

MemoNote semi Project 4 230202

dlhjwww.cafe24.com

카페24 호스팅 | 온라인 비즈니스의 시작


Tomcat JSP, node.js 기반의 호스팅 서비스는 다양한 개발 니즈를 만족하는 서비스를 구축할 수 있습니다. 웹, 트래픽 용량은 웹/스트리밍/이미지CDN 합산 용량입니다. Tomcat JSP 호스팅의 장점 단독 Tomcat 서버 제공 전 사양에 단독 Tomcat을 제공하여 서버릿 재구동이나 수정된 클래스 파일을 적용하기 위해 톰캣 재구동 시간을 기다릴 필요가
https://hosting.cafe24.com/?controller=new_product_page&page=language



▶ 서버 환경설정

- ☐ Tomcat 10.0.x / JSP 3.0 / Servlet 5.0 / JDK 17 / MariaDB 10.1.x UTF-8
- ☐ Tomcat 10.0.x / JSP 3.0 / Servlet 5.0 / JDK 11 / MariaDB 10.1.x UTF-8
- ☒ Tomcat 8.5.x / JSP 2.3 / Servlet 3.0 / JDK 8 / MariaDB 10.1.x UTF-8

Google Colaboratory

 <https://colab.research.google.com/drive/1q6erndsnZzuS3rDdKWRgRaRXrzelv6Q5?usp=sharing>



Python Basic

List



```
a = []
b = [1,2,3]
c = ['a','b','c']
d = [1, 2, 'a', 'b']

print(b[0]) #인덱싱
print(c[1:]) #슬라이싱
print(d[-1]) #마지막 요소는 -1

# a[0] = 9 # a 리스트에는 아무 원소도 없어서 에러 발생
b[0] = 9 # b 리스트는 원소가 있으므로 수정 가능
print(b)

print(c)
del c[2] # 리스트는 삭제가 가능하다.
print(c)

print(a)
a.append(0) # 리스트는 삽입이 가능하다.
print(a)
```



```
# 리스트에서 많이 사용되는 함수 정리

# 요소 추가 append 마지막에 요소추가
a.append(0)
print(a)
a.append('a')
print(a)

# 요소 추가 insert 특정 위치를 지정하여 맘대로 요소 추가
d = [1,2,3]
d.insert(0,4)

# 정렬 sort
a = [1,3,2,4]
a.sort()
print(a)

# 뒤집기 reverse
a.reverse()
print(a)

# 인덱스 index
a.index(3)
# a.index(5) 없는 요소는 에러 발생한다.
```



```
# 삭제 remove 중복되는 수는 처음 등장하는 하나만 삭제한다.
print()
a = [1,2,3,3,5]
a.remove(3)
print(a)

# 삭제 pop 리스트의 마지막 요소를 삭제한다.
a = [1,2,3]
a.pop() # 괄호에 값이 없으면 마지막 요소를 하나 삭제
print(a)

a = [1,2,3]
a.pop(0) # 괄호에 값이 있으면 그 요소를 삭제
print(a)

# 개수 세기 count 특정 요소의 갯수 세기

a = [1,2,3,3]
print(a.count(3))
```



```
[1, 2, 3, 5]
[1, 2]
[2, 3]
2
```

Tuple



```
# TUPLE
# () 소괄호로 만든다.
# immutable 수정이 불가능하다.
# indexing & slicing 이 가능하다.
# 한 개의 요소로 tuple을 만들때는 (1,) 쉼표를 괜히 넣는다.

a = ()
b = (1,) #한 개의 요소로 tuple을 만들때는 (1,) 쉼표를 괜히 넣는다.
c = (1,2,3)
d = (1,2,'a','b')

# del d[0]    immutable 삭제가 불가능하다.
# d[0] = 9    immutable 수정이 불가능하다.
# d.append(5) immutable 삽입이 불가능하다.

print(d[0])   # 인덱싱이 가능하다.
print(d[0:2]) # 슬라이싱이 가능하다. 마지막은 미만으로 처리한다.

print(len(d)) # 튜플 요소의 개수를 반환하는 함수
```



```
1
(1, 2)
4
```

Dict

```
# 딕셔너리에서 많이 사용되는 함수 정리
# 딕셔너리에서 키값들만 추출

a = {'이름': '홍길동', '나이': 20, '보유코인': ['비트코인', '에이다', '이더리움']}
print(a.keys())

# 딕셔너리에서 밸류값들만 추출
print(a.values())

# 쌍으로 추출 (튜플 쌍으로 추출)
print(a.items())

# 키값으로 밸류값 추출
print(a.get('이름'))
print(a.get('몸무게')) # 찾는 키값이 없을때 None 출력한다.

# 키값으로 밸류값 추출 (인덱싱이 불가능하기 때문에 꺼낼때 많이 사용된다.)
print(a['이름'])
# print(a['몸무게']) # 찾는 키값이 없을때 예외 발생된다.

# 키가 딕셔너리안에 있는지 확인
print('이름' in a) # 찾는 키값이 있을때 true 반환
print('몸무게' in a) # 찾는 키값이 없을때 false 반환
```

dict_keys(['이름', '나이', '보유코인'])
dict_values(['홍길동', 20, ['비트코인', '에이다', '이더리움']])
dict_items([('이름', '홍길동'), ('나이', 20), ('보유코인', ['비트코인', '에이다', '이더리움'])])
홍길동
None
홍길동
True
False

Set

```

# SET
# {} 중괄호로 만든다.
# mutable 수정 가능하다.
# indexing & slicing 이 불가능하다.
# unordered 순서가 없다.
# 중복데이터를 허용하지 않는다.

# a = {} # 딕셔너리와 셋은 동일하게 {} 를 사용한다. 따라서 빈 값으로 생성할때는 주의하자.
a = set()
b = {1,2,3}
c = {1,2,'a','b'}
d = {1,2,2,2,3,4,4}
print(d)

a.add(0).....# mutable 삽입 가능하다.
print(a)
print(c)
c.remove(2) # mutable 삭제 가능하다.
print(c)

```

{1, 2, 3, 4}
 {0}
 {'b', 1, 2, 'a'}
 {'b', 1, 'a'}

LIST,TUPLE,DICT,SET 네 가지 자료형 종합 표정리

```

▶ # 불 Bool 자료형 boolean

# 반드시 대문자로 시작한다.

a = True
b = False
c = "true"

print(type(a))
print(type(c))

print(1 == 1)
print(1 != 1)

```

```

↳ <class 'bool'>
   <class 'str'>
   True
   False

```

```

▶ # 키보드로 입력 받기 |

```

```

a = input()
print(a)

```

```

↳ 하이
   하이

```

```

[82] a = input("지금이 오후인가요?")
     print(a)

```

```

지금이 오후인가요?네네
네네

```

```

[84] # 파이썬에서 input 은 항상 일단 먼저 모두 문자 취급한다.

```

```

#a = input("지금이 몇시예요")
#print(a+1)

a = int(input("지금 부터 한시간 뒤는 몇 시 인가요"))
print(a+1)

```

```

지금이 몇시예요3
4

```

🔒 화면 공유 중입니다

069

Create a tuple containing the names of five countries and display the whole tuple. Ask the user to enter one of the countries that have been shown to them and then display the index number (i.e. position in the list) of that item in the tuple.

print() input() index()

1 예건 김

```
a = ("korea", "japan", "china", "america", "somecontry")
print(a)
b = input()
print(a.index(b))
```

2 혁주 이

```
c = ('Korea', 'USA', 'India', 'Japan', 'Brazil')
print(c)
a = input("인덱스를 추출할 나라 이름 : ")
print(c.index(a))
```

069

```
country_tuple = ("France", "England", "Spain", "Germany", "Australia")
print(country_tuple)
print()
country = input("Please enter one of the countries from above: ")
print(country, "has index number", country_tuple.index(country))
```

071

Create a list of two sports. Ask the user what their favourite sport is and add this to the end of the list. Sort the list and display it.

1 예건 김

```
sports = ['soccor', 'baseball']
b = input("가장 좋아하는 스포츠는?")
sports.append(b)
sports.sort()
print(sports)
```

2 강세빈

```
sport = ['soccer', 'baseball']
sport.append(input('가장 좋아하는 스포츠가 무엇인가요?\n'))
print(sport)
```

3. 이혁주

```
s = ['헬스', '복싱']
```



```
a = input("좋아하는 스포츠 : ")
s.append(a)s.sort()
print(s)
```

071

```
sports_list = ["tennis","football"]
sports_list.append(input("What is your favourite sport? "))
sports_list.sort()
print(sports_list)
```

072

Create a list of six school subjects. Ask the user which of these subjects they don't like. Delete the subject they have chosen from the list before you display the list again.

```
1. 김지수
subjects = ['국어', '수학', '영어', '체육', '음악', '미술']
a = input('싫어하는 과목?')
subjects.remove(a)
print(subjects)
```

2. 이혁주

```
s = ['국어', '영어', '수학', '체육', '미술', '음악']
print(s)
a = input("싫어하는 과목 : ")
s.remove(a)
print(s)
```

072

```
subject_list = ["maths","english","computing","history","science","spanish"]
print(subject_list)
dislike = input("Which of these subjects do you dislike? ")
getrid = subject_list.index(dislike)
del subject_list[getrid]
print(subject_list)
```

073

Ask the user to enter four of their favourite foods and store them in a dictionary so that they are indexed with numbers starting from 1. Display the dictionary in full, showing the index number and the item. Ask them which they want to get rid of and remove it from the list. Sort the remaining data and display the dictionary.

