

## 문제해결 시나리오 평가 문제지

훈련과정	[디지털컨버전스] 공공데이터를 융합한 자바스프링&부트 웹개발자 과정 B		
교과목	02. 데이터베이스 구현		
능력단위	데이터베이스 구현	출제자	조성현 (인)
평가일시	2023.03.20. (5, 6 교시)	성명	박유리
배점기준	각 문제 난이도별 개별 점수	점수	

※ 아래에 제시된 지문에 따라 실행하고 결과를 도출하시오.

1. Mysql 8.0 버전을 설치된 GUI 환경을 Capture 하시오. (20점)

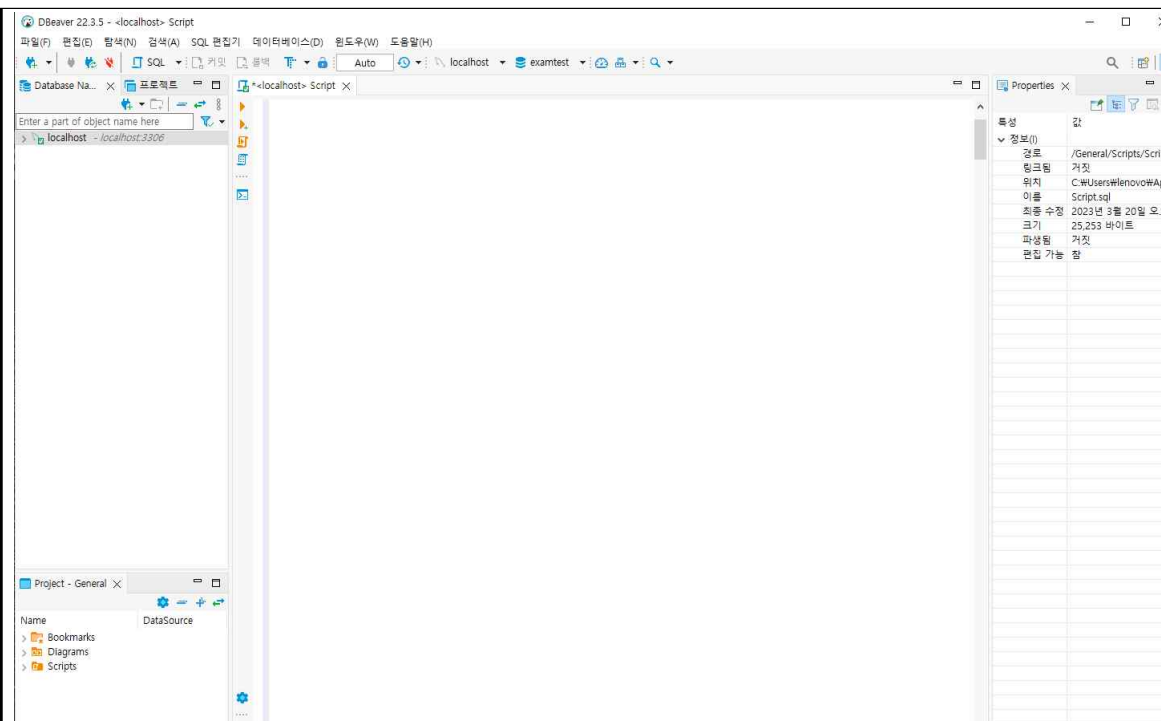
① WIN10 또는 Win11 운영체제 환경에서 MySQL8.0 실행을 위한 환경변수 PATH 설정 (5점)

cmd명령 프롬프트

Microsoft Windows [Version 10.0.19045.2728]  
(c) Microsoft Corporation. All rights reserved.  
C:\Users\lenovo>SETX PATH "C:\Program Files\MySQL\MySQL Server 8.0\bin"

성공: 지정한 값을 저장했습니다.  
C:\Users\lenovo>

② Client 프로그램( DBeaver 또는 MySQL workbench 8.0 CE)을 이용한 연결 (10점)



