

데이터베이스 및 데이터시각화

Project 3 - Tutorial

Kwangwoon Univ.
Dept. of Computer Engineering
Ki-Hoon Lee



Contents

1. 데이터 시각화
2. 데이터베이스 연동



데이터 시각화

- 데이터 시각화
 - 데이터 분석 결과를 쉽게 이해할 수 있도록 시각적으로 표현하고 전달되는 과정으로 차트라는 수단을 통해 정보를 전달
- 데이터 시각화 툴
 - D3.js
 - Google Chart



데이터 시각화

- 1차 튜토리얼에서 사용했던 C:\WmyNode\joinForm 사용
- routes 폴더에 visualization.js 파일 생성

```
visualization.js x
1  var express = require('express');
2  var router = express.Router();
3
4  /* GET users listing. */
5  router.get('/', function(req, res, next) {
6    res.send('visualization');
7  });
8
9  module.exports = router;
```



데이터 시각화

- app.js에 소스를 추가하여 visualization.js 파일 연결

```
visualization.js x app.js x
1 var createError = require('http-errors');
2 var express = require('express');
3 var path = require('path');
4 var cookieParser = require('cookie-parser');
5 var logger = require('morgan');
6
7 var indexRouter = require('./routes/index');
8 var usersRouter = require('./routes/users');
9 var join = require('./routes/joinForm');
10 var board = require('./routes/board');
11 var visual = require('./routes/visualization');
25 app.use('/', indexRouter);
26 app.use('/users', usersRouter);
27 app.use('/join', join);
28 app.use('/board', board);
29 app.use('/visual', visual);
```



데이터 시각화

- 다음 소스코드를 따라서 작성하여 views 폴더에 visualization.ejs 템플릿 생성

```
1 <!DOCTYPE html>
2 <html>
3   <script src="https://www.gstatic.com/charts/loader.js"></script>
4   <head>
5     <script src="/javascripts/jquery-3.6.0.min.js"></script>
6     <title><%= title %></title>
7     <link rel='stylesheet' href='/stylesheets/style.css' />
8   </head>
9   <body>
10    <div id="myChart" style="max-width:700px; height:400px"></div>
11    <script>
12      google.charts.load('current', {'packages':['corechart']});
13      google.charts.setOnLoadCallback(drawChart);
14
15      function drawChart() {
16        var data = google.visualization.arrayToDataTable([
17          ['Contry', 'Mhl'],
18          ['Italy',54.8],
19          ['France',48.6],
20          ['Spain',44.4],
21          ['USA',23.9],
22          ['Argentina',14.5]
23        ]);
24
25        var options = {
26          title:'World Wide Wine Production'
27        };
28
29        var chart = new google.visualization.PieChart(document.getElementById('myChart'));
30        chart.draw(data, options);
31      }
32    </script>
33  </body>
34 </html>
```

Google Chart API 링크

Chart의 layout

Chart의 데이터

Chart의 옵션

Chart의 종류



데이터 시각화

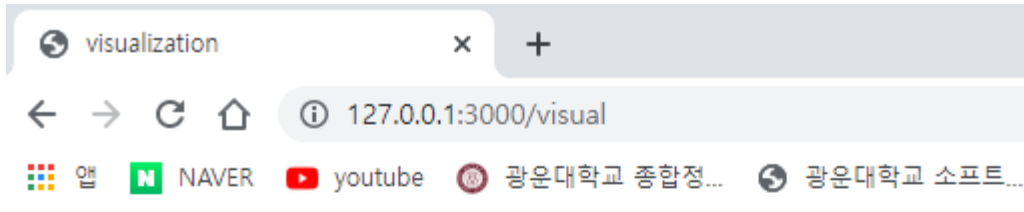
- Routes폴더의 visualization.js의 res.send 부분 수정

```
visualization.js  x  app.js  x  visualization.ejs  x
1  var express = require('express');
2  var router = express.Router();
3
4  /* GET users listing. */
5  router.get('/', function(req, res, next) {
6    res.render('visualization', {title: 'visualization'});
7  });
8
9  module.exports = router;
```