김지수

```
foods = {'1':input("좋아하는 음식 첫 번째"), '2':input("좋아하는 음식 두 번째"), '3':input("좋아하는 음식 세 번째"), '4':input("좋아하는 음식 네 번째"))
print(foods)

a = input('제외할 음식?')
del foods[(a)]
print(foods)
```

073

```
food_dictionary = {}
food1 = input("Enter a food you like: ")
food_dictionary[1] = food1
food2 = input("Enter another food you like: ")
food_dictionary[2] = food2
food3 = input("Enter a third food you like: ")
food_dictionary[3] = food3
food4 = input("Enter one last food you like: ")
food_dictionary[4] = food4
print(food_dictionary)
dislike = int(input("Which of these do you want to get rid of? "))
del food_dictionary[dislike]
print(sorted(food_dictionary.values()))
```

074

Enter a list of ten colours.
Ask the user for a starting
number between 0 and 4
and an end number
between 5 and 9. Display
the list for those colours
between the start and end
numbers the user input.

리스트에서 슬라이싱 문제

1 권두현

2 강세빈

```
color = ['red', 'green', 'blue', 'orange', 'yellow', 'purple', 'black', 'white', 'navy', 'gray']start = int(input('0~4사이의 숫자를 입력해'))
```

3 박성준

4 김예건

```
color_list = ["빨", "주", "노", "초", "파", "남", "보", "흰", "검", "회"]a = int(input('시작숫자(0~4)를 입력해주세요'))b = int(input('끝숫자(5~9)를 입력해주세요'))
```

5 김지수

```
colors = ['red', 'yellow', 'orange', 'green', 'blue', 'black', 'white', 'gray', 'purple', 'navy']a= int(input('0 ~ 4 사이 숫자를 입력해주세요'))
```

6 변예린

```
colours = ('빨', '주', '노', '초', '파', '남', '보', '흰', '검', '분')a = (int)(input('시작 숫자? '))b = (int)(input('끝 숫자? '))print(colours[a:b])
```

7 이혁주

```
c0 = input("리스트에 넣을 색 : ")c1 = input("리스트에 넣을 색 : ")c2 = input("리스트에 넣을 색 : ")c3 = input("리스트에 넣을 색 : ")c4 = input("리스트에 넣을 색 : ")
```

```
a = ['빨', '주', '노', '초', '파', '남', '보', '검', '흰', '그레이']
q1 = int(input())
q2 = int(input())
print(a[q1:q2])
```

0
5
['빨', '주', '노', '초', '파']

```
a = [ "빨", "주", "노", "초", "파", "남", "보", "검", "흰", "분" ]
```

```
ask = int(input("시작할 위치 (0~4)"))
```

```
ask2 = int(input("끝낼 위치 (5~9)"))
```

```
print(a[ask:ask2])
```

시작할 위치 (0~4) 3

끝낼 위치 (5~9) 7

['초', '파', '남', '보']

074

```
colours = ["red", "blue", "green", "black", "white", "pink", "grey", "purple", "yellow", "brown"]
start = int(input("Enter a starting number (0-4): "))
end = int(input("Enter an end number (5-9): "))
print(colours[start:end])
```

Deploy

Basic

Cefe24 Setting

MySQL 아이피 등록은 하는 것이 좋다. (경험상)

• MySQL 외부 IP 접근설정

사용유무	사용중 👉 설정하기
등록된 IP	111.118.52.74

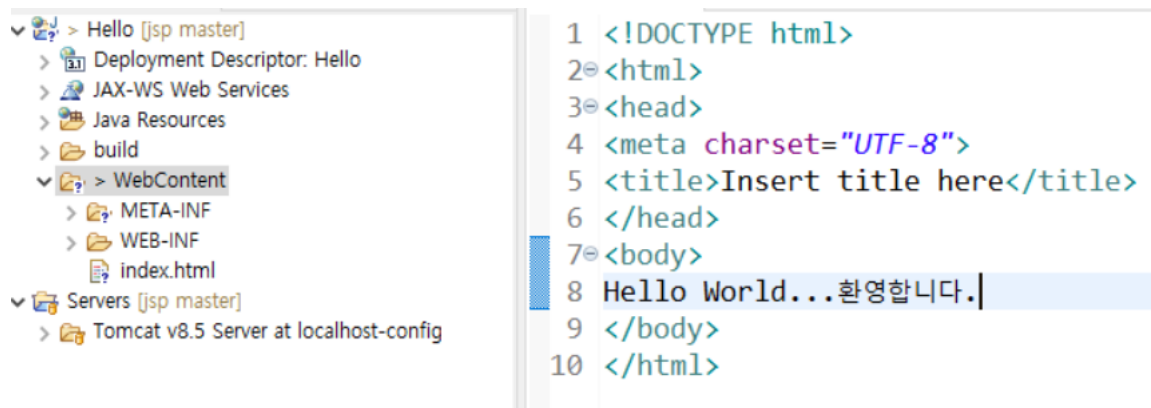
• FTP 접속 설정

접속 설정	국내에서만 접속 허용 👉 설정하기
접속 차단 해제 ?	👉 FTP 접속차단 해제

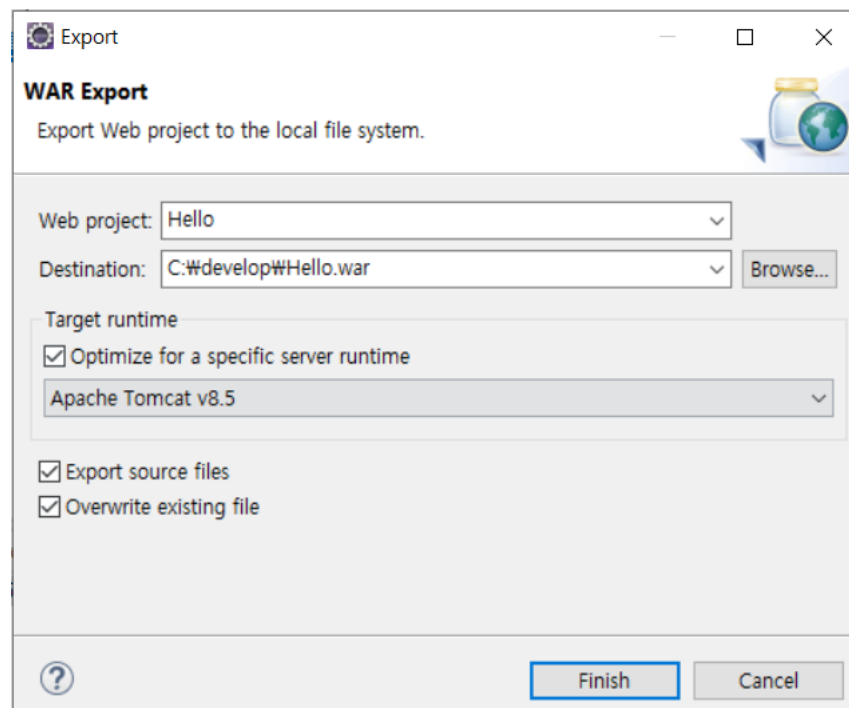
- FTP 비밀번호 변경
- DB 비밀번호 변경

FTP 비번과 DB 비번은 동일하게 설정하는 것을 추천(헤깔린다)

test



기본 프로젝트 작성...



전부 체크 후.....생성한 war 를 cafe24로 ftp 전송 해주는 프로그램 설치

[←](#)
[→](#)
[↻](#)
[🏠](#)
filezilla-project.org
[🔖](#)
[🌐](#)
[🔍](#)
[🔄](#)
[📶](#)
[📄](#)
[🐱](#)

[📱](#)
[📧](#)
[📺](#)
[📷](#)
[📺](#)
[📺](#)
[📺](#)
[📺](#)
[📺](#)
[📺](#)
[📺](#)
[📺](#)
[📺](#)
[📺](#)
[📺](#)



FileZilla

The free FTP solution

[Home](#)

FileZilla

- Features
- Screenshots
- Download
- Documentation
- FileZilla Pro

FileZilla Server

- Download

Community

- Forum
- Wiki

General

- FAQ
- Support
- Contact
- License
- Privacy Policy
- Trademark Policy

Development

- Source code
- Nightly builds
- Translations
- Version history
- Changelog
- Issue tracker

Other projects

Promotion:



FileZilla® Pro

The Best FTP Solution



[GET I](#)

Overview

Welcome to the homepage of FileZilla®, the free FTP solution. The *FileZilla Client* not only supports FTP but also FTP over TLS (FTPS) and SFTP. It is open source software distributed free of charge under the terms of the GNU General Public License.

We are also offering *FileZilla Pro*, with additional protocol support for WebDAV, Amazon S3, B2, Dropbox, Microsoft OneDrive, Google Drive, Microsoft Azure Blob and File Storage Service.

Last but not least, *FileZilla Server* is a free open source FTP and FTPS Server.

