



MySQL Node.js 와의 연동

Web 개발 기초





1. MySQL

2. phpmyadmin

3. DB 연동





1. MySQL

DB?

이제 HTML에 들어갈 데이터를 가져와야 한다.

>>> 어디서? 어떻게?

바로 DB(Database)에서 가져온다.

DB : 데이터들을 저장 및 관리하는 소프트웨어

우리는 MySQL을 DB로 쓴다.

그리고 phpmyadmin으로 MySQL을 쉽게 관리할 수 있다

Node.js를 이용해 DB와 연결을 한다.

1. MySQL

어떻게 작동하는가?



User_idx = 1

{Title,content,
date,author}



```
<?foreach($portfolios as $portfolio):?>
  <div class="service_<?=$portfolio->service_idx?> col-xs-12 co
    <div class="portfolio_item">
      <a href="/portfolio/item/<?=$portfolio->idx?>" data-f
        <figure style="background-image:url(<?=$portfolio->
          <svg viewBox="0 0 180 320" preserveAspectRatio
            <path d="M 180,0 0,0 0,0 180,0 z"/>
          </svg>
          <figcaption>
            <p><?=$portfolio->description?></p>
            <div class="view_button">View</div>
          </figcaption>
        </figure>
      </a>
      <div class="portfolio_description">
        <h3><a class="logcolored" href="/portfolio/ite
          <p>travel</p> -->
        </div>
      </div>
    </div>
  <?endforeach;?>
</div>
>
```

1. MySQL

DB



DB에서 할 수 있는 동작은 크게 4가지.

>>> SELECT, INSERT, UPDATE, DELETE

- 예1) html에 보여주기 전에 DB에 **SELECT**를 해서 데이터를 가져온다.
- 예2) html에서 사용자가 입력한 데이터는 **INSERT**를 통해 저장한다
- 예3) 사용자가 프로필을 변경하거나 하면 **update**를 통해 변경한다
- 예4) 사용자가 탈퇴하면, **delete**를 통해 해당 데이터를 모두 삭제한다.



1. MySQL

DB

DB의 구성은 Excel 과 비슷하다.

DB	EXCEL
Database	엑셀 한 파일
Table	엑셀 파일의 sheet
Row	엑셀의 한 row
Column	엑셀의 한 column
Data value	셀의 value

1. MySQL

DB

Table

테이블 ▲	실행	행 ⓘ	종류	데이터정렬방식	크기	부담
<input type="checkbox"/> Gift	★ 보기 구조 검색 삽입 비우기 삭제	0	InnoDB	utf8_general_ci	16 KB	-
<input type="checkbox"/> Homework	★ 보기 구조 검색 삽입 비우기 삭제	7	InnoDB	utf8_general_ci	16 KB	-
<input type="checkbox"/> note	★ 보기 구조 검색 삽입 비우기 삭제	6	InnoDB	utf8_general_ci	16 KB	-
<input type="checkbox"/> sessions	★ 보기 구조 검색 삽입 비우기 삭제	52	InnoDB	utf8_general_ci	16 KB	-
<input type="checkbox"/> Stampcoin	★ 보기 구조 검색 삽입 비우기 삭제	12	InnoDB	utf8_general_ci	16 KB	-
<input type="checkbox"/> TEST	★ 보기 구조 검색 삽입 비우기 삭제	3	InnoDB	utf8_general_ci	16 KB	-
6개 테이블 계		80	InnoDB	utf8_general_ci	96 KB	0 B



1. MySQL

DB

Row

		note_idx	note_id	content_text	note_date	isDisabled
<input type="checkbox"/> 수정 복사 삭제		150	joonew	연 : 1~10번 수 : 16~30번 외 : p23~28	2017-03-21 23:36:29	0
<input type="checkbox"/> 수정 복사 삭제		154	joonew	연 : 11~24번 수 : 1~15번 외 : p29~35	2017-03-22 23:53:35	0
<input type="checkbox"/> 수정 복사 삭제		155	joonew	연 : 25~38번 수 : 16~30번 외 : p36~41	2017-03-23 23:58:38	0
<input type="checkbox"/> 수정 복사 삭제		156	joonew	오답노트 다시 보기 지1:(1)단원 생1:(1)단원 (아직 책을 안사서 자...	2017-03-24 23:50:53	0
<input type="checkbox"/> 수정 복사 삭제		157	joonew	연 : 39번~45번 수: 3회 1~15번 외 : p42~p47	2017-03-27 00:00:12	0
<input type="checkbox"/> 수정 복사 삭제		158	joonew	수:16~30번 연,외:오답노트	2017-03-30 11:47:55	0