### ③ MySQL 8.0 command Line Client 또는 CMD 창을 활용한 MySQL 연결 (5점)

**명령 프롬프트 - mysql -u root -p**

Microsoft Windows [Version 10.0.19045.2728]  
(c) Microsoft Corporation. All rights reserved.

```
D:\Users\lenovo>mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 17
Server version: 8.0.32 MySQL Community Server - GPL
```

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

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> █
```

2. DDL문을 활용한 데이터베이스를 구축하는 SQL구문을 작성하고 결과를 도출(Capture) 하시오. (30점)

- 데이터베이스의 이름은 examtest 이다.

- members , orders의 2개의 테이블이 존재하며 테이블은 아래와 내용과 같은 구조이다.

회원테이블 : members

한글명	영문명	데이터 타입	길이	PK / FK 여부	NULL 허용	기본값
아이디	user_id	문자	8	PK		
이름	name	문자	10			
생일	birth_year	숫자	INT			
주소	addr	문자	2			
핸드폰1	mobile_brand	문자	3		허용	
핸드폰2	mobile_number	문자	8		허용	
키	user_height	숫자	SMALLINT		허용	
등록일	reg_date	날짜	DATE		허용	

구매테이블 : orders

주문번호	num	숫자	INT	PK		자동증가
주문자아이디	order_id	문자	8	FK		
상품명	prod_name	문자	6			
상품종류	category	문자	4		허용	
가격	price	숫자	INT			
수량	quantity	숫자	SMALLINT			

입력데이터 : members

KHK	강효경	1973	서울	010	12345678	176	2013-05-05
PYR	박유리	1972	경북	011	33445689	174	2014-03-03
KTY	김태영	1965	경기	018	42589124	172	2010-10-10
JYP	조용필	1950	경기	011	55981247	166	2009-04-04
KJJ	강진주	1979	경남	011	36458974	173	2012-04-04
KDK	김도겸	1971	전남	019	56894137	177	2007-07-07
LJB	임재범	1963	서울	016	25697874	182	2009-09-09
WMH	원무현	1987	서울	011	25254671	182	2008-08-08
SYW	손영우	1979	서울			186	2013-12-12
SHJ	신현제	1969	경남			170	2005-05-05

입력데이터 : orders

1	KJJ	운동화		30	2
2	KJJ	노트북	전자	1000	1
3	JYP	모니터	전자	200	1
4	KHK	모니터	전자	200	5
5	KJJ	청바지	의류	50	3
6	KHK	메모리	전자	80	10
7	SYW	책	서적	15	5
8	PYR	책	서적	15	2
9	PYR	청바지	의류	50	1
10	KHK	운동화		30	2
11	PYR	책	서적	15	1
12	KHK	운동화		30	2

① 데이터베이스를 생성하고 결과를 확인 (5점)

create database examtest;  
show databases;

Results 1

show databases

ABC Database
1 employees
2 examdb
3 examtest
4 exercise
5 information_schema
6 mysql

② 회원테이블을 생성하고 결과를 확인 (5점)

```
create table members(
  user_id char(8) primary key not null,
  name char(10) not null,
  birth_year int not null,
  addr char(2) not null,
  mobile_brand char(3) null,
  mobile_number char(8) null,
  user_height smallint null,
  reg_date date null);

desc members;
```

columns 1

desc members

ABC Field	ABC Type	ABC Null	ABC Key	ABC Default	ABC Extra
1 user_id	char(8)	NO	PRI	[NULL]	
2 name	char(10)	NO		[NULL]	
3 birth_year	int	NO		[NULL]	
4 addr	char(2)	NO		[NULL]	
5 mobile_brand	char(3)	YES		[NULL]	
6 mobile_number	char(8)	YES		[NULL]	
7 user_height	smallint	YES		[NULL]	
8 reg_date	date	YES		[NULL]	

③ 구매 테이블을 생성하고 결과를 확인 (5점)