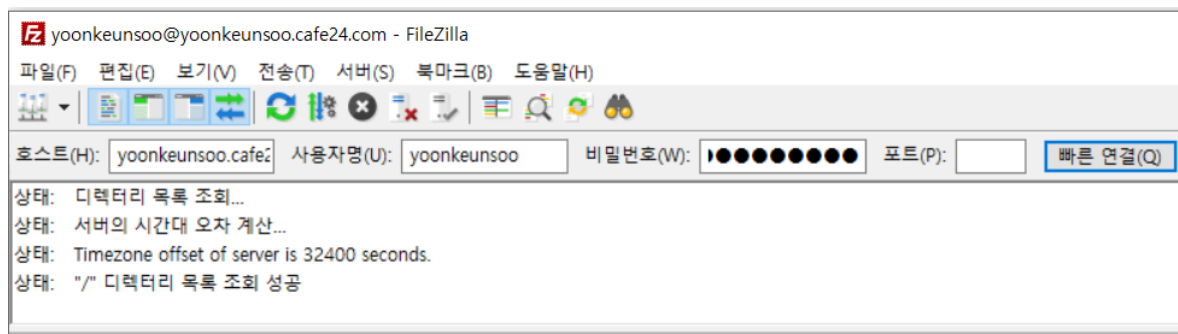
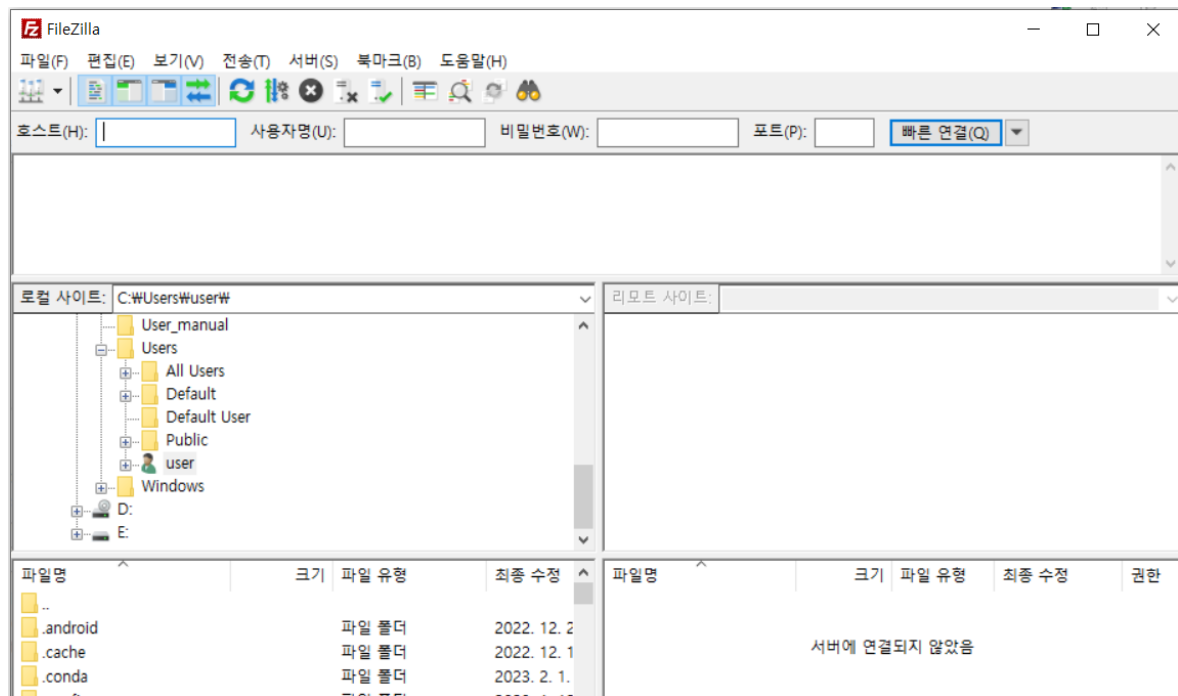
Support is available through our [forums](#), the [wiki](#) and the [bug and feature request tracker](#).

In addition, you will find documentation on how to compile FileZilla and nightly builds for various platforms in the development section.