1. MySQL

DB

DB에서 데이터를 가져올 때 사용하는 코드를 query 라고 한다.
단어 뜻 그대로 질문(질의)을 통해 데이터를 가져오는 것이다.

```
SELECT `note_idx`, `note_id`, `content_text`, `note_date`, `isDisabled` FROM  
`note` WHERE 1
```

```
INSERT INTO `note` (`note_idx`, `note_id`, `content_text`, `note_date`,  
`isDisabled`) VALUES ([value-1],[value-2],[value-3],[value-4],[value-5])
```

```
UPDATE `note` SET `note_idx`=[value-1],`note_id`=[value-  
2],`content_text`=[value-3],`note_date`=[value-4],`isDisabled`=[value-5] WHERE 1
```

```
DELETE FROM `note` WHERE 1
```

2. phpmyadmin

DB 접속

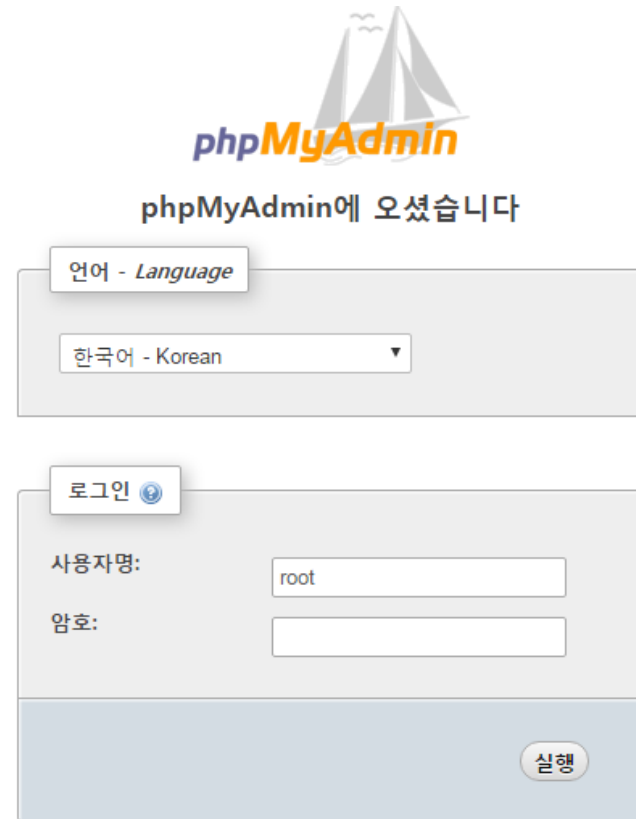
그러면 우리가 초반에 설치했던 DB에 접속해보자.

초반에 설치했던 phpmyadmin을 통해서 접속이 가능하다.

http://본인public ip/phpmyadmin

초반에 설치할 때 잘 따라했다면

이런 화면을 볼 수 있다.



The image shows the phpMyAdmin login page. At the top, there is a logo with a sailboat and the text 'phpMyAdmin'. Below the logo, it says 'phpMyAdmin에 오셨습니다' (Welcome to phpMyAdmin). There are two main sections: a language selection section and a login section. The language section has a dropdown menu currently set to '한국어 - Korean'. The login section has fields for '사용자명:' (Username) with 'root' entered and '암호:' (Password) which is empty. A '로그인' (Login) button is at the bottom right of the login section.

언어 - *Language*

한국어 - Korean ▼

로그인 ⓘ

사용자명: root

암호:

실행

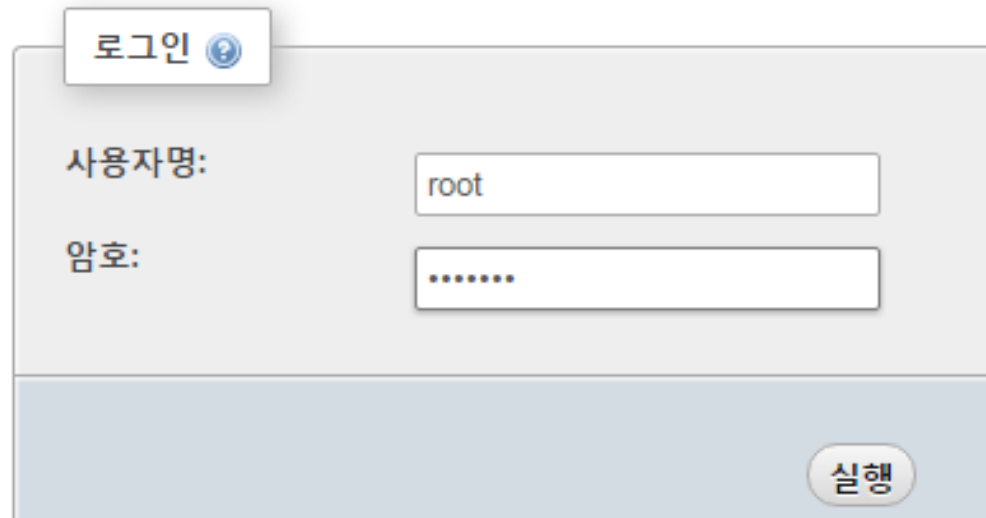
2. phpmyadmin

DB 접속

DB 접속은

ID : root

PW : 각자 중간 중간 입력했던 비밀번호



The image shows a screenshot of the phpMyAdmin login interface. At the top left, there is a tab labeled '로그인' (Login) with a blue circular icon containing a white key symbol. Below this, the form is divided into two sections. The first section is labeled '사용자명:' (Username:) and contains a text input field with the value 'root'. The second section is labeled '암호:' (Password:) and contains a text input field with masked characters '.....'. At the bottom right of the form, there is a blue button labeled '실행' (Execute).

2. phpmyadmin

DB 접속

로그인에 성공하면, 아래와 같이 나옵니다.



2. phpmyadmin

데이터베이스 / 테이블 만들기

이제 DB 와 table, row를 순차적으로 만들자.

데이터베이스 클릭



2. phpmyadmin

데이터베이스 / 테이블 만들기

새 데이터베이스 만들기에 아래와 같이 입력하고 만들기 클릭

데이터베이스

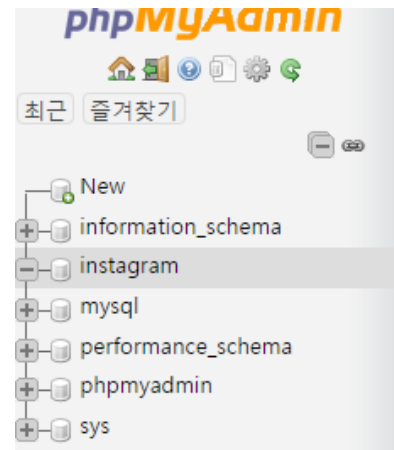
새 데이터베이스 만들기 ?

instagram utf8_general_ci **만들기**

데이터베이스 데이터정렬방식

☐ information_schema utf8_general_ci [사용권한 검사](#)