데이터 시각화

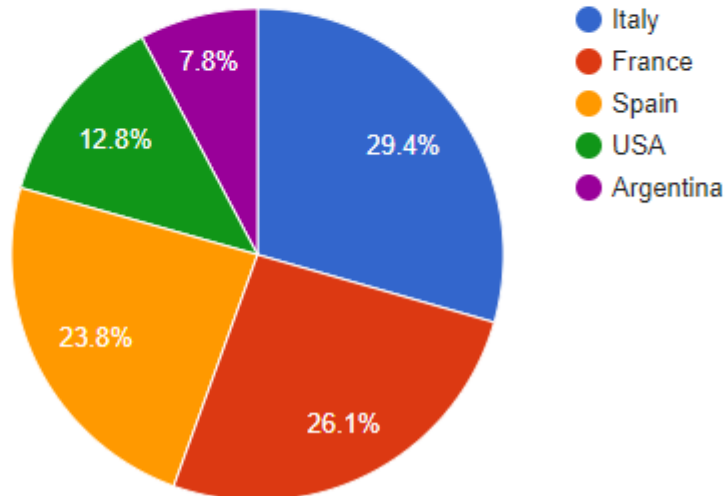
■ 웹 브라우저를 통해 결과 확인



```
function drawChart() {
  var data = google.visualization.arrayToDataTable([
    ['Contry', 'Mhl'],
    ['Italy', 54.8],
    ['France', 48.6],
    ['Spain', 44.4],
    ['USA', 23.9],
    ['Argentina', 14.5]
  ]);

  var options = {
    title: 'World Wide Wine Production'
  };
}
```

World Wide Wine Production

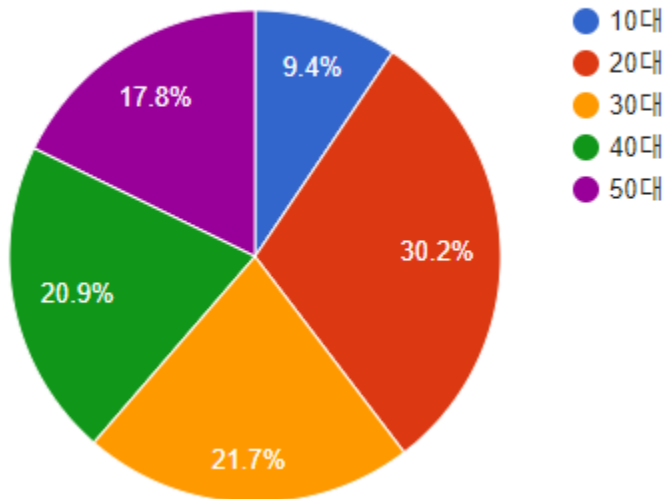




데이터 시각화

- 데이터를 수정하고 결과 확인

연령대별 접종 현황



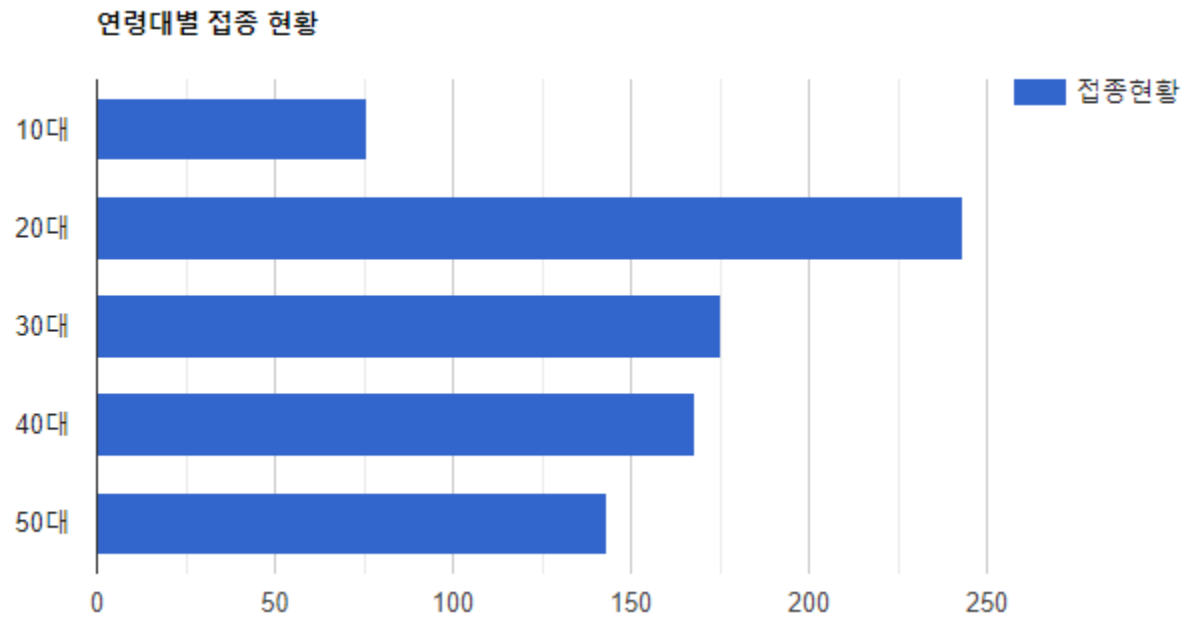
```
function drawChart() {  
  var data = google.visualization.arrayToDataTable([  
    ['연령대', '접종현황'],  
    ['10대', 76],  
    ['20대', 243],  
    ['30대', 175],  
    ['40대', 168],  
    ['50대', 143]  
  ]);  
  
  var options = {  
    title: '연령대별 접종 현황'  
  };  
}
```