Quick download links





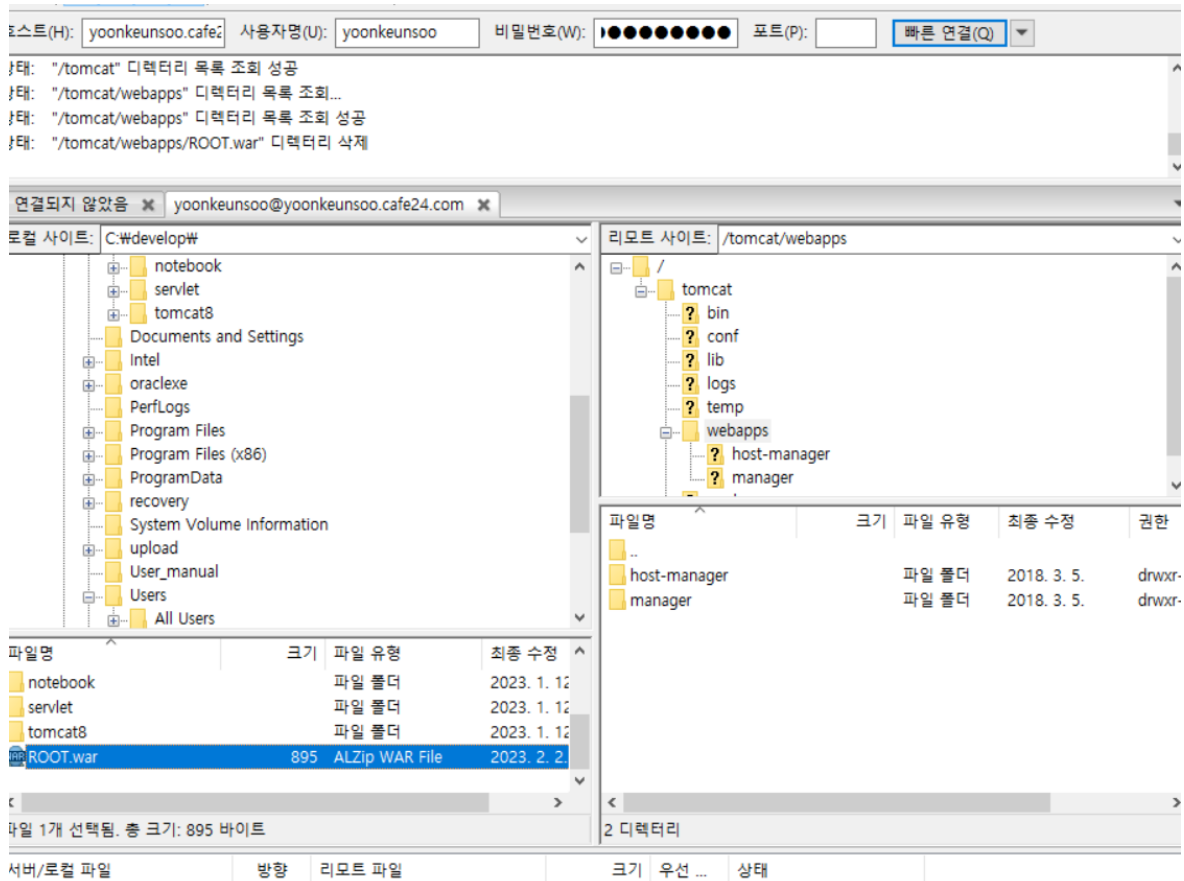


호스트 칸 : 아이디.cafe24.com

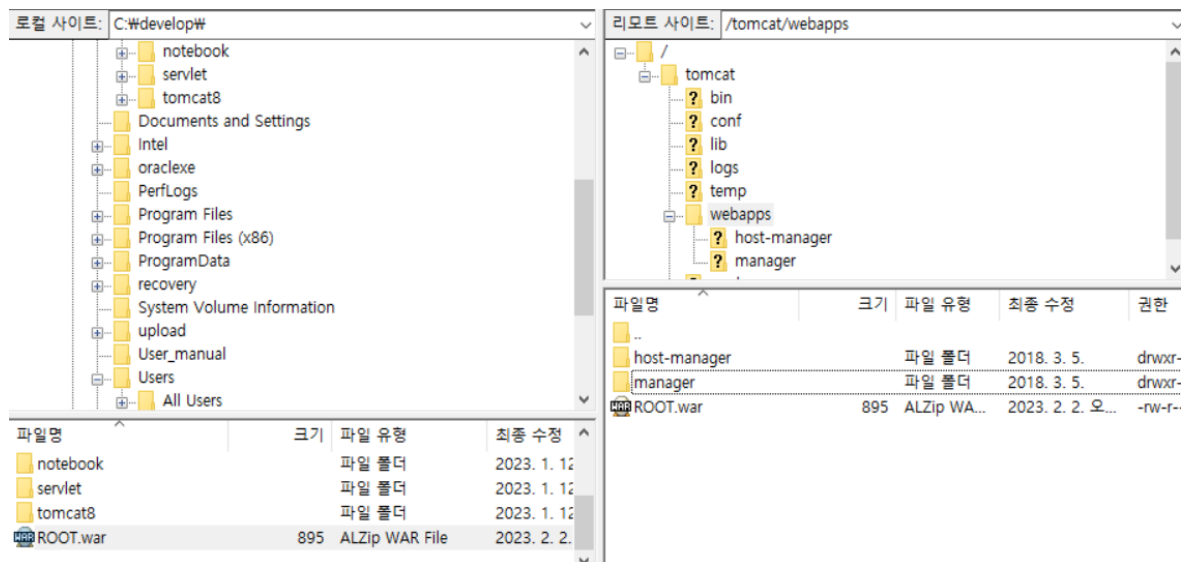
사용자명 칸 : 아이디

비밀번호 : 비번 <— FTP 비번

포트 : 빈칸



생성한 war 파일의 이름을 ROOT.war 로 변경하고 업로드

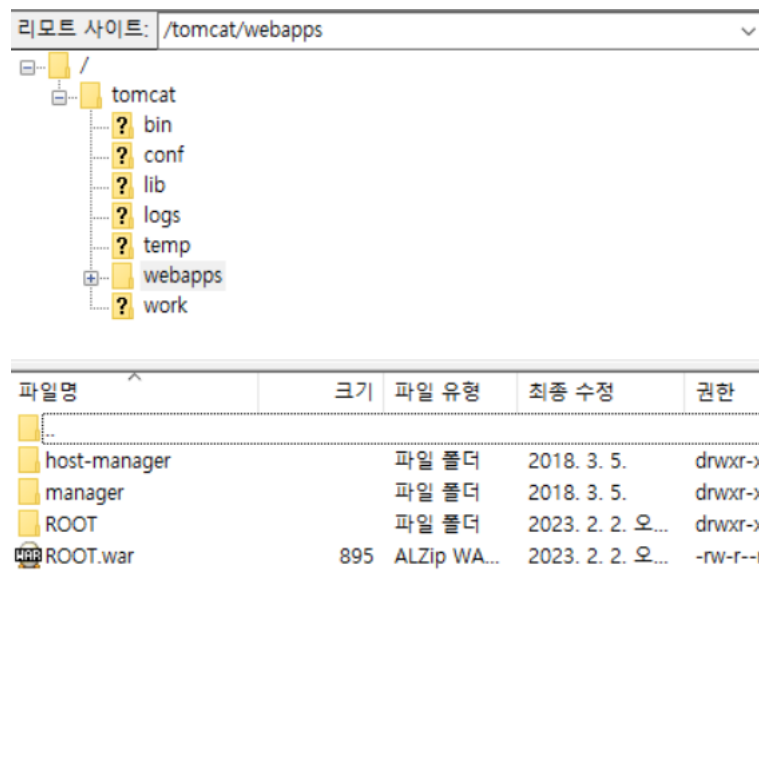


위에서 생성한 war파일의 이름을 ROOT.war 로 변경하여

오른쪽 즉 cafe24 로 보낸다 (더블클릭 또는 드래그 앤 드랍)

그리고 파일질라를 완전히 종료 후 다시 들어가면 ROOT폴더가 생성되어 있다.

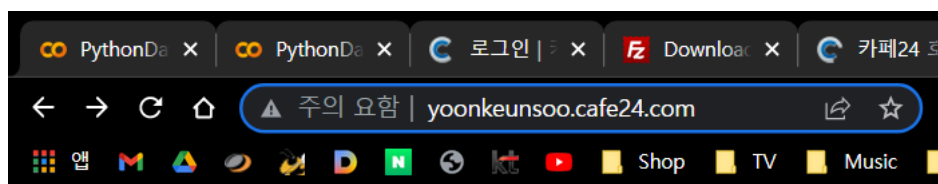
설정 끝...



파일 질라를 종료 후 다시 들어가면 ROOT 폴더가 생성되어 있다.

주소에 https <— s 를 삭제한다.

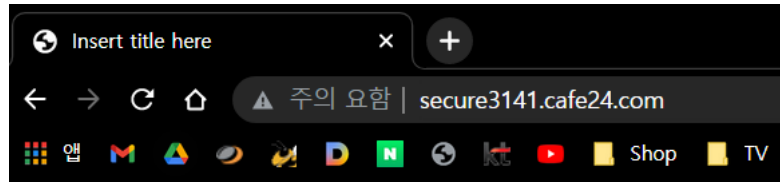
<http://yoonkeunsoo.cafe24.com/>



Hello World...환영합니다.

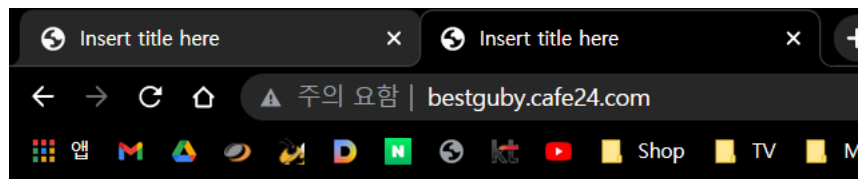
일단 성공!!!

권두현님 성공!!



안녕하세요 hello world

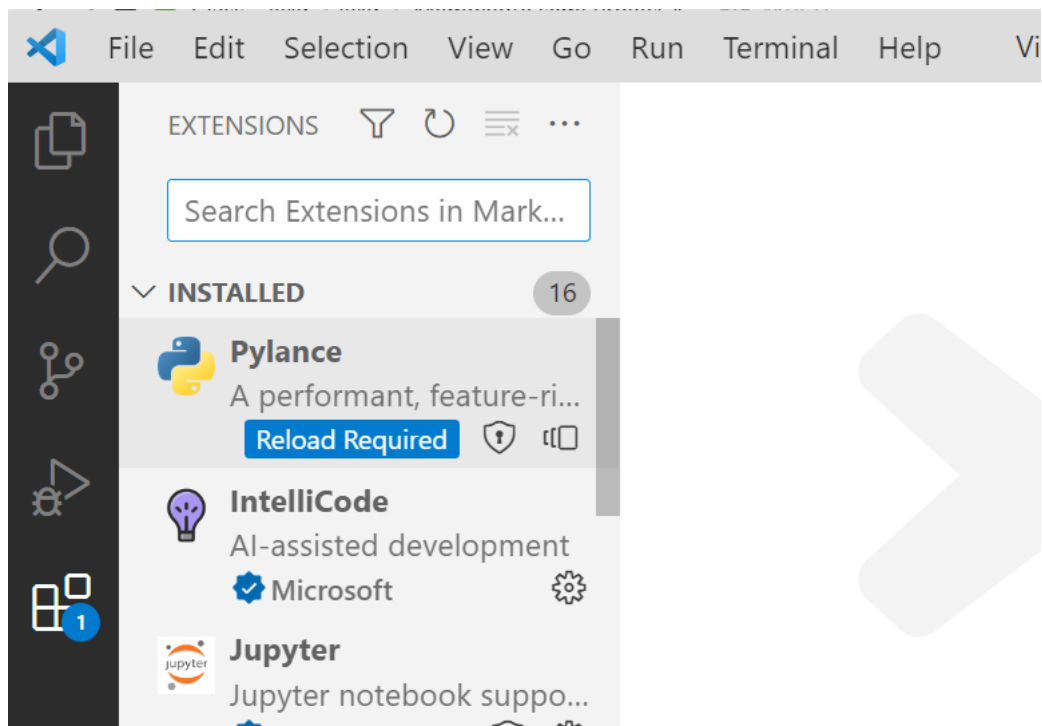
박성준님 성공



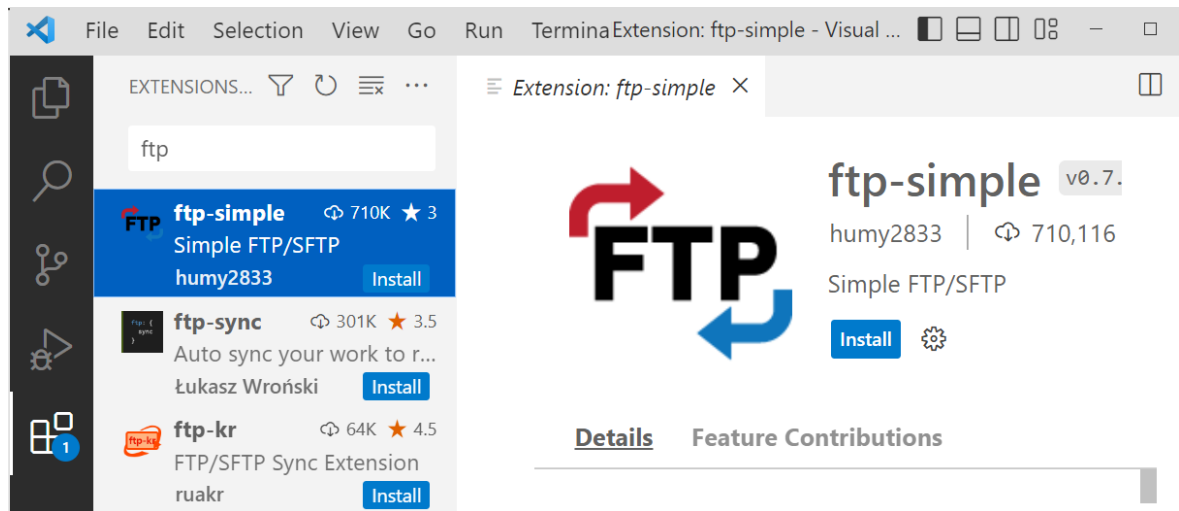
Hello World... 환영합니다.

사이트의 일부 내용등을 조금 수정할때도 새로운 와르를 만들어서 다시 파일질라로 보내기는 부담스럽다.

이렇게 조금 변경을 할때는 VS Code 를 사용하면 아주 편하다.

