("#246","o")},"${emlak2.2:
5  replaceAll("<|>",""):replace("#304","I"):replace("#286","G"):replace("#350","S"):replace("#199","C"):
6  replace("#220","U"):replace("#214","O"):replace("#305","i"):replace("#287","g"):replace("#351","s"):
7  replace("#231","c"):replace("#252","u"):replace("#246","o")}","${emlak2.3:replaceAll("<|>",""):
8  replace("#304","I"):replace("#286","G"):replace("#350","S"):replace("#199","C"):replace("#220","U"):
9  replace("#214","O"):replace("#305","i"):replace("#287","g"):replace("#351","s"):replace("#231","c"):
10 replace("#252","u"):replace("#246","o")}","${emlak2.4:replaceAll("<|>",""):}${emlak2.5:replaceAll("<|>",""):
11 replace("#304","I"):replace("#286","G"):replace("#350","S"):replace("#199","C"):replace("#220","U"):
12 replace("#214","O"):replace("#305","i"):replace("#287","g"):replace("#351","s"):replace("#231","c"):
13 replace("#252","u"):replace("#246","o")}","${emlak2.6:replaceAll("<|>",""):replace(" im²/i","")},"${emlak2.7:
14 replaceAll("<|>",""):replace(" TL",""):replace(".",",")}"]

```

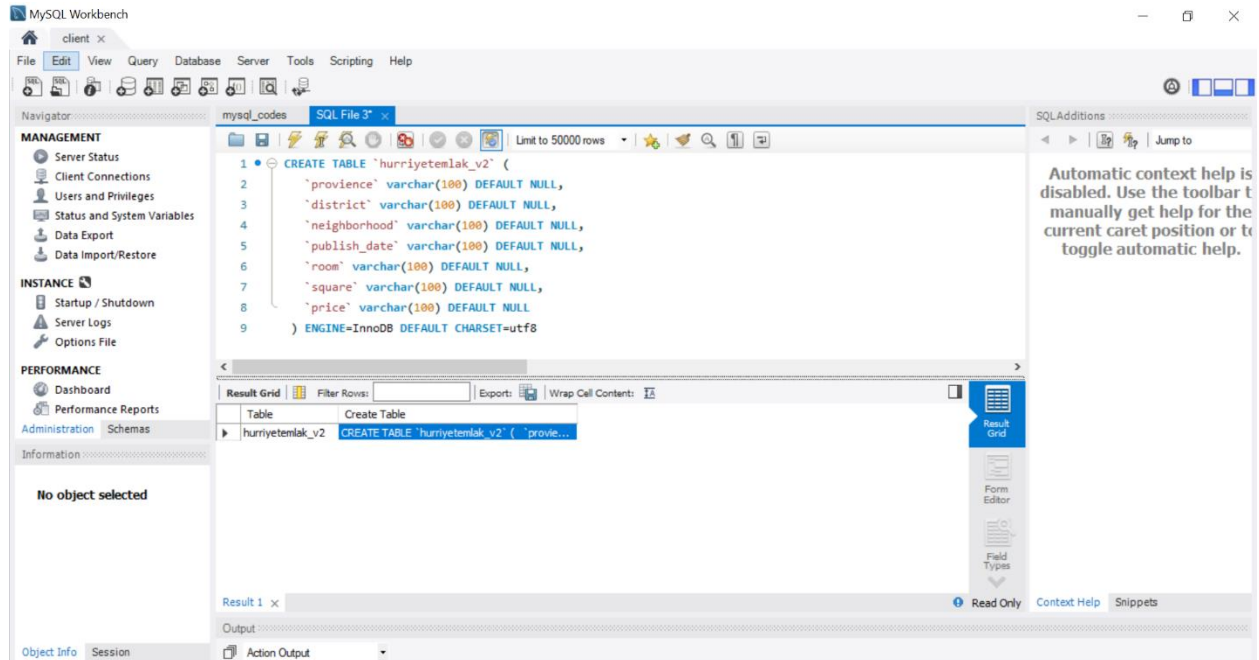
After this, code block successfully transformed to insert scripts.