데이터 시각화

- 차트 종류를 바꾸고 결과 확인

```
//var chart = new google.visualization.PieChart(document.getElementById('myChart'));  
var chart = new google.visualization.BarChart(document.getElementById('myChart'));
```

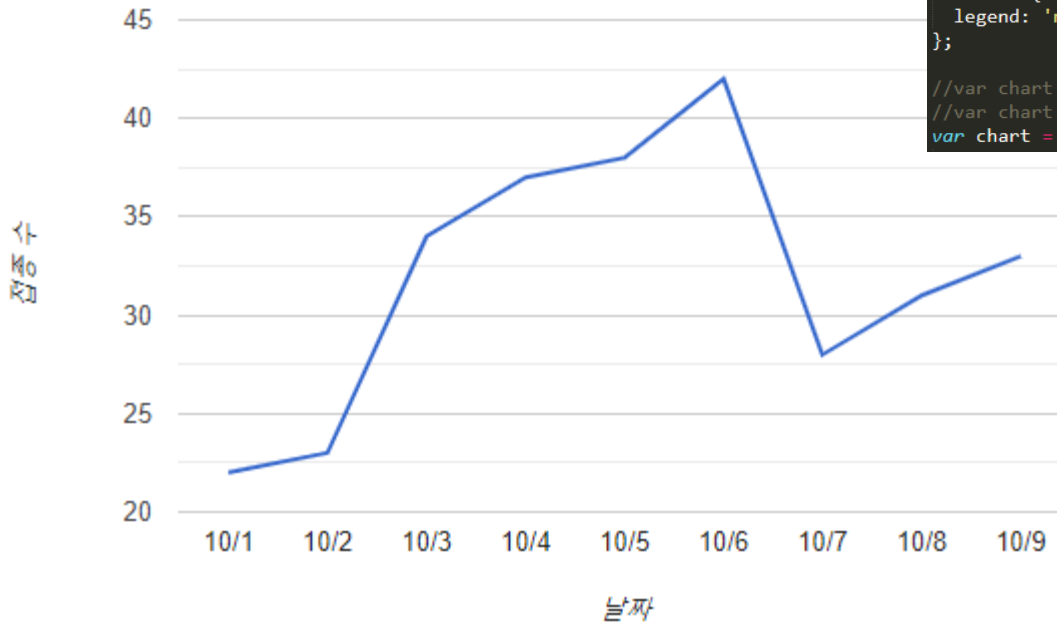




데이터 시각화

■ 차트 종류를 바꾸고 결과 확인

날짜별 접종 현황



```
function drawChart() {  
  var data = google.visualization.arrayToDataTable([  
    ['날짜', '접종현황'],  
    ['10/1', 22],  
    ['10/2', 23],  
    ['10/3', 34],  
    ['10/4', 37],  
    ['10/5', 38],  
    ['10/6', 42],  
    ['10/7', 28],  
    ['10/8', 31],  
    ['10/9', 33]  
  ]);  
  
  var options = {  
    title: '날짜별 접종 현황',  
    hAxis: {title: '날짜'},  
    vAxis: {title: '접종 수'},  
    legend: 'none'  
  };  
  
  //var chart = new google.visualization.PieChart(document.getElementById('myChart'));  
  //var chart = new google.visualization.BarChart(document.getElementById('myChart'));  
  var chart = new google.visualization.LineChart(document.getElementById('myChart'));
```



데이터베이스 연동

- 데이터베이스에 테이블 생성 후 데이터 입력

```
use tutorial;  
create table stat(  
  ID int unsigned not null primary key auto_increment,  
  age int not null,  
  vacc bool default false  
);
```

10 • select * from stat;

Result Grid |  Filter Rows:

	ID	age	vacc
1	15	1	
2	23	1	
3	24	0	
4	25	1	
5	37	0	
6	48	1	
7	53	0	
8	36	1	
9	19	1	
10	23	1	
11	25	1	
12	48	1	
13	53	0	
14	36	1	
15	19	1	
16	54	1	
	NULL	NULL	NULL