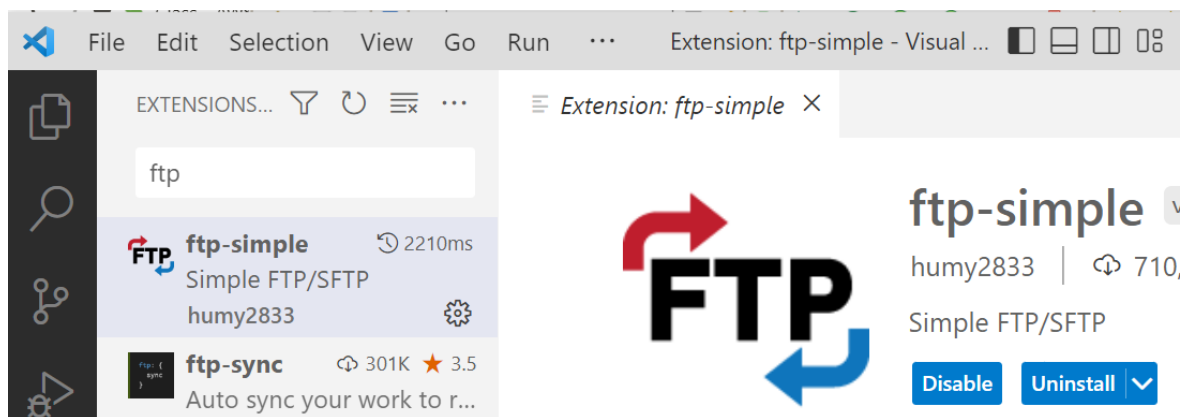


왼쪽하단 익스텐션

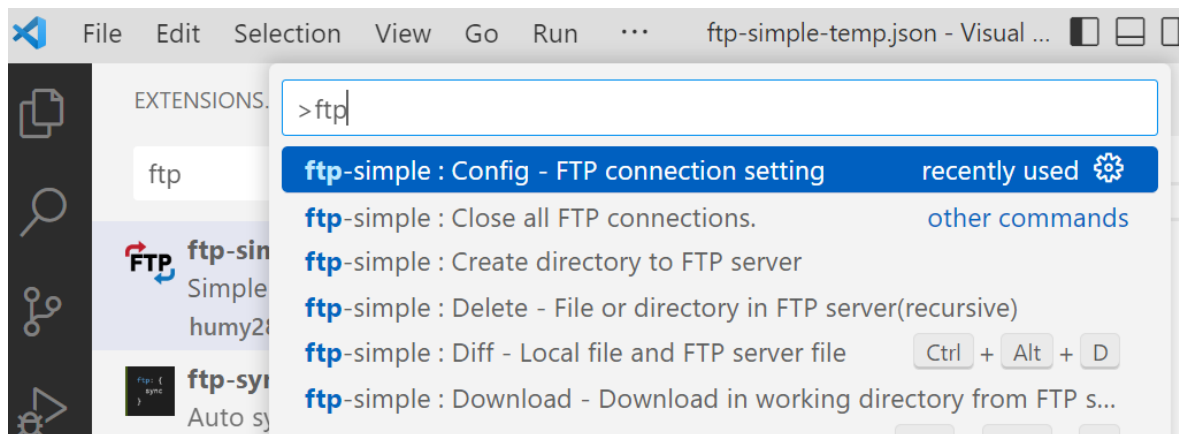
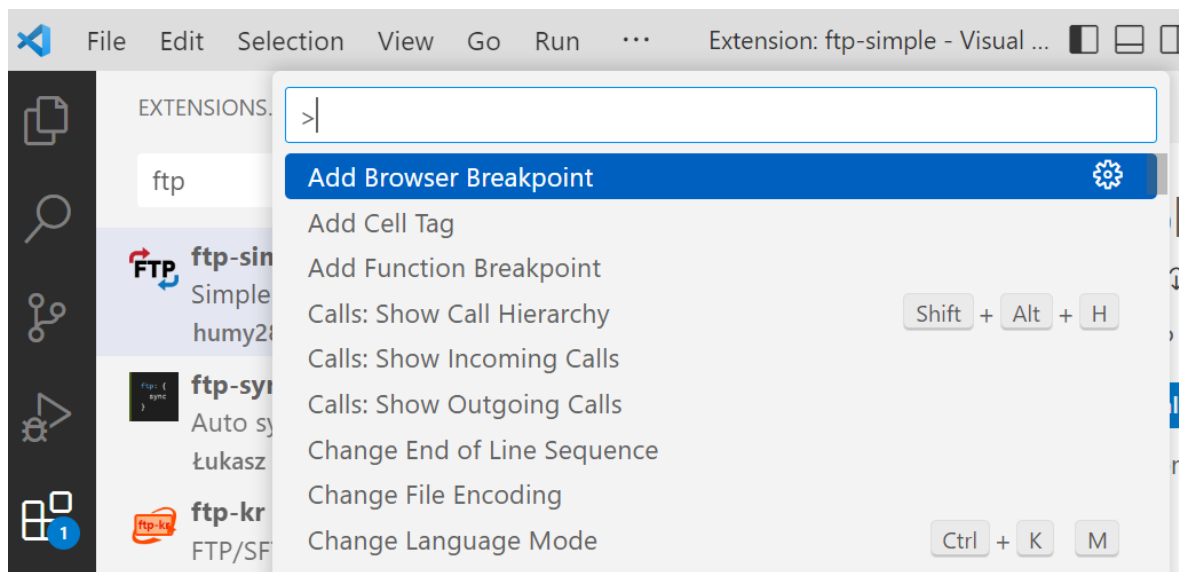


ftp 로 검색하여 ftp simple 설치

설치 종료 후



설정 들어가 F1 - 검색: ftp

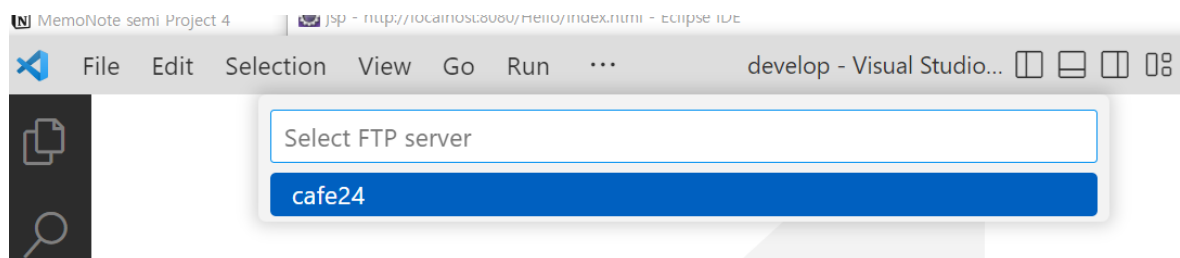
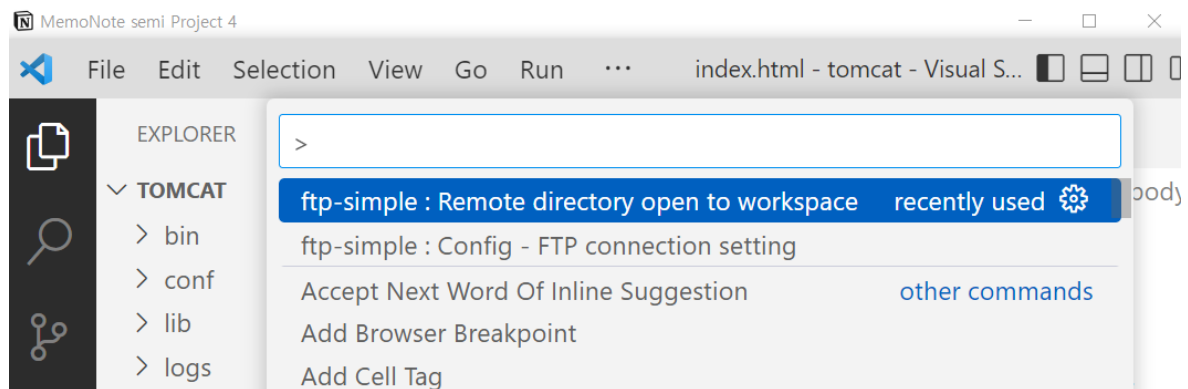


```
{} ftp-simple-temp.json ×
lser > globalStorage > humy2833.ftp-simple > {} ftp-simple-temp.json > ...
1  [
2      {
3          "name": "cafe24",
4          "host": "yoonkeunsoo.cafe24.com",
5          "port": 21,
6          "type": "ftp",
7          "username": "yoonkeunsoo",
8          "password": "guest0505",
9          "path": "/",
10         "autosave": true,
11         "confirm": false
12     }
13 ]
```

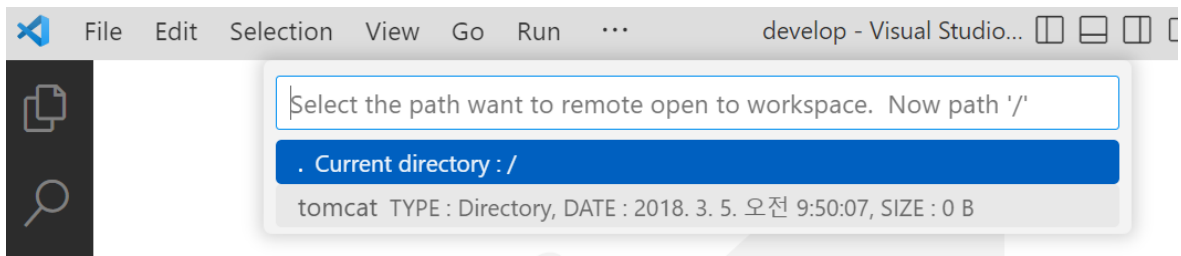
설정은 한번만 하면 앞으로는 계속 사용만 한다.