```

create table orders(
  num int primary key not null AUTO_INCREMENT,
  order_id char(8) not null,
  prod_name char(6) not null,
  category char(4) null,
  price int not null,
  quantity smallint not null,
  FOREIGN key(order_id) references members(user_id));

```

```
desc orders;
```

columns 1 X

desc orders Enter a SQL expression to filter results (use Ctrl+Space)

	ABC Field	ABC Type	ABC Null	ABC Key	ABC Default	ABC Extra
1	num	int	NO	PRI	[NULL]	auto_increment
2	order_id	char(8)	NO	MUL	[NULL]	
3	prod_name	char(6)	NO		[NULL]	
4	category	char(4)	YES		[NULL]	
5	price	int	NO		[NULL]	
6	quantity	smallint	NO		[NULL]	

④ 회원 테이블 데이터를 입력하고 결과를 확인 (5점)

```

insert into members (user_id, name, birth_year, addr, mobile_brand, mobile_number, user_height, reg_date)
values
('KHK', '강효경', 1973, '서울', '010', '12345678', 176, '2013-05-05');
('PYR', '박유리', 1972, '경북', '011', '33445689', 174, '2014-03-03');

insert into members (user_id, name, birth_year, addr, mobile_brand, mobile_number, user_height, reg_date)
values
('KTY', '김태영', 1965, '경기', '018', '42589124', 172, '2010-10-10');

insert into members (user_id, name, birth_year, addr, mobile_brand, mobile_number, user_height, reg_date)
values
('JYP', '조용필', 1950, '경기', '011', '55981247', 166, '2009-04-04'),
('KJJ', '강진주', 1979, '경남', '011', '36458974', 173, '2012-04-04');

insert into members (user_id, name, birth_year, addr, mobile_brand, mobile_number, user_height, reg_date)
values
('KDK', '김도겸', 1971, '전남', '019', '56894137', 177, '2007-07-07'),
('LJB', '임재범', 1963, '서울', '016', '25697874', 182, '2009-09-09'),
('WMH', '원무현', 1987, '서울', '011', '25254671', 182, '2008-08-08'),
('SYW', '손영우', 1979, '서울', null, null, 186, '2013-12-12'),
('SHJ', '신현제', 1969, '경남', null, null, 170, '2005-05-05');

select * from members ;

```

	user_id	name	birth_year	addr	mobile_brand	mobile_number	user_height	reg_date
1	JYP	조용필	1,950	경기	011	55981247	166	2009-04-04
2	KDK	김도겸	1,971	전남	019	56894137	177	2007-07-07
3	KHK	강효경	1,973	서울	010	12345678	176	2013-05-05
4	KJJ	강진주	1,979	경남	011	36458974	173	2012-04-04
5	KTY	김태영	1,965	경기	018	42589124	172	2010-10-10
6	LJB	임재범	1,963	서울	016	25697874	182	2009-09-09
7	PYR	박유리	1,972	경북	011	33445689	174	2014-03-03
8	SHJ	신현제	1,969	경남	[NULL]	[NULL]	170	2005-05-05
9	SYW	손영우	1,979	서울	[NULL]	[NULL]	186	2013-12-12
10	WMH	원무현	1,987	서울	011	25254671	182	2008-08-08

⑤ 구매 테이블 데이터를 입력하고 결과를 확인 (5점)

word

©