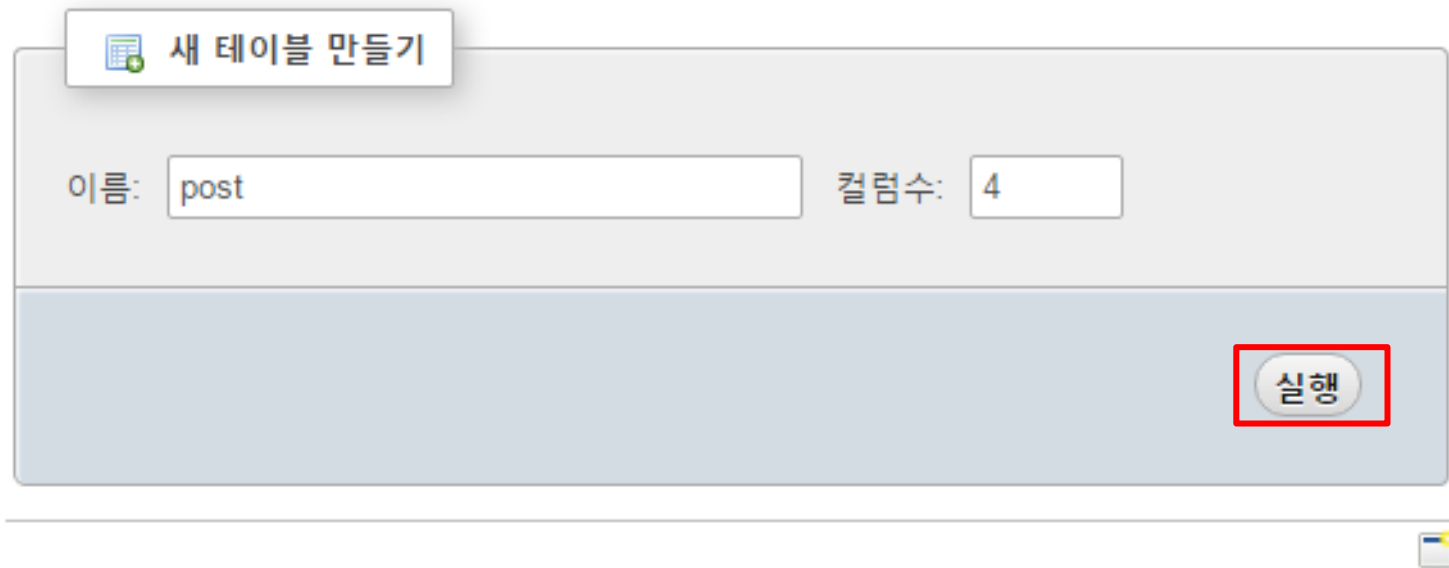
왼쪽 메뉴에 보면 데이터베이스가 생성된 것을 확인할 수 있다.



2. phpmyadmin

데이터베이스 / 테이블 만들기

Instagram database를 클릭하고 새 테이블을 만든다.



새 테이블 만들기

이름: 컬럼수:

실행



2. phpmyadmin

데이터베이스 / 테이블 만들기

컬럼은 이런 식으로 만들어 준다.

이름	종류	길이/값*	A_I
<input type="text" value="post_idx"/>	<input type="text" value="INT"/>	<input type="text"/>	<input checked="" type="checkbox"/>
<small>Pick from Central Columns</small>			
<input type="text" value="user_id"/>	<input type="text" value="VARCHAR"/>	<input type="text" value="30"/>	<input type="checkbox"/>
<small>Pick from Central Columns</small>			
<input type="text" value="img_src"/>	<input type="text" value="TEXT"/>	<input type="text"/>	<input type="checkbox"/>
<small>Pick from Central Columns</small>			
<input type="text" value="reg_date"/>	<input type="text" value="VARCHAR"/>	<input type="text" value="30"/>	<input type="checkbox"/>
<small>Pick from Central Columns</small>			

저장해준다.

2. phpmyadmin

데이터베이스 / 테이블 만들기

테이블이 만들어졌다.

이제 Node.js 와 연동을 해보자


 테이블 구조



 Relation view

	#	이름	종류	데이터정렬방식	보기	Null	기본값	추가	실행
<input type="checkbox"/>	1	post_idx 	int(11)			아니오	없음	AUTO_INCREMENT	 변경  삭제  기본  고유값 ▼ 더보기
<input type="checkbox"/>	2	user_id	varchar(30)	utf8_general_ci		아니오	없음		 변경  삭제  기본  고유값 ▼ 더보기
<input type="checkbox"/>	3	img_src	text	utf8_general_ci		아니오	없음		 변경  삭제  기본  고유값 ▼ 더보기
<input type="checkbox"/>	4	reg_date	varchar(30)	utf8_general_ci		아니오	없음		 변경  삭제  기본  고유값 ▼ 더보기

 ☐ 모두 체크

선택한 것을:

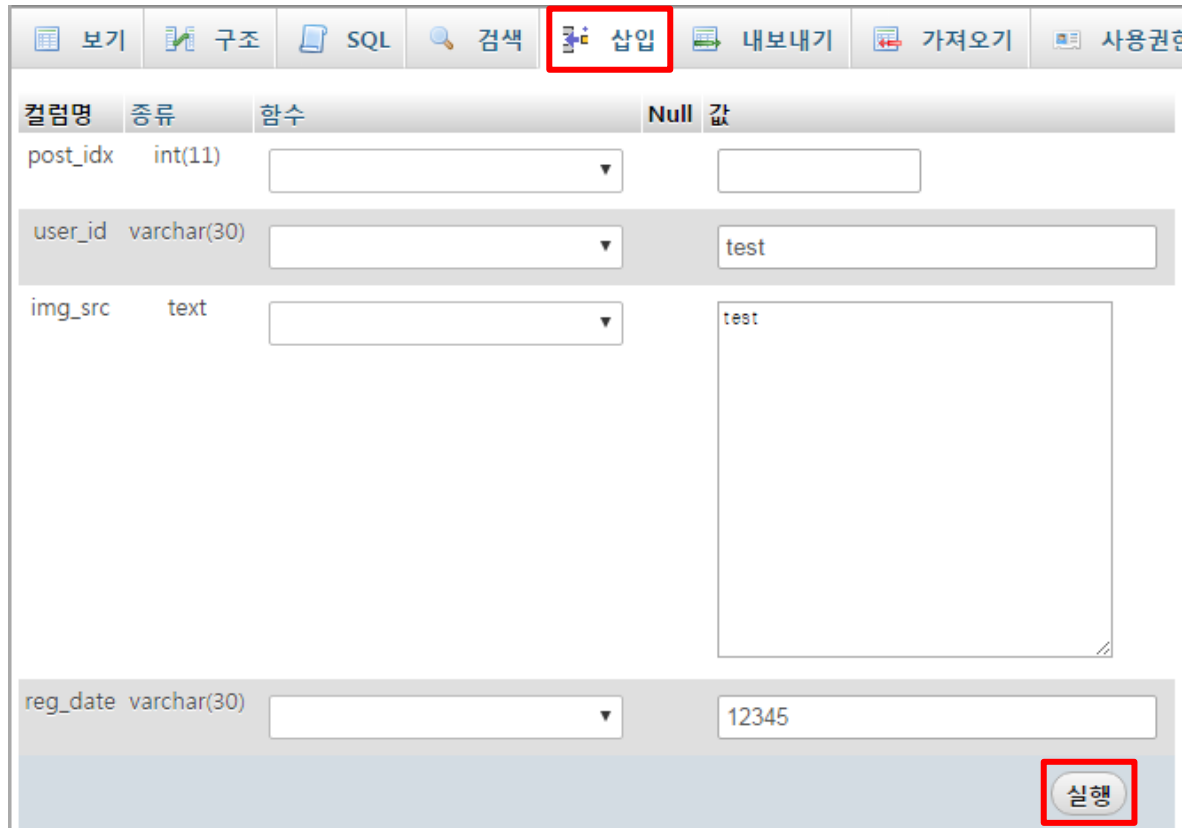
 보기  변경  삭제  기본  고유값  인덱스

 Add to central columns  Remove from central columns

3. DB 연동

Phpmyadmin 에 데이터 넣기

먼저 연동되는지 확인하기 위한 예시 데이터를 넣어보자.
삽입을 클릭하여 아래와 같이 데이터를 입력해보자.
이때 post_idx는 값을 비워둬도 된다.



