

테이블의 중복 제거 검색

```
mysql> desc customer;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
name	varchar(11)	YES		NULL	
city	varchar(11)	YES		NULL	
country	varchar(11)	YES		NULL	

```
4 rows in set (0.01 sec)
```

```
mysql> select * from customer;
```

id	name	city	country
1	알프레드	베를린	독일
2	안나	멕시코	멕시코
3	안토니오	멕시코	멕시코
4	토마스	런던	영국
5	크리스티나	스톡홀름	스웨덴
6	김봉삼	서울	한국

```
6 rows in set (0.01 sec)
```

```
mysql>
```

다음과 같이 테이블을 작성합니다.

테이블의 중복 제거 검색

Select distinct [필드명] from [테이블명];

```
mysql> select distinct country from customer;
```

country
독일
멕시코
영국
스웨덴
한국

5 rows in set (0.00 sec)

```
mysql> select * from customer;
```

id	name	city	country
1	알프레드	베를린	독일
2	안나	멕시코	멕시코
3	안토니오	멕시코	멕시코
4	토마스	런던	영국
5	크리스티나	스톡홀름	스웨덴
6	김봉삼	서울	한국

6 rows in set (0.01 sec)

```
mysql> _
```

“멕시코” 2개 중에 중복된 값이 제거되어 나옴을 확인

Null 속성

데이터가 아무것도 들어있지 않은 것을 의미

```
mysql> desc customer;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id    | int(11)       | NO   | PRI | NULL    | auto_increment |
| name  | varchar(11)   | YES  |     | NULL    |                |
| city  | varchar(11)   | YES  |     | NULL    |                |
| country | varchar(11)  | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
```

4 rows in set (0.00 sec)

```
mysql> insert into customer (name) values ("백구");
Query OK, 1 row affected (0.02 sec)
```

```
mysql> select * from customer;
```

id	name	city	country
1	알프레드	베를린	독일
2	안나	멕시코	멕시코
3	안토니오	멕시코	멕시코
4	토마스	런던	영국
5	크리스티나	스톡홀름	스웨덴
6	김봉삼	서울	한국
7	백구	NULL	NULL

7 rows in set (0.00 sec)

아무것도 값을 넣지 않으면
Default에 적혀진 값이 들어가며
Default가 null이므로
Null값이 적혀들어감

Null 속성

데이터가 아무것도 들어있지 않은 것을 의미

```
mysql> insert into customer(  
  -> name, country) values (  
  -> "이영준", " ");  
Query OK, 1 row affected (0.08 sec)
```

```
mysql> select * from customer;
```

id	name	city	country
1	알프레드	베를린	독일
2	안나	멕시코	멕시코
3	안토니오	멕시코	멕시코
4	토마스	런던	영국
5	크리스티나	스톡홀름	스웨덴
6	김봉삼	서울	한국
7	백구	NULL	NULL
8	이영준	NULL	

```
8 rows in set (0.00 sec)
```

“역시도 빈칸이라는
특정한 값을 가지므로
Null이 아님

Select된 결과물을 융합

Select [필드명] from [테이블명]
union

Select [필드명] from [테이블명];

```
mysql> select * from shop union select * from pro;
```

num	product	qty	sell
1	tv	5	900000
2	냉장고	2	3000000
3	컴퓨터	9	1500000
4	자전거	5	200000
5	마우스	90	10000
6	키보드	60	5000
7	에어컨	3	1800000
2713	자전거	60000	일본
15112	tv	450000	인도
15114	마우스	2000	베트남
15115	키보드	1000	중국
25112	냉장고	1500000	미국
25113	컴퓨터	750000	대한민국
50115	에어컨	900000	러시아

```
14 rows in set (0.01 sec)
```

Select된 결과물을 융합

```
mysql> select product from shop union select * from pro;
```

```
ERROR 1222 (21000): The used SELECT statements have a different number of columns
```

Select된 결과값이 같아야지만 union이 가능

그룹만들기

Select [필드명] from [테이블명]
group by [필드명];