```
DESC orders ;

select * from members ;
```

columns 1 X

DESC orders | Enter a SQL expression to filter results (use Ctrl+Space)

ABC Field	ABC Type	ABC Null	ABC Key	ABC Default	ABC Extra
num	int	NO	PRI	[NULL]	auto_increment
order_id	char(8)	NO	MUL	[NULL]	
prod_name	char(6)	NO		[NULL]	
category	char(4)	YES		[NULL]	
price	int	NO		[NULL]	
quantity	smallint	NO		[NULL]	

DESC members ;

columns 1 X

DESC members | Enter a SQL expression to filter results (use Ctrl+Space)

ABC Field	ABC Type	ABC Null	ABC Key	ABC Default	ABC Extra
1 user_id	char(8)	NO	PRI	[NULL]	
2 name	char(10)	NO		[NULL]	
3 birth_year	int	NO		[NULL]	
4 addr	char(2)	NO		[NULL]	
5 mobile_brand	char(3)	YES		[NULL]	
6 mobile_number	char(8)	YES		[NULL]	
7 user_height	smallint	YES		[NULL]	
8 reg_date	date	YES		[NULL]	

3. DML문을 활용한 SQL구문을 작성하고 결과를 도출(Capture) 하시오. (34점)

① '김도겸' 회원의 모든 정보를 조회 (2점)

DESC members ;

select \* from members where name LIKE '김도겸';

	ABC user_id	ABC name	123 birth_year	ABC addr	ABC mobile_brand	ABC mobile_number	123 user_height	reg_date
1	KDK	김도겸	1,971	전남	019	56894137	177	2007-07-07

② 키가 180 이상이고 1970년 이후 출생인 회원 조회 (4점)

select \* from members where user\_height >= 180 and birth\_year > 1790;


	ABC user_id	ABC name	123 birth_year	ABC addr	ABC mobile_brand	ABC mobile_number	123 user_height	reg_date
1	LJB	임재범	1,963	서울	016	25697874	182	2009-09-09
2	SYW	손영우	1,979	서울	[NULL]	[NULL]	186	2013-12-12
3	WMH	원우현	1,987	서울	011	25254671	182	2008-08-08

③ 지역이 '경남','전남'인 회원들의 지역과 이름을 조회 (in() ) (4점)

select addr ,name from members where addr in ('경남', '전남');

	ABC addr	ABC name
1	전남	김도겸
2	경남	강진주
3	경남	신현제

④ 회원의 키가 180 이상 183 이하인 회원의 이름과 키를 조회 (between ...and) (4점)

 `select name ,user_height from members where user_height between 180 and 183;`

members 1 X

 `select name ,user_height from members`   Enter a SQL expression to filter results (use Ctrl+Space)

	ABC name	123 user_height
1	임재범	182
2	원무현	182

⑤ 강씨 성을 가진 회원의 이름과 지역을 조회(4점)

 `select name ,addr from members where name like '강%';`

members 1 X

 `select name ,addr from members where`   Enter a SQL expression to filter results (use Ctrl+Space)

	ABC name	ABC addr
1	강효경	서울
2	강진주	경남

⑥ 휴대폰을 가진 회원 수를 조회(4점)

```
select count(mobile_number) from members ;
```

Results 1 X

select count(mobile\_number) from mem | Enter a SQL expression to filter results (use Ctrl+Space)

	123 count(mobile_number)
1	8

⑦ 'SYW' 아이디를 가진 회원의 구매 총 금액을 조회(4점)

```
select order_id ,sum(price*quantity) from orders where order_id like 'SYW' group by order_id ;
```

orders 1 X

select order\_id ,sum(price\*quantity) from | Enter a SQL expression to filter results (use Ctrl+Space)

	order_id	123 sum(price*quantity)
1	SYW	75

⑧ 구매액이 가장 낮은 주문건에 대한 아이디, 상품, '총구매액'을 조회 (4점)

```
select order_id , price , price*quantity from orders where price*quantity = (select min(price) from orders ) ;
```

orders 1 X

select order\_id , price , price\*quantity fro | Enter a SQL expression to filter results (use Ctrl+Space)

	order_id	123 price	123 price*quantity
1	PVR	15	15

⑨ 지역이 '경남'인 회원의 키보다 큰 회원들의 지역, 이름, 키를 조회 (4점)

SQL Query Editor showing the following query:

```
select addr ,name ,user_height from members where user_height > (select user_height from members where addr = '경남');
```

Results window (members 1) showing the following data:

	addr	name	user_height
1	서울	임재범	182
2	서울	손영우	186
3	서울	원무현	182

4. SQL구문을 작성하고 결과를 도출(Capture) 하시오. (16점)

- ① 샘플 데이터베이스인 employees.employees 테이블의 직원번호, 고용일, 이름(first\_name) 컬럼을 복사하고 결과 사번 110000번부터 10개 조회 (4점)

SQL Query Editor showing the following query:

```
create table emp_test ( select emp_no ,hire_date ,first_name from employees.employees);
```

Statistics window (Statistics 1) showing the following data:

Name	Value
Updated Rows	300024
Query	create table emp_test ( select emp_no ,hire_date ,first_name from employees.employees)
Start time	Mon Mar 20 15:28:07 KST 2023
Finish time	Mon Mar 20 15:28:10 KST 2023

select emp\_no from emp\_test limit 110000 ,10 ;

emp\_test 1 X

select emp\_no from emp\_test limit 1100 | Enter a SQL expression to filter results (u.

	123 emp_no
1	209,976
2	209,977
3	209,978
4	209,979
5	209,980
6	209,981
7	209,982
8	209,983
9	209,984
10	209,985

② emp\_test 테이블의 이름이 'sumant'이고 고용일이 '1986-07-25'인 데이터를 삭제 (4점)

delete from emp\_test where first\_name = 'sumant' and hire\_date = '1986-07-25';

Statistics 1 X

Name	Value
Updated Rows	1
Query	delete from emp_test where first_name = 'sumant' and hire_date = '1986-07-25'
Start time	Mon Mar 20 15:36:16 KST 2023
Finish time	Mon Mar 20 15:36:16 KST 2023

③ 1998년 이후 입사한 'Ymte'라는 이름을 가진 직원들의 이름을 'SnowMan'으로 수정 (4점)

update emp\_test set first\_name = 'SnowMan' where hire\_date >= 19980101 and first\_name = 'Ymte';

Statistics 1 X

Name	Value
Updated Rows	6
Query	update emp_test set first_name = 'SnowMan' where hire_date >= 19980101 and first_name = 'Ymte'
Start time	Mon Mar 20 15:45:39 KST 2023
Finish time	Mon Mar 20 15:45:40 KST 2023



④ 구매(delete) , 회원(), emp\_test(drop) 테이블의 자료 삭제 (4점)

The screenshot shows the SQL Developer interface with the query `drop table emp_test;` entered in the SQL Editor. Below the editor, the 'Statistics 1' window is open, displaying the following data:

Name	Value
Updated Rows	0
Query	drop table emp_test
Start time	Mon Mar 20 15:47:58 KST 2023
Finish time	Mon Mar 20 15:47:58 KST 2023

The screenshot shows the SQL Developer interface with the query `drop table orders ;` entered in the SQL Editor. Below the editor, the 'Statistics 1' window is open, displaying the following data:

Name	Value
Updated Rows	0
Query	drop table orders
Start time	Mon Mar 20 16:23:31 KST 2023
Finish time	Mon Mar 20 16:23:31 KST 2023

The screenshot shows the SQL Developer interface with the query `drop table members;` entered in the SQL Editor. Below the editor, the 'Statistics 1' window is open, displaying the following data:

Name	Value
Updated Rows	0
Query	drop table members
Start time	Mon Mar 20 16:23:46 KST 2023
Finish time	Mon Mar 20 16:23:46 KST 2023

[채점기준표]

환경설정	DDL	DML		합계(총점)
DBMS 설치	데이터베이스 및 테이블 생성	SQL 자료검색	대량 데이터 생성 및 삭제	
20	30	34	16	100