```

1  INSERT INTO bitirme.hurriyetemlak_v2 (provience, district, neighborhood, publish_date, room, square, price)
2  VALUES ("İstanbul","Adalar","Buyukada-Maden","01.12.2019","5+2","210","830000")

```

Before sending insert script to MySQL database, a table which satisfies this attributes created on 'bitirme' database with 'sinan' user.



After creating 'hurriyetemlak_v2' table, last processor of the data flow, 'PutSQL' processor sends sql insert scripts to MySQL database. The processor setting is:

Configure Processor

✖ Disabled

SETTINGS SCHEDULING PROPERTIES COMMENTS

Required field +

Property	Value
JDBC Connection Pool	MySQLDBCPCConnection →
SQL Statement	No value set
Support Fragmented Transactions	false
Database Session AutoCommit	false
Transaction Timeout	No value set
Batch Size	100
Obtain Generated Keys	false
Rollback On Failure	false

CANCEL APPLY

To get processor work, connection properties of MySQL database has to be defined in 'JDBC connection pool' controller service.

Controller Service Details

SETTINGSPROPERTIESCOMMENTS

Required field

Property	Value
Database Connection URL	? jdbc:mysql://localhost:3306/bitirme
Database Driver Class Name	? com.mysql.jdbc.Driver
Database Driver Location(s)	? C:\Users\Sinan\Desktop\bitirme\mysql-conne...
Kerberos Credentials Service	? No value set
Database User	? root
Password	? Sensitive value set
Max Wait Time	? 500 millis
Max Total Connections	? 8
Validation query	? No value set
Minimum Idle Connections	? 0
Max Idle Connections	? 8
Max Connection Lifetime	? -1
Time Between Eviction Runs	? -1
Minimum Evictable Idle Time	? 30 mins

OK

The design explained above has worked for 12,5 hours. It has download 60202 records from Hürriyetemlak.com.

MySQL Workbench

client x

File Edit View Query Database Server Tools Scripting Help

mysql_codes SQL File 3'

Limit to 50000 rows

1 • select * from hurriyetemlak_v2

Result Grid

Filter Rows:

Exports: Wrap Cell Contents Fetch rows:

province	district	neighborhood	publish_date	room	square	price
Istanbul	Adalar	Buyukada-Nizam	23.11.2019	3+1	90	535000
Istanbul	Adalar	Buyukada-Maden	24.11.2019	9 ve üzeri	302	9400000
Istanbul	Adalar	Buyukada-Maden	23.11.2019	3+1	120	600000
Istanbul	Adalar	Buyukada-Maden	23.11.2019	4+1	133	660000
Istanbul	Adalar	Buyukada-Maden	22.11.2019	3+1	130	1100000
Istanbul	Adalar	Burgazadası	23.11.2019	3+1	150	2700000
Istanbul	Adalar	Kınalada	23.11.2019	2+1	105	590000
Istanbul	Adalar	Buyukada-Maden	23.11.2019	5+2	210	825000
Istanbul	Adalar	Buyukada-Maden	18.11.2019	3+1	150	1730000
Istanbul	Adalar	Buyukada-Maden	22.11.2019	9 ve üzeri	901	5400000
Istanbul	Adalar	Buyukada-Nizam	23.11.2019	3+1	140	2400000
Istanbul	Adalar	Buyukada-Nizam	23.10.2019	4+1	150	800000
Istanbul	Adalar	Buyukada-Nizam	23.11.2019	3+1	160	3500000
Istanbul	Adalar	Burgazadası	22.11.2019	6+2	280	18000000
Istanbul	Adalar	Burgazadası	22.11.2019	7+2	450	2650000
Istanbul	Adalar	Buyukada-Nizam	20.11.2019	3+1	110	825000

hurriyetemlak_v2.3 x

Output

Read Only Context Help Snippets

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

After this process, another table created from hurriyetemlak_v2 to prepare data analyses which would have been made in Pycharm.

```
1 create table hurriyetemlak_v4 as
2 select lower(province) as province, lower(district) as district, lower(neighborhood) as neighborhood,
3 convert(price, signed int) as price, convert(square, signed int) as square,
4 convert(case
5 when room = '3+1' then 4
6 when room = '9 ve üzeri' then 10
7 when room = '4+1' then 5
8 when room = '2+1' then 3
9 when room = '5+2' then 7
10 when room = '6+2' then 8
11 when room = '7+2' then 9
12 when room = '5+1' then 6
13 when room = '6+1' then 7
14 when room = '7+4 ve üzeri' then 12
15 when room = '3+2' then 5
16 when room = '4+2' then 6
17 when room = '8+2' then 10
18 when room = '8+3' then 11
19 when room = '5+3' then 8
20 when room = '7+1' then 8
21 when room = '7+3' then 10
22 when room = '1+1' then 2
23 when room = '6+3' then 9
24 when room = '8+1' then 9
25 when room = '8+4 ve üzeri' then 13
26 when room = '6+4 ve üzeri' then 11
27 when room = '2+2 ve üzeri' then 4
28 when room = '4+4 ve üzeri' then 8
29 when room = '4+3' then 7
30 when room = '0' then 0
31 when room = '5+4 ve üzeri' then 10
32 when room = '2' then 2
33 when room = '3+3 ve üzeri' then 6
34 when room = '1+0' then 1
35 when room = '3' then 3
36 when room = '4' then 4
37 when room = '5' then 5
38 when room = '7' then 7
39 when room = '6' then 6
40 when room = '8' then 8
41 else room
42 end, signed int) as room
43 from hurriyetemlak_v2;
```