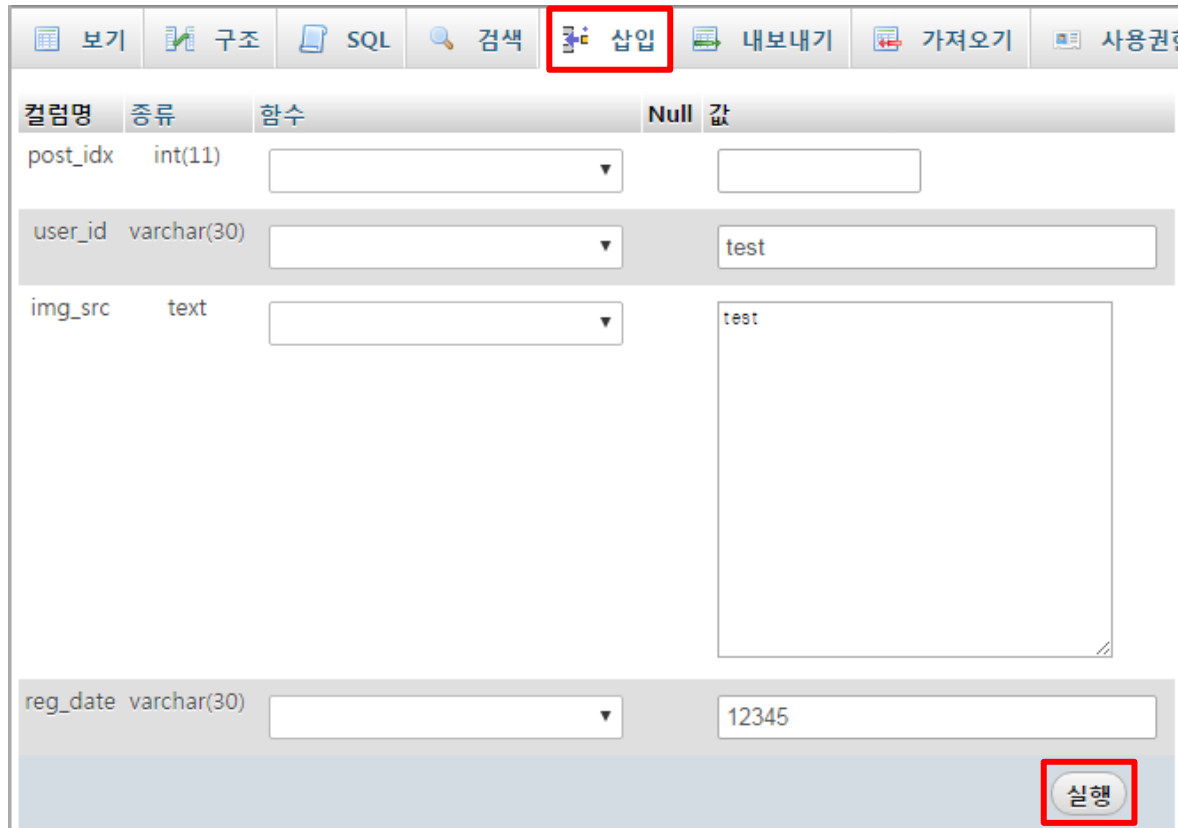
컬럼명	종류	함수	Null	값
post_idx	int(11)			
user_id	varchar(30)			test
img_src	text			test
reg_date	varchar(30)			12345

3. DB 연동

Phpmyadmin 에 데이터 넣기

그 이유는 아까 컬럼 생성시 A.I(Auto Increment)를 선택하여 자동으로 값이 증가하면서 추가되기 때문이다.

값을 다 입력했으면 실행을 클릭.



컬럼명	종류	함수	Null	값
post_idx	int(11)			
user_id	varchar(30)			test
img_src	text			test
reg_date	varchar(30)			12345

실행

3. DB 연동

Phpmyadmin 에 데이터 넣기

실행을 클릭하면 아래와 같이 열이 삽입되었다고 나온다.

✓ 1 열이 삽입되었습니다.
삽입된 행의 id: 1

```
INSERT INTO `post` (`post_idx`, `user_id`, `img_src`, `reg_date`) VALUES (NULL, 'test', 'test', '12345');
```

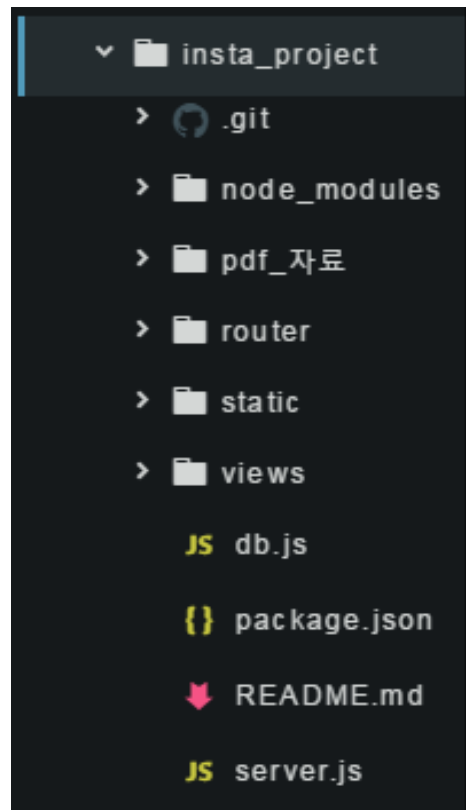
보기 메뉴에서도 아래와 같이 데이터가 들어간 것을 확인할 수 있다.

+ 옵션					
← T →		post_idx	user_id	img_src	reg_date
<input type="checkbox"/>	 수정	1	test	test	12345

3. DB 연동

Node.js – DB 연결

이제 Atom을 켜고 프로젝트 폴더에 db.js 파일을 만든다.



3. DB 연동

Node.js – DB 연결

db.js 에 아래와 같이 작성한다.