```
mysql> select * from customer group by country;
```

id	name	city	country
1	알프레드	베를린	독일
2	안나	멕시코	멕시코
4	토마스	런던	영국
5	크리스티나	스톡홀름	스웨덴
6	김봉삼	서울	한국
7	백구	NULL	NULL
8	이영준	NULL	

```
7 rows in set (0.00 sec)
```

3 안토니오 멕시코 멕시코 가 하나의 그룹이 되어 보이지 않음

테이블을 생성

```
mysql> desc zodiac;
```

Field	Type	Null	Key	Default	Extra
num	int(11)	NO	PRI	NULL	auto_increment
chara	varchar(11)	YES		NULL	
name	varchar(11)	YES		NULL	
race	varchar(11)	YES		NULL	
eat	varchar(11)	YES		NULL	

```
5 rows in set (0.01 sec)
```

```
mysql> select * from zodiac;
```

num	chara	name	race	eat
1	자	쥐	설치류	잡식
2	축	소	포유류	초식
3	인	호랑이	포유류	육식
4	묘	토끼	포유류	초식
5	진	용	신화형류	불명
6	사	뱀	파충류	육식
7	오	말	포유류	초식
8	미	양	포유류	초식
9	신	원숭이	포유류	잡식
10	유	달	조류	잡식
11	술	개	포유류	잡식
12	해	돼지	포유류	잡식

```
12 rows in set (0.00 sec)
```


그룹에 조건 주기(그룹만들기)

Select [필드명] from [테이블명]
group by [필드명];

```
mysql> select eat, count(eat) from zodiac group by eat;
```

eat	count(eat)
잡식	5
초식	4
육식	2
불명	1

```
4 rows in set (0.00 sec)
```

그룹에 조건 주기(조건주기)

Select [필드명] from [테이블명]
group by [필드명] having [조건];

```
mysql> select eat, count(eat) from zodiac group by eat  
-> having count(eat)>2;
```

eat	count(eat)
잡식	5
초식	4

```
2 rows in set (0.00 sec)
```

그룹에 조건 주기(나열기준)

Select [필드명] from [테이블명]
group by [필드명] having [조건]
order by [필드명] [기준];

```
mysql> select eat, count(eat) from zodiac group by eat  
-> having count(eat)>2 order by count(eat) asc;
```

eat	count(eat)
초식	4
잡식	5

2 rows in set (0.00 sec)

실습 (테이블 생성)

```
mysql> desc eng_zod;
```

Field	Type	Null	Key	Default	Extra
num	int(11)	NO	PRI	NULL	auto_increment
chara	varchar(11)	YES		NULL	
name	varchar(11)	YES		NULL	

```
3 rows in set (0.01 sec)
```

```
mysql> select *from eng_zod;
```

num	chara	name
1	백양궁	양자리
2	금우궁	황소자리
3	쌍자궁	쌍둥이자리
4	거해궁	게자리
5	사자궁	사자자리
6	처녀궁	처녀자리
7	천칭궁	천칭자리
8	천갈궁	전갈자리
9	인마궁	사수자리
10	마갈궁	염소자리
11	보병궁	물병자리
12	쌍어궁	물고기자리
13	사견궁	양꼬리자리

```
13 rows in set (0.00 sec)
```

실습

월	12지	별자리
1	쥐	양자리
2	소	황소자리
3	호랑이	쌍둥이자리
4	토끼	게자리
5	용	사자자리
6	뱀	처녀자리
7	말	천칭자리
8	양	전갈자리
9	원숭이	사수자리
10	닭	염소자리
11	개	물병자리
12	돼지	물고기자리

12 rows in set (0.00 sec)

Select 문을 한번만 사용해서 다음과 같은 결과를 만드시오

실습2

```
+-----+-----+-----+
| 별자리명 | 12지   | 식성   |
+-----+-----+-----+
| 백양궁   | 쥐     | 잡식   |
| 금우궁   | 소     | 초식   |
| 쌍자궁   | 호랑이 | 육식   |
| 사자궁   | 용     | 불명   |
+-----+-----+-----+
4 rows in set (0.00 sec)
```

식성을 기준으로 대표되는 별자리이름과 12지를 검색