F1 키를 누른후 ftp 를 치면 목록에서 Remote directory open to workspace 를 선택한다.

다음 부터는 F1키만 누르면 항상 상단에서 고를 수 있다.

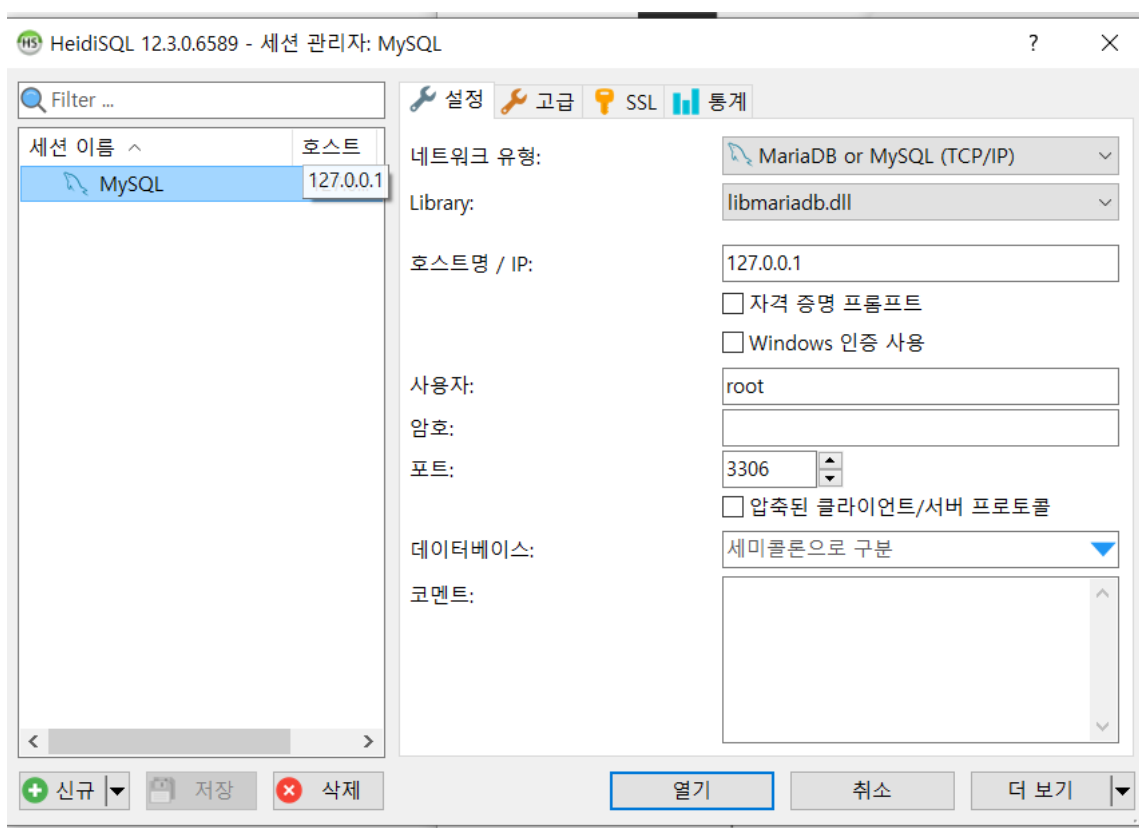


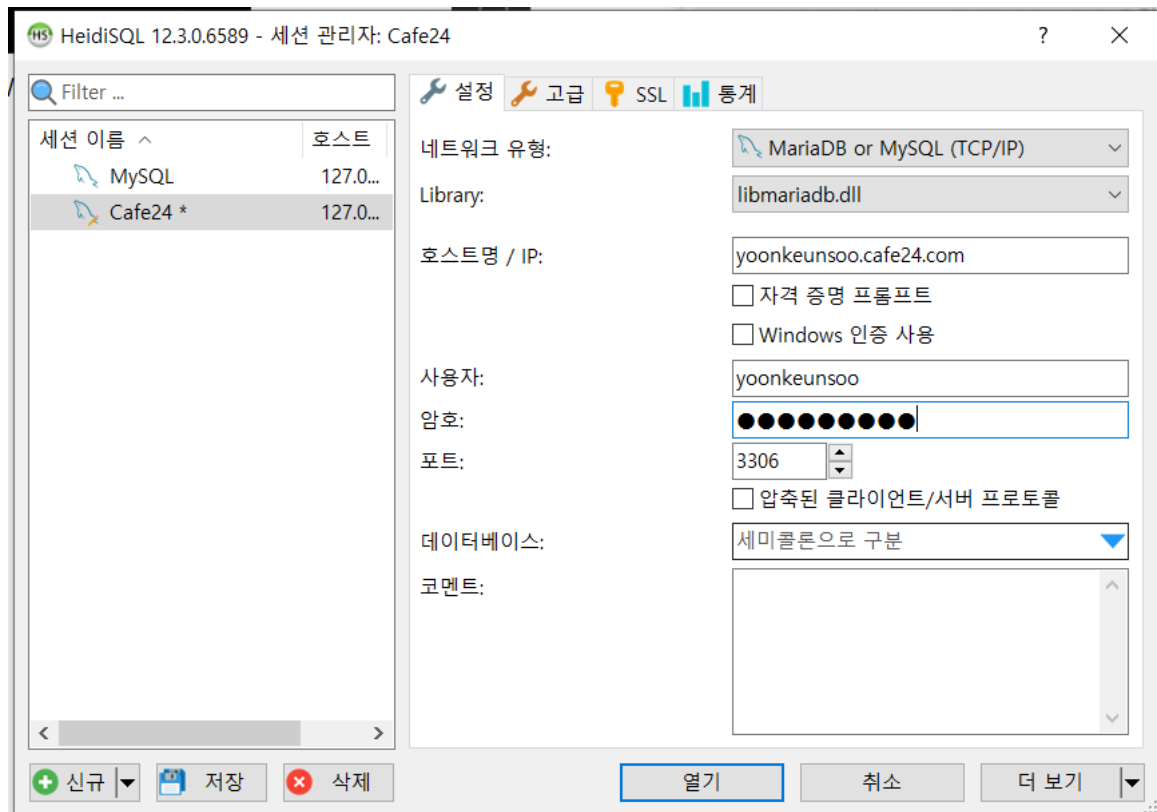
아래에서 tomcat 클릭



heidi (new user for cafe24)

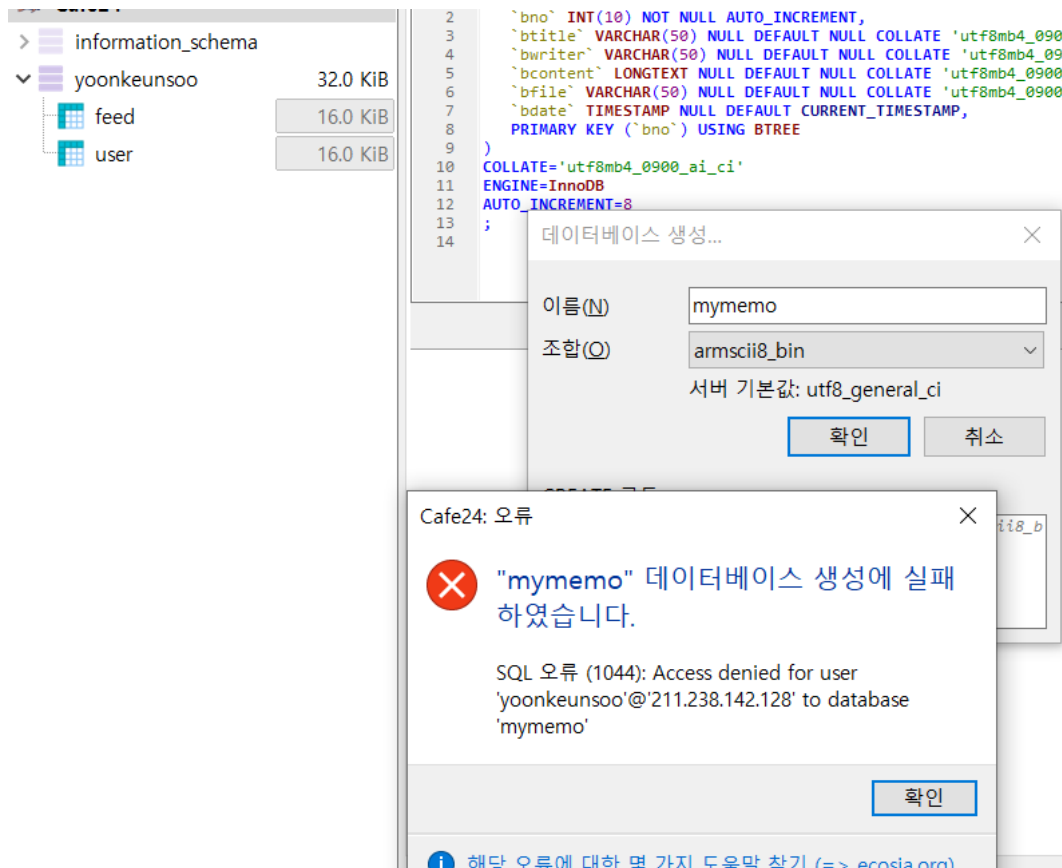
heidi 설정 시 중요한 건꼭 cafe24 사이트 내에서 db ip설정이 제대로 되어 있어야 한다.





```
CREATE TABLE user (
  id VARCHAR(128) NOT NULL,
  password VARCHAR(32) NULL DEFAULT NULL,
  name VARCHAR(32) NULL DEFAULT NULL,
  ts TIMESTAMP NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (id) USING BTREE
)
ENGINE=InnoDB
;
```

```
CREATE TABLE feed (
  no INT(10) NOT NULL AUTO_INCREMENT,
  id VARCHAR(50) NULL DEFAULT NULL,
  content LONGTEXT NULL DEFAULT NULL,
  ts TIMESTAMP NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (no) USING BTREE
)
ENGINE=InnoDB
;
```



테이블은 마음대로 만들었지만 cafe24의 정책 상 새로운 디비를 생성할수는 없다.