Host – 'localhost'

User – 'root'

Password – 본인 비번

Database - instagram

```
var db_config = {  
  host      : 'localhost',  
  user      : 'root',  
  password  : 'webprac',  
  database  : 'instagram'  
};  
module.exports = db_config;
```

3. DB 연동

Node.js – DB 연결

Route.js 의 상단에 내용을 추가해 준다.

Require를 이용해 mysql 모듈을 가져오고, 아까 작성한 db.js 를 db_config 의 변수명을 가지는 모듈로 가져온 것이다.

```
var mysql = require('mysql');
var db_config = require('../db');

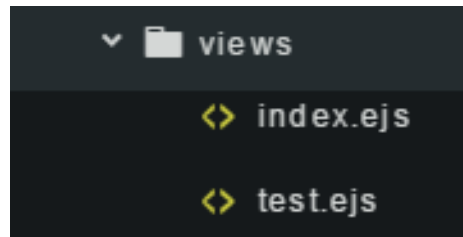
module.exports = function(app)
{
    app.get('/', function(req, res){
        res.render('index', {
            title: "insta_project"
        });
    });
};
```

3. DB 연동

Node.js – DB 연결

일단 DB가 잘 연결되는지 확인하는 것이 목적이므로, 예시용 페이지 하나를 만든다.

Views 에 test.ejs 파일을 하나 만들고 아래 내용을 채운다.



```
<html>
  <head>
    <title>
      테스트페이지
    </title>
  </head>
  <body>
    정보들여가는곳
  </bod>
</html>
```


3. DB 연동

Node.js – DB 연결

Route.js 에 내용을 추가한다.

```
var mysql = require('mysql');
var db_config = require('../db');

module.exports = function(app)
{
  app.get('/',function(req,res){
    res.render('index',{
      title: "insta_project"
    });
  });

  app.get('/test',function(req,res){
    var db = mysql.createConnection(db_config);
    var sql = 'SELECT * FROM `post` WHERE 1';
    db.query(sql, function(err,rows){
      if(err){
        console.error('mysql connection error');
        console.error(err);
        throw err;
      }
      var posts = rows;
      var isWrite = 0;
      res.render('test',{
        posts:posts
      });
      db.end();
    });
  });
};
```

3. DB 연동

Node.js – DB 연결

본인ip/test 로 접속해보면?

```
Cannot GET /test
```

접속이 안된다 ㅠ

서버 측 내용은 터미널에서 현재 켜져있는 node.js의 연결을 끊었다가 다시 연결해 줘야 반영이 된다!



3. DB 연동

Node.js – DB 연결

먼저 ctrl+c 를 눌러서 node.js 의 연결을 끊는다.

(tmux 로 연결이 되어있다면

\$ tmux attach -t 본인세션이름

으로 들어가서 ctrl+c)

```
ubuntu@ip-172-31-6-29:~$ tmux attach -t insta is
```

```
ubuntu@ip-172-31-6-29:~$ cd insta_project/
ubuntu@ip-172-31-6-29:~/insta_project$ nodesserver.js
Express server has started on port 3000
^C
ubuntu@ip-172-31-6-29:~/insta_project$
```

3. DB 연동

Node.js – DB 연결

다시 node server.js 를 입력해서 연결한다.

```
ubuntu@ip-172-31-6-29:~/insta_project$ node server.js  
Express server has started on port 3000
```

브라우저에서 확인 가능하다.

정보들어가는곳

3. DB 연동

Node.js – DB 연결

※ 여기서 오류가 뜨면?

```
$ mysql -u root -p
```

그리고 본인 패스워드 입력하면

아래와 같이 mysql> 이라는 입력창으로 바뀐다

```
ubuntu@ip-172-31-6-29:~/insta_project$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 7
Server version: 5.7.17-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> █
```

3. DB 연동

Node.js – DB 연결

※ 여기서 오류가 뜨면?

```
GRANT ALL ON insta.* TO 'root'@'127.0.0.1' IDENTIFIED BY 'webprac';  
flush privileges;
```

을 차례대로 입력해주고 exit

```
mysql> GRANT ALL ON insta.* TO 'root'@'127.0.0.1' IDENTIFIED BY 'webprac';  
Query OK, 0 rows affected, 1 warning (0.00 sec)  
  
mysql> flush privileges;  
Query OK, 0 rows affected (0.00 sec)  
  
mysql> exit
```

3. DB 연동

Node.js – DB 연결

※ 여기서 오류가 뜨면?