```
+-----+-----+-----+
| 별자리명 | 12지   | 식성   |
+-----+-----+-----+
| 사자궁   | 용     | 불명   |
+-----+-----+-----+
1 row in set (0.00 sec)
```

이들 중 식성이 불명인 조건을 추가

검색 결과가 존재할 경우만 표시

Select [필드명] from [테이블명]
where exist (조건);

```
mysql> select * from zodiac where exists (select num from zodiac where num=13);  
Empty set (0.00 sec)
```

Zodiac 테이블에는 13번이 존재하지 않아서 검색결과없음

```
mysql> select * from zodiac where exists (select num from eng_zod where num=13);
```

num	chara	name	race	eat
1	자축	쥐	설치류	잡식
2	축인	소	포유류	초식
3	인묘	호랑이	포유류	육식
4	묘진	토끼	포유류	초식
5	진사	용	신화	불명
6	사오	뱀	파충류	육식
7	오미	말	포유류	초식
8	미신	양	포유류	초식
9	신유	원숭이	포유류	잡식
10	유술	개	포유류	잡식
11	술해	돼지	포유류	잡식
12	해		포유류	잡식

12 rows in set (0.00 sec)

조건이 결과와는 무관하게 앞의 select의 값만이 실행됨

필드 속성(Not null)

Null값이 입력될 수 없는 속성을 의미

```
mysql> desc zodiac
-> ;
```

Field	Type	Null	Key	Default	Extra
num	int(11)	NO	PRI	NULL	auto_increment
chara	varchar(11)	YES		NULL	
name	varchar(11)	YES		NULL	
race	varchar(11)	YES		NULL	
eat	varchar(11)	YES		NULL	

```
5 rows in set (0.00 sec)
```

```
mysql> insert into zodiac (chara,name,race) values ("웅","곰","포유류");
Query OK, 1 row affected (0.08 sec)
```

Eat 필드에 아무것도 입력하지 않아도 값이 들어가는 것을 확인

```
mysql> select * from zodiac where name="곰";
```

num	chara	name	race	eat
13	웅	곰	포유류	NULL

```
1 row in set (0.00 sec)
```

Null이 입력된 것을 확인

필드 속성(Not null)

들어간 값을 지우고 eat 필드에 not null속성을 추가

```
mysql> delete from zodiac where name="곰";  
Query OK, 1 row affected (0.01 sec)
```

```
mysql> alter table zodiac modify eat varchar(11) not null;  
Query OK, 0 rows affected (0.20 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> desc zodiac;
```

Field	Type	Null	Key	Default	Extra
num	int(11)	NO	PRI	NULL	auto_increment
chara	varchar(11)	YES		NULL	
name	varchar(11)	YES		NULL	
race	varchar(11)	YES		NULL	
eat	varchar(11)	NO		NULL	

5 rows in set (0.01 sec)

필드 속성(Not null)

들어간 값을 지우고 eat 필드에 not null속성을 추가

```
mysql> insert into zodiac (chara,name,race) values ("웅","곰","포유류");  
ERROR 1364 (HY000): Field 'eat' doesn't have a default value
```

```
mysql> desc zodiac;
```

Field	Type	Null	Key	Default	Extra
num	int(11)	NO	PRI	NULL	auto_increment
chara	varchar(11)	YES		NULL	
name	varchar(11)	YES		NULL	
race	varchar(11)	YES		NULL	
eat	varchar(11)	NO		NULL	

5 rows in set (0.01 sec)

Null이 입력될 수 없도록 되어 있어 정보가 입력되지 않음

필드 속성(default)

```
mysql> alter table zodiac modify eat varchar(11) not null default "잡식";
Query OK, 0 rows affected (0.09 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

```
mysql> desc zodiac;
```