자신의 id 명으로만 db를 사용할수 있다.

따라서 지금까지 사용한 디비명은 사용할 수 없고 cafe24 의 아이디 명으로만 db를 쓸수 있다.

또, 지금까지는 root 사용자로 접근하였는데 역시 cafe24의 정책상 반드시 cafe24 아이디와 비번으로만 접근할 수 있다.

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>

<!-- Step 1 import SQL Packages -->
<%@ page import="java.sql.*" %>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<%
//Step 2 load JDBC Driver
try {
    Class.forName("com.mysql.jdbc.Driver");
} catch (ClassNotFoundException err) {
    out.print("JDBC Driver loading error<br>" + err.getMessage());
}

// Step 3 create Connection Object

Connection conn = null;

try {
    conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/yoonkeunsoo", "yoonkeunsoo", "guest0505");
} catch (SQLException err) {
    out.print("Connection Object error<br>" + err.getMessage());
}

// Step 4 create Statement Object
```



```
String sql = "CREATE TABLE student("
    + "hakbun varchar(10),"
    + "name varchar(10),"
    + "dept varchar(20),"
    + "addr varchar(30),"
    + "primary key(hakbun))";

PreparedStatement pstmt = conn.prepareStatement(sql);

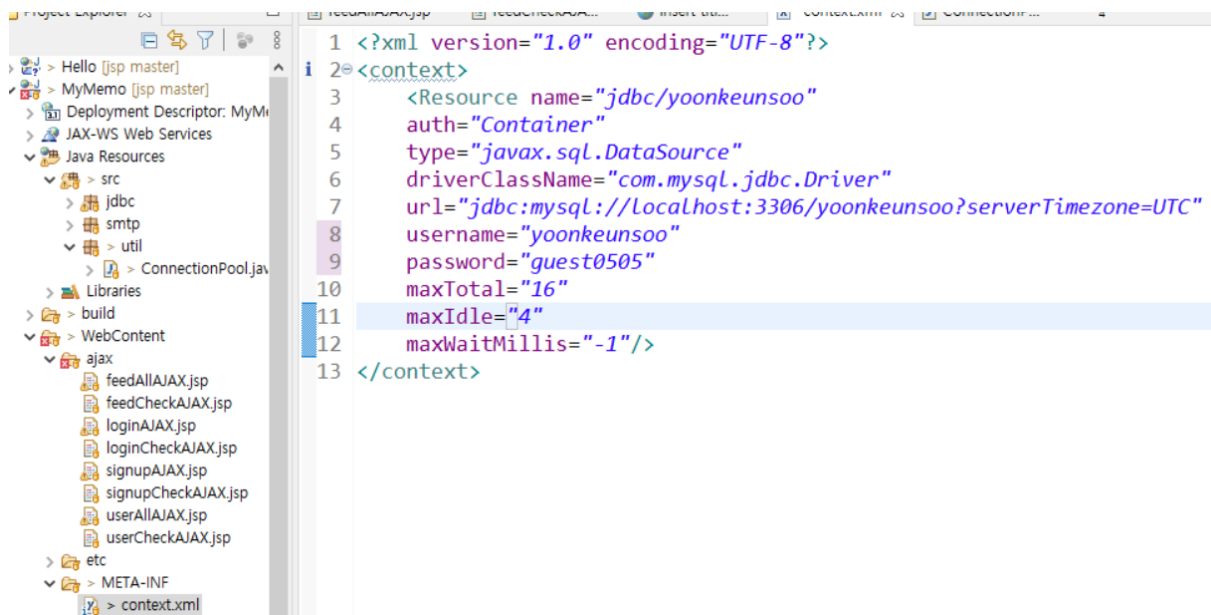
// Step 5 excute SQL Query

pstmt.executeUpdate();

// Step 6 close Connection

pstmt.close();
conn.close();