데이터베이스 연동

■ routes와 visualization.js 수정

```
visualization.js  x  app.js  x  visualization.ejs  x  list.ejs

var express = require('express');
var router = express.Router();
var mysql = require('mysql');
var pool = mysql.createPool({
  connectionLimit: 5,
  host: 'localhost',
  user: 'root',
  password: '1234',
  database: 'tutorial'
});

/* GET users listing. */
router.get('/', function(req, res, next) {
  pool.getConnection(function(err, connection){
    var sql = "SELECT case when age < 20 then '10대' when age < 30 then '20대'
    connection.query(sql, function(err, rows){
      if (err) console.error("err: " + err);
      console.log("rows: " + JSON.stringify(rows));

      res.render('visualization', {title: 'visualization', rows: rows});
      connection.release();
    })
  })
});

module.exports = router;
```

```
select case
  when age < 20 then '10대'
  when age < 30 then '20대'
  when age < 40 then '30대'
  when age < 50 then '40대'
  when age < 60 then '50대'
end
as age_group, count(*) as cnt
from stat
where vacc=1
group by age_group
order by age_group;
```



데이터베이스 연동

■ viewWvisualization.ejs 수정

```
function drawChart() {  
  var data = google.visualization.arrayToDataTable([  
    ['age_group', 'cnt'],  
    <%  
    for(var i=0; i<rows.length; i++){  
      var oneItem = rows[i];  
    %>  
  
    ["<%=oneItem.age_group%", parseInt("<%=oneItem.cnt%")"],  
  
    <%  
    }  
    %>  
  ]);  
  
  var options = {  
    title: "연령별 접종 현황",  
    hAxis: {title: '연령대'},  
    vAxis: {title: '접종 수', viewWindow: {min:0, max:5}},  
    legend: 'none'  
  };  
};
```



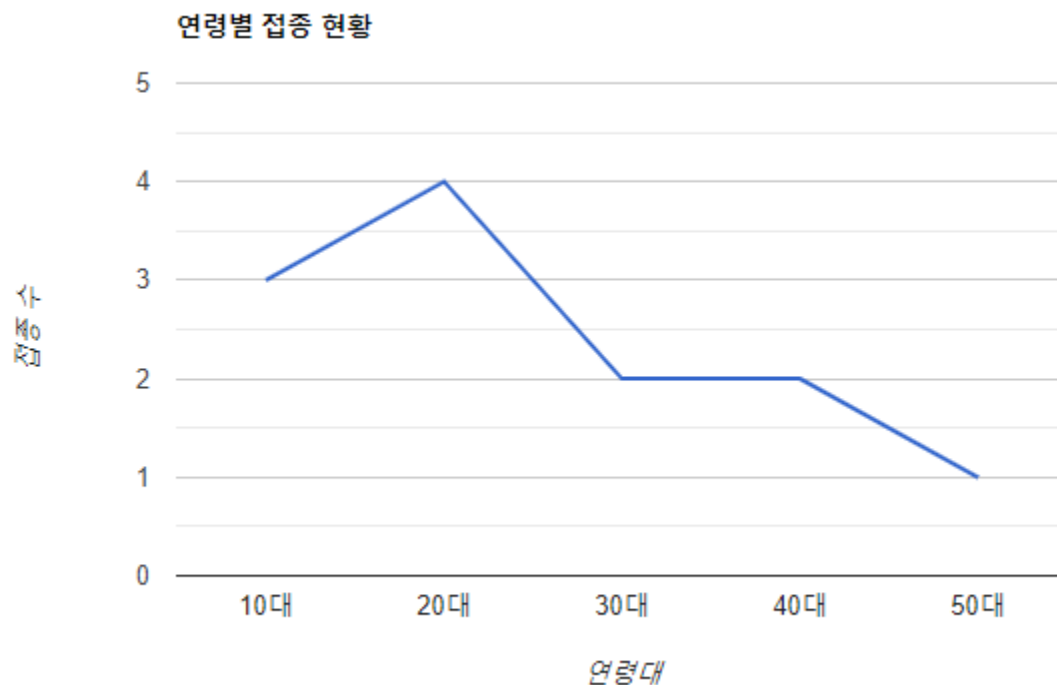
데이터베이스 연동

■ 결과 확인

```
12 • select case
13     when age < 20 then '10대'
14     when age < 30 then '20대'
15     when age < 40 then '30대'
16     when age < 50 then '40대'
17     when age < 60 then '50대'
18     end
19   as age_group, count(*) as cnt
20   from stat
21  where vacc=1
22  group by age_group
23  order by age_group;
```

Result Grid | Filter Rows:

age_group	cnt
10대	3
20대	4
30대	2
40대	2
50대	1





관련 사이트

- Google Chart API
 - <https://developers.google.com/chart/interactive/docs>
 - https://www.w3schools.com/ai/ai_google_chart.asp