Field	Type	Null	Key	Default	Extra
num	int(11)	NO	PRI	NULL	auto_increment
chara	varchar(11)	YES		NULL	
name	varchar(11)	YES		NULL	
race	varchar(11)	YES		NULL	
eat	varchar(11)	NO		잡식	

```
5 rows in set (0.00 sec)
```

값을 입력하지 않아도 당연히 해당 필드에 잡식이 입력됨

필드 속성(default)

```
mysql> alter table zodiac modify eat varchar(11) not null default "잡식";
Query OK, 0 rows affected (0.09 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> desc zodiac;
```

Field	Type	Null	Key	Default	Extra
num	int(11)	NO	PRI	NULL	auto_increment
chara	varchar(11)	YES		NULL	
name	varchar(11)	YES		NULL	
race	varchar(11)	YES		NULL	
eat	varchar(11)	NO		잡식	

```
5 rows in set (0.00 sec)
```

값을 입력하지 않아도 당연히 해당 필드에 잡식이 입력됨

필드 속성(default)

```
mysql> insert into zodiac (chara,name,race) values ("웅","곰","포유류");  
Query OK, 1 row affected (0.05 sec)
```

```
mysql> select * from zodiac;
```

num	chara	name	race	eat
1	자	쥐	설치류	잡식
2	축	소	포유류	초식
3	인	호랑이	포유류	육식
4	묘	토끼	포유류	초식
5	진	용	중형	불명
6	사	뱀	파충류	육식
7	오	말	포유류	초식
8	미	양	포유류	초식
9	신	원숭이	포유류	잡식
10	유	갈래	조류	잡식
11	술	개	포유류	잡식
12	해	돼지	포유류	잡식
14	웅	곰	포유류	잡식

```
13 rows in set (0.00 sec)
```

값을 지정하지 않아도 자동으로 default값인 잡식이 들어가며
Not null속성이 있다고 하여도 잘 입력됨을 확인

필드 속성(unique)

```
mysql> alter table zodiac modify name varchar(11) unique;  
Query OK, 0 rows affected (0.08 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> desc zodiac;
```

Field	Type	Null	Key	Default	Extra
num	int(11)	NO	PRI	NULL	auto_increment
chara	varchar(11)	YES		NULL	
name	varchar(11)	YES	UNI	NULL	
race	varchar(11)	YES		NULL	
eat	varchar(11)	NO		잡식	

5 rows in set (0.00 sec)

Name 필드가 unique로 지정되어 같은 값이 들어갈 수 없도록 되었음

필드 속성(unique)

```
mysql> insert into zodiac (chara,name,race) values ("웅","곰","포유류");
ERROR 1062 (23000): Duplicate entry '곰' for key 'name'
```

중복되는 값이 있어 값을 추가할 수 없게 변경됨

```
mysql> select * from zodiac;
```

num	chara	name	race	eat
1	자	취	설치류	잡식
2	축	소	포유류	초식
3	인	호랑이	포유류	육식
4	묘	토끼	포유류	초식
5	진	용	중생물	명식
6	사	뱀	파충류	육식
7	오	말	포유류	초식
8	미	양	포유류	초식
9	신	원숭이	포유류	잡식
10	유	달	조류	잡식
11	술	개	포유류	잡식
12	해	돼지	포유류	잡식
13	웅	곰	포유류	잡식

13 rows in set (0.00 sec)

필드 속성(primary key)

```
mysql> desc zodiac;
```

Field	Type	Null	Key	Default	Extra
num	int(11)	NO	PRI	NULL	auto_increment
chara	varchar(11)	YES		NULL	
name	varchar(11)	YES	UNI	NULL	
race	varchar(11)	YES		NULL	
eat	varchar(11)	NO		잡식	

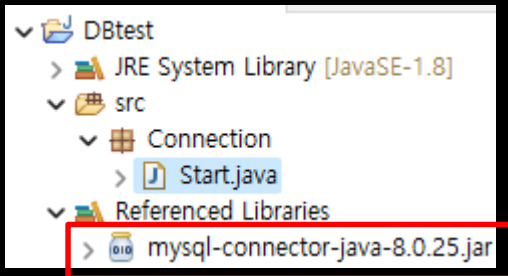
```
5 rows in set (0.00 sec)
```

Primary key는 해당 테이블에서 데이터를 구별해주는 고유한 키

기본적으로 unique 속성과 not null 속성을 가지고 있어서
중복될 수 없고 공백으로 둘 수 없다.