%>
</body>
</html>
```



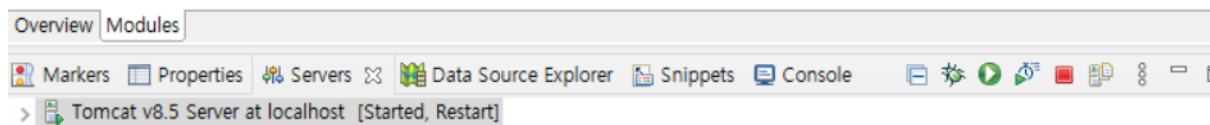
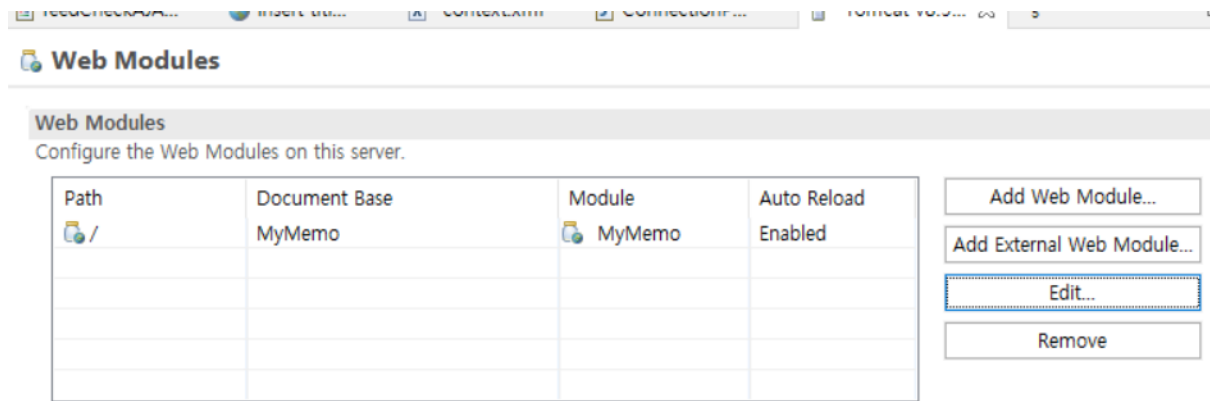
실제 프로젝트를 올리기 위해 사전 점검이 필요하다.

context.xml 디비 정보 수정



ConnectionPool에 디비명 수정

루트 디렉토리 설정



db update

서버에 설치된 DB 이름은 직접 만들 수 없고 변경할 수 없으므로, 카페24 에서 제공하는 DB 명(ID명과 동일)에 맞춘다.

마리아 DB드라이버는 자바 1.8 버전과 호환이 되는 **mariadb-java-client-1.8.0.jar**를 사용하여야 오류가 나지 않는다 (2.X 버전 이용시 자바 1.8과 호환 안됨)

DB.jdbc. connectionpool

test

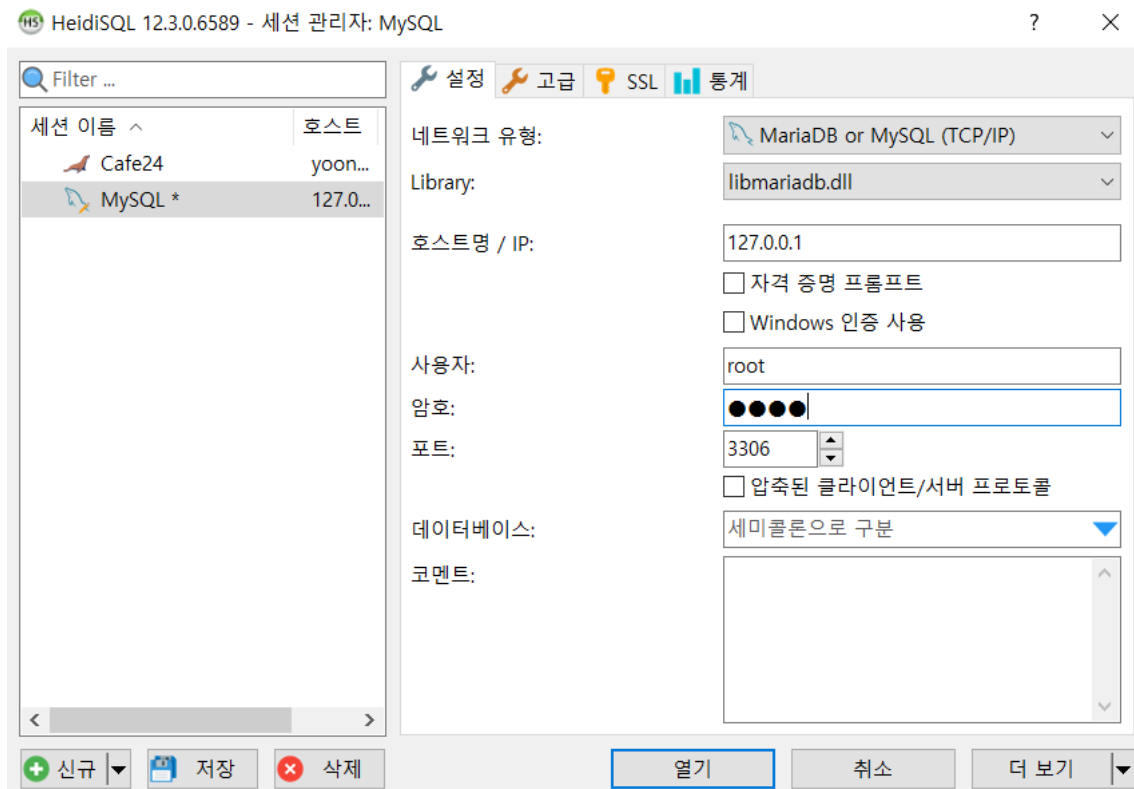
Adv

DB Setting. new db user

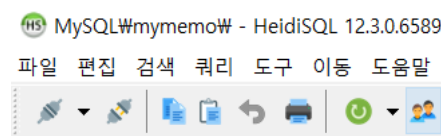
ROOT war

로컬 디비환경과 카페24 db환경이 달라서 로컬에서 테스트가 힘든 상태이다.

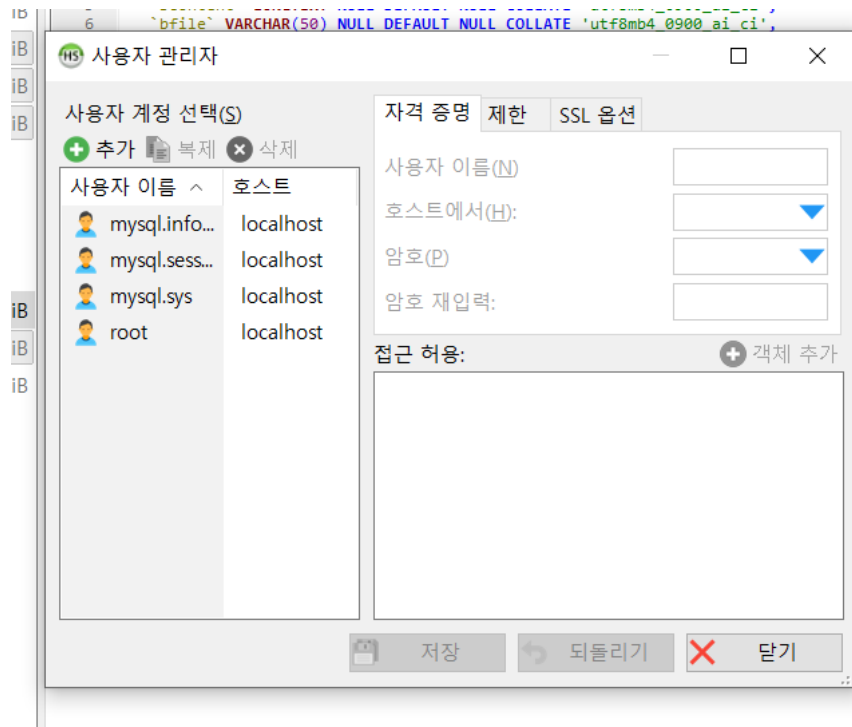
따라서 로컬에 카페24 와 동일하게 사용자 1명을 만들어 권한을 풀로 주면 동일하게 로컬 작업하고 그대로 업로드 하여 돌릴 수 있다.

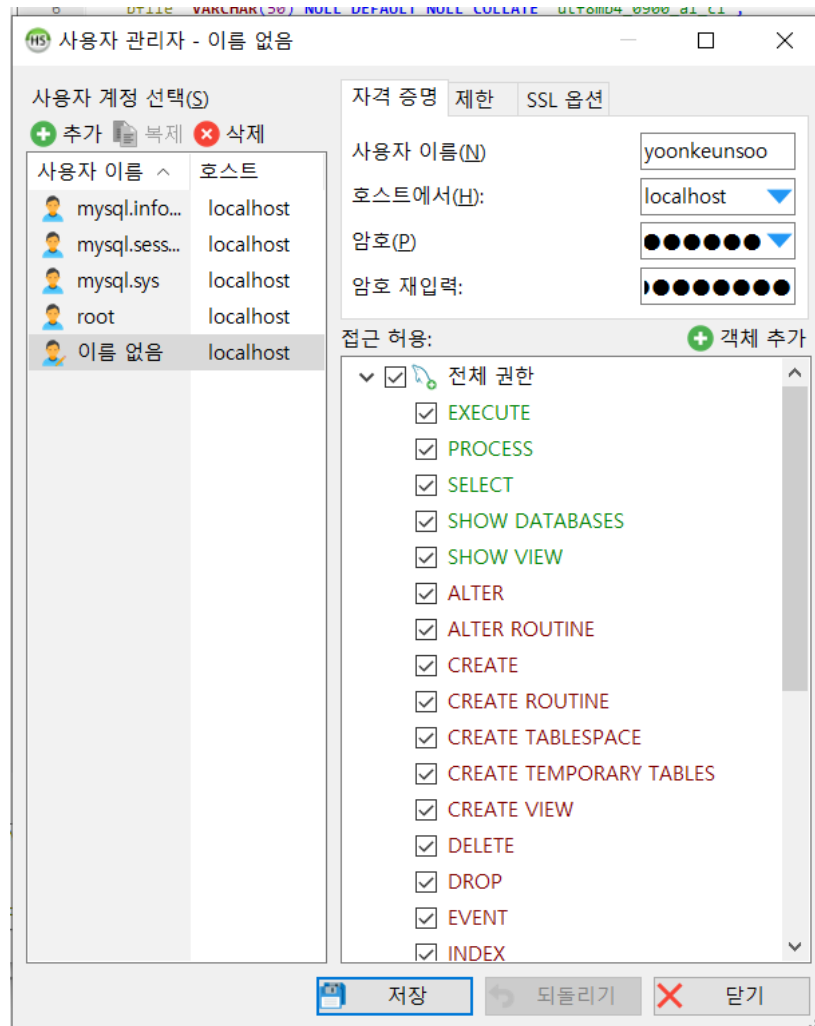


먼저 로컬에 기존 아이디인 root로 들어간다.

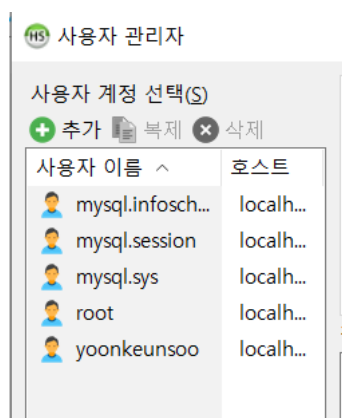


화명 상단의 사람 그림 '사용자 인증 및 권한 관리'

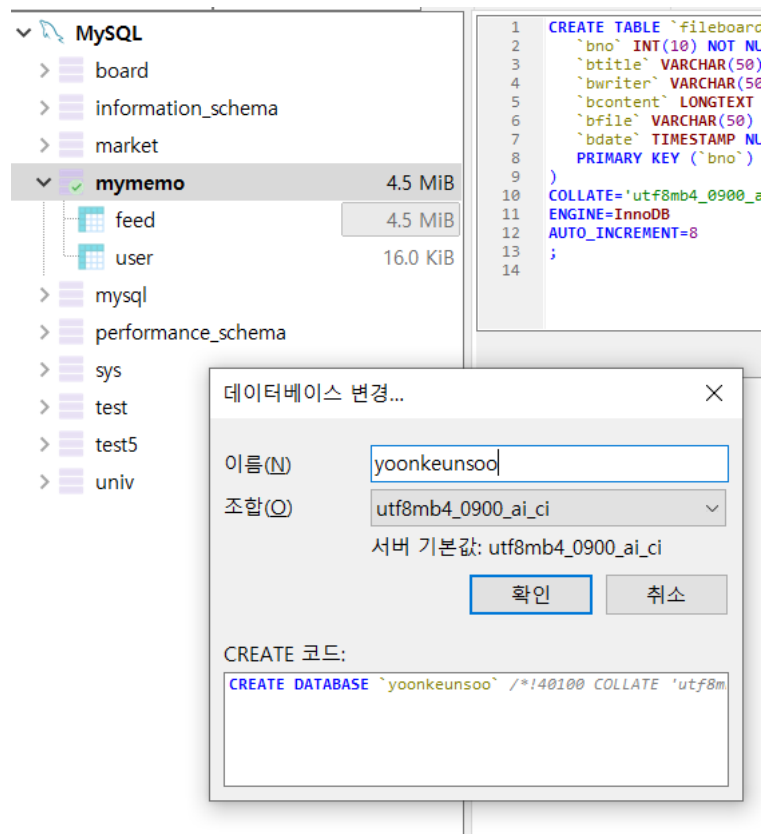




카페24와 동일한 아이디 동일한 패스워드를 넣고 전체 권한을 부여한다.



생성완료



마지막으로 카페24의 디비명과 동일한 환경을 갖기 위해 변경한다.

- DB 교체 MySQL → mariaDB
- java 교체 11 → 8
- connectionpool 확인