마무리로

\$ sudo service mysql restart

```
mysql> exit
Bye
ubuntu@ip-172-31-6-29:~/insta_project$ sudo service mysql restart
ubuntu@ip-172-31-6-29:~/insta_project$
```

그리고 다시 node server.js 해준다.

3. DB 연동

Node.js – DB 연결

이제 DB연결도 됐으니 아까 가져온 정보를 브라우저에 띄워 보자

Test.ejs에서 body부분을 아래와 같이 바꿔준다.

```
<body>  
  <%=posts%>  
</bod>
```

그리고 브라우저를 다시 보면

[object Object]

뭔가 오브젝트가 뜨긴 뜬 것을 확인할 수 있다.

3. DB 연동

Node.js – DB 연결

우리는 자세한 내용을 알고 싶으니 body 부분을 다시 바꿔준다.

```
<body>
  <%-JSON.stringify(posts)%>
</bod>
```

그리고 브라우저를 확인해보면?

```
[{"post_idx":1,"user_id":"test","img_src":"test","reg_date":"12345"}]
```

우리가 DB에 추가했었던 정보가 뜨는 것을 확인할 수 있다.

이로써 DB 연결 성공!

3. DB 연동

※ 매번 꺾다가 켜기가 귀찮다. >> nodemon

서버측 파일을 변경하고 나서 매번 꺾다가 켜기가 귀찮다.

그래서 tmux 마냥 또 하나의 프로그램을 설치한다.

바로 nodemon 이라는 프로그램인데, 이 프로그램은 서버측 파일이 변경되면 자동으로 서버를 재실행 시켜준다.

먼저 터미널에

\$ sudo npm install nodemon -g

입력

```
ubuntu@ip-172-31-6-29:~/insta_project$ sudo npm install nodemon -g
loadDep:nopt → 304
```

3. DB 연동

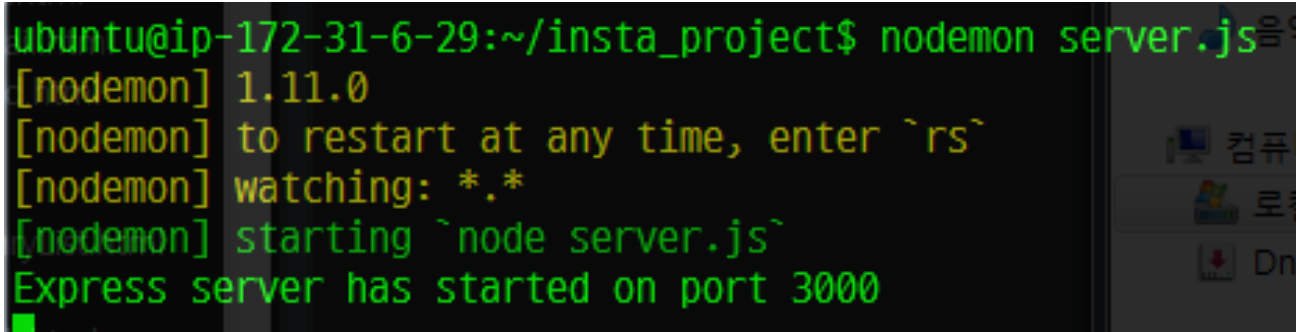
※ nodemon

설치가 되고 나면 실행은 아주 쉽다

\$ node server.js 하던 것을

\$ nodemon server.js

로 바꿔서 입력한다.

A terminal window screenshot showing the installation and usage of nodemon. The prompt is 'ubuntu@ip-172-31-6-29:~/insta_project\$'. The command 'nodemon server.js' is entered. The output shows '[nodemon] 1.11.0', '[nodemon] to restart at any time, enter `rs`', '[nodemon] watching: *.*', '[nodemon] starting `node server.js`', and 'Express server has started on port 3000'.

```
ubuntu@ip-172-31-6-29:~/insta_project$ nodemon server.js
[nodemon] 1.11.0
[nodemon] to restart at any time, enter `rs`
[nodemon] watching: *.*
[nodemon] starting `node server.js`
Express server has started on port 3000
```



다음 시간에는...

1. DB에서 데이터 가져와서, 웹페이지 콘텐츠 구성하기