테이블은 반드시 하나의 primary key를 두어야 한다

자바를 통한 접속



```
1 package Connection;
2
3 import java.sql.*;
4
5 public class Start {
6     public static void main(String[] args) {
7         Connection conn=null;
8         Statement stmt=null;
9         ResultSet rs=null;
10        try {
11            Class.forName("com.mysql.cj.jdbc.Driver");
12            String url="jdbc:mysql://localhost:3306/sample";
13            conn=DriverManager.getConnection(url,"root","iotiot");
14
15            stmt=conn.createStatement();
16            String sql="select * from customer";
17            rs=stmt.executeQuery(sql);
18
19            while(rs.next()) {
20                String data1=rs.getString(1);
21                String data2=rs.getString(2);
22                String data3=rs.getString(3);
23                System.out.println(data1+"\t"+data2+"\t"+data3);
24            }
25        } catch (Exception e) {
26            System.out.println(e);
27        } finally {
28            try {
29                if(conn!=null) {conn.close();}
30            } catch (Exception ex) {
31                System.out.println(ex);
32            }
33        }
34    }
35 }
```

메모달력

달력 메모

↓ ← 2018년 7월 → ↑

일	월	화	수	목	금	토
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

메모추가 메모삭제

이상이 있습니까?

```
mysql> select * from memo;
```

```
+-----+-----+-----+-----+  
| memo          | year | month | day |  
+-----+-----+-----+-----+  
| 이상이 있습니까? | 2018 | 7     | 18  |  
+-----+-----+-----+-----+  
1 row in set (0.01 sec)
```

데이터베이스에 메모를 남겨 Java Swing을 통해 값을 추가하고 빼는 기능
메모가 작성된 날짜는 색상을 바꾸어 표시한다
데이터베이스를 이용하여 프로그램을 종료하여도 메모가 남아 있다
콘솔을 통하여 쿼리문이 잘 작동하는지 확인할 수 있도록 한다.

메모달력 구조 잡기



- 1 JFrame의 제목
- 2 기본 종료 옵션
- 3 North패널 버튼x4 필드x2
- 4 버튼 공간 7X7
- 5 메모 입력용 텍스트필드

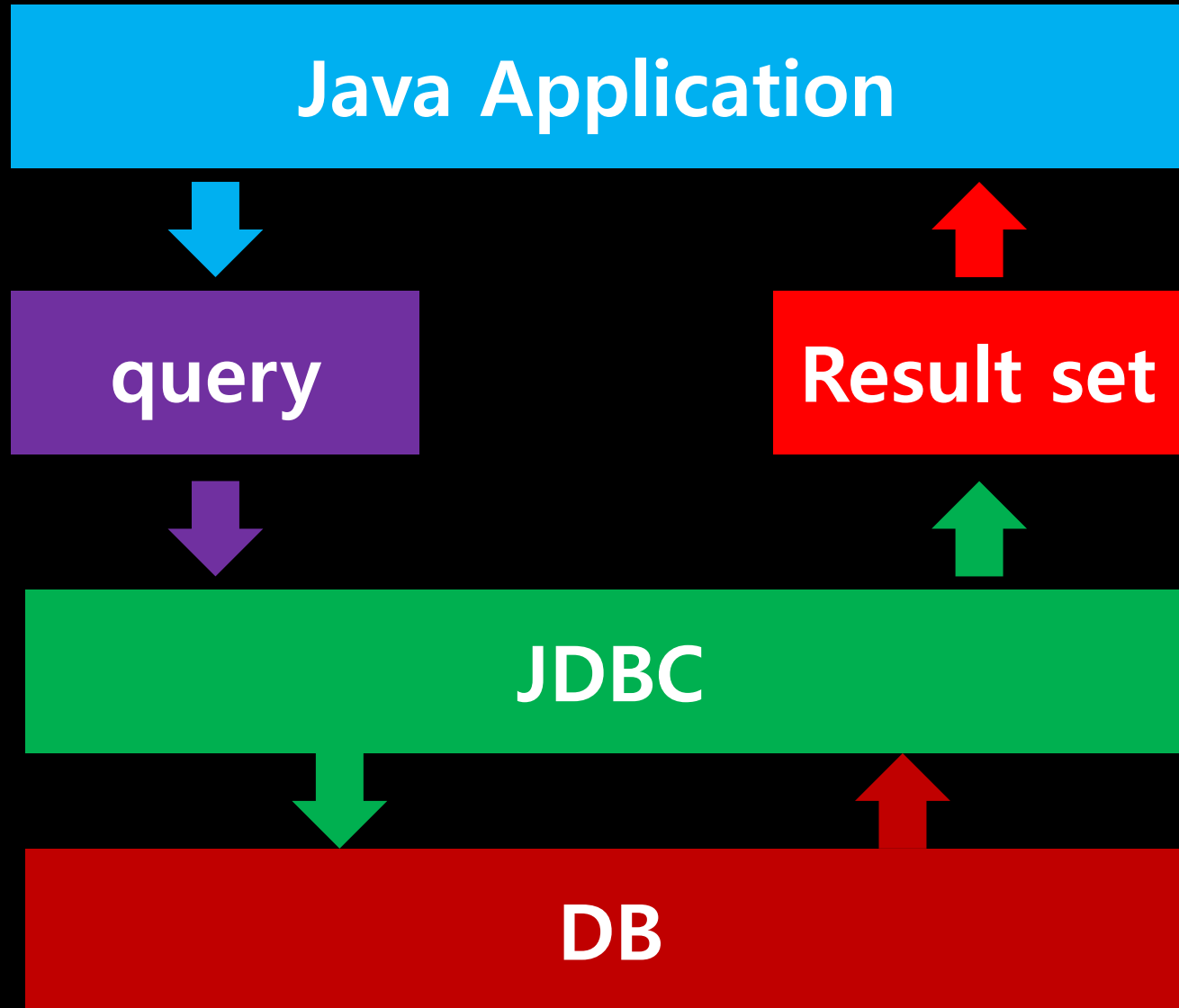
메모달력 데이터베이스

```
mysql> desc memo;
```

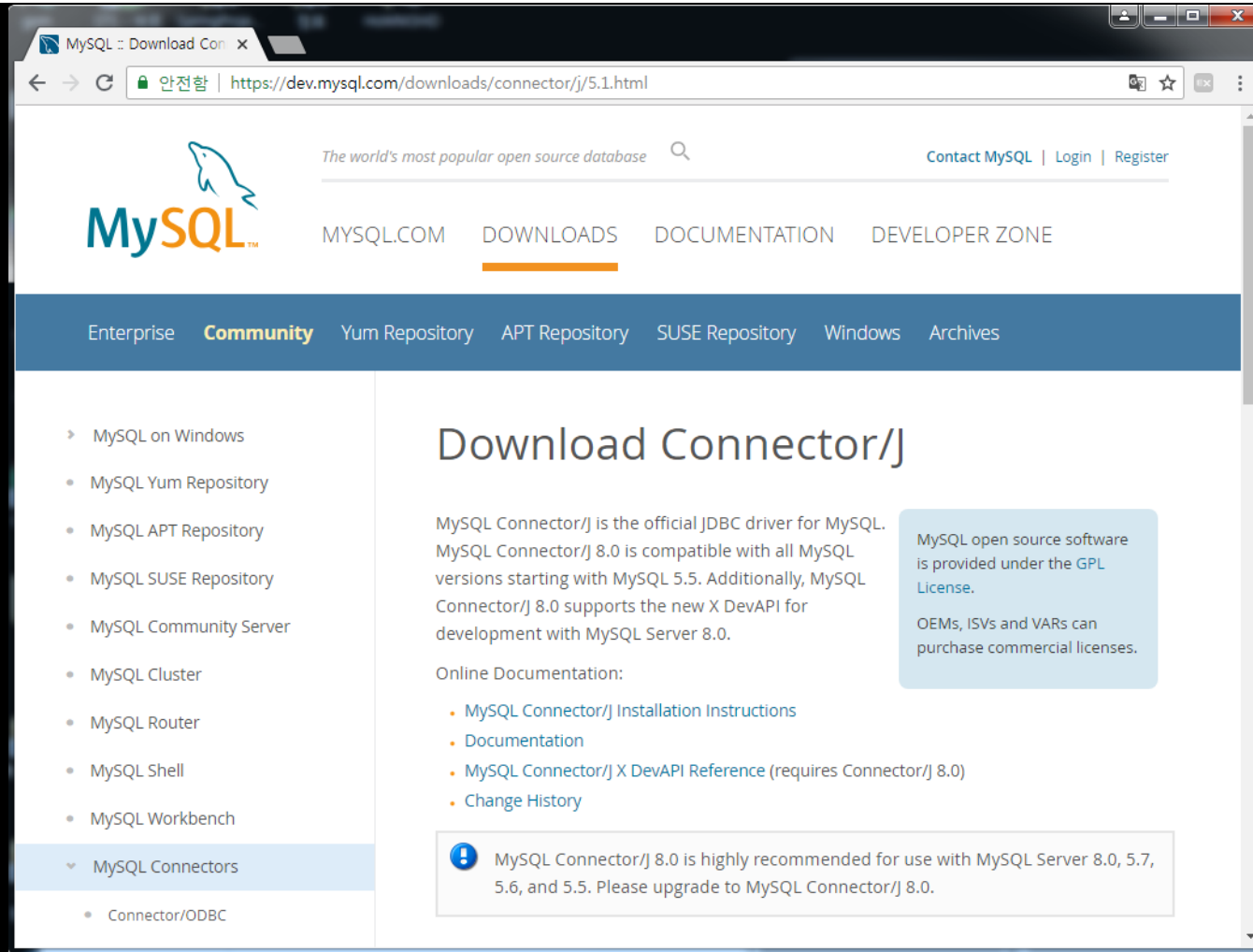
Field	Type	Null	Key	Default	Extra
memo	text	YES		NULL	
year	varchar(10)	YES		NULL	
month	varchar(10)	YES		NULL	
day	varchar(10)	YES		NULL	

```
4 rows in set (0.01 sec)
```

DB와 Java의 연동



드라이버의 설치(1)



<https://dev.mysql.com/downloads/connector/j/5.1.html>

드라이버의 설치(3)

MySQL :: Download Connector/J

← → ↻ | [안전함](#) | <https://dev.mysql.com/downloads/connector/j/5.1.html>

- Connector/Python
- Connector/C++
- Connector/C (libmysqlclient)
- MySQL Native Driver for PHP
- Other Downloads

Generally Available (GA) Releases

Connector/J 5.1.46

Select Operating System: [Looking for the latest GA version?](#)

Platform Independent

Platform Independent (Architecture Independent), Compressed TAR Archive (mysql-connector-java-5.1.46.tar.gz)	5.1.46	4.2M	Download
Platform Independent (Architecture Independent), ZIP Archive (mysql-connector-java-5.1.46.zip)	5.1.46	4.6M	Download

MD5: 6419f5a73c76991d73e27a93babca263 | [Signature](#)

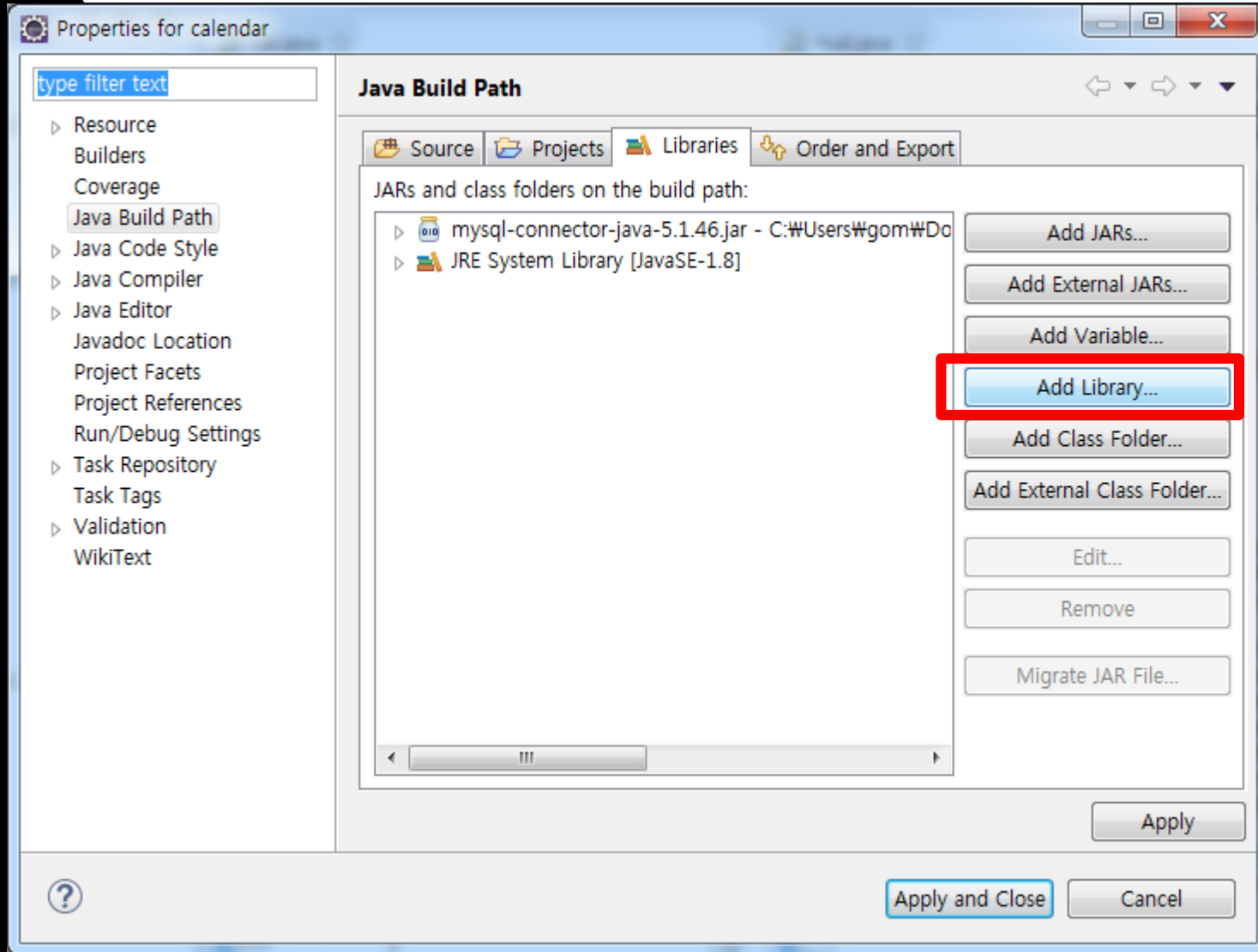
MD5: d1104e22634e1645db418c16d5ff7631 | [Signature](#)

! We suggest that you use the [MD5 checksums](#) and [GnuPG signatures](#) to verify the integrity of the packages you download.

Contact MySQL Sales
<https://dev.mysql.com/downloads/file/?id=476198> Countries »

f t in g+ You Tube

드라이버의 설치(3)



압축을 푼 경로를 지정하고 mysql-connector-java-5.1.46.jar를 선택