1. **많은 사용자들은 보다 효율적인 데이터의 관리 뿐만 아니라 사건으로 인한 데이터의 손상을 피하고, 데이터를 복구하기**

**위한 강력한 소프트웨어를 ( ) 시스템이라고 한다.**

1. **다음중 DBMS 특징이 아닌 것은?**
2. 자료는 여러 응용프로그램에 공유 할 수 없다.
3. 데이터 무결성을 유지 할 수 있다.
4. 데이터 보안 유지/표준화/ 통합성의 유지 용이하다
5. 실시간 접근, 자료의 계속적인 변화의 적용에 유리하다.
6. 인증된 사용자만이 참조 할 수 있는 보안기능이 제공된다
7. **관계형 데이터베이스에서 데이터 정의, 데이터 조작, 데이터 제어을 하기위해사용하는 언어는 ( ) 이다**
8. **테이블 생성시 주의 할 사항으로 적절하지 않은 것은?**
9. 컬럼 뒤에 데이터 유형은 꼭 지정되어야 한다.
10. 테이블 생성시 대소문자 구분은 하지 않는다.
11. 모든 특수문자 사용이 가능하다.
12. 테이블명과 컬럼명은 반드시 문자로 시작해야한다.
13. 컬럼들은 ',' 구분되고, 테이블 생성문의 끝은 ';'로 끝난다.
14. **제약조건이란 ( )를 위한 데이터베이스의 보편적인 방법으로 테이블의 특정 컬럼에 설정하는 제약이다.**
15. **다음 중 설명이 올바르지 않은 것은?**
16. 테이블 전체 데이터 삭제시 , 시스템 활용 측면에선 DML명령어 구문이 DDL 명령어 보다 뛰어나다.
17. 테이블 생성 후 필요에 의해 제약조건을 추가 할 수 있다.
18. 테이블 생성 후 컬럼 명은 'RENAME COLUMN' 문구로 컬럼 명 변경이 가능하다.
19. 테이블 생성 후 컬럼의 크기는 줄일 수 없다.
20. 테이블 생성 후 DROP COLUMN 구문은 하나의 컬럼만 삭제 가능하다.
21. **트랜잭션의 특성이 아닌 것은?**
22. 일관성 2) 원자성 3) 고립성 4) 지속성 5) 중복성
23. **테이블 생성 시 규칙을 설명한 것이다. 다음 중 틀린 설명은?**
24. 한 테이블 내에서는 칼럼명이 중복되게 지정될 수 없다.
25. 칼럼 뒤에 데이터 유형은 꼭 지정되어야 한다.
26. 테이블 명은 다른 테이블의 이름과 중복되어도 된다.
27. 다른 테이블의 칼럼 이름과는 같을 수 있다.
28. **제약조건에 대한 설명 중 틀린 것은?**
29. UNIQUE KEY : 테이블에 저장된 행 데이터를 고유하게 식별하기 위한 고유키를 정의한다.
30. PK : 하나의 테이블에 하나의 기본키 제약만 정의가능하다.
31. NOT NULL : 해당 칼럼은 입력 필수이다.
32. FK : 관계형 DB에서 테이블 간의 관계를 정의하기 위해 외래키를 다른 테이블의 외래키로 복사하는 경우 생성된다.
33. **ALTER TABLE에 대한 설명으로 옳은 것은?**
34. 새롭게 추가된 컬럼은 컬럼의 위치를 지정할 수 있다.
35. DROP COLUMN 은 데이터가 있을 경우에만 가능하다.
36. DROP COLUMN 시 한번에 하나의 컬럼만 삭제 가능하다.
37. MODIFY COLUMN 시 컬럼의 크기를 줄이지는 못한다.
38. **DML에 대한 설명으로 옳은 것은?**
39. DELETE 사용 시FROM 문구는 생략이 불가능하다.
40. 원하는 데이터 DELETE 시 WHERE 절은 반드시 붙이지 않아도 된다.
41. DML 문장은 실시간으로 테이블에 영향을 미친다.
42. DML은 반드시 COMMIT을 입력하여 TRANSACTION을 종료해야 한다.
43. **COMMIT 이전 상태에 대한 설명으로 옳지 않은 것은?**
44. 현 사용자는 SELECT 문장으로 결과를 확인 가능하다.
45. 다른 사용자도 현재 사용자가 수행한 명령의 결과를 볼 수 없다.
46. 단지 메모리 BUFFER에만 영향을 받기 때문에 데이터 변경 이전 상태로 복구 가능하다.
47. 변경된 행은 LOCK이 설정되어 다른 사용자가 변경할 수 없다.
48. 데이터에 대한 변경 사항이 데이터 베이스에 반영된다.
49. **다음 중 테이블 생성 시에 주의해야 할 규칙으로 적절하지 않은 것은?**
50. 테이블 명은 다른 테이블의 이름과 중복되지 않아야 한다.
51. 한 테이블 내에서는 칼럼명이 중복되게 지정될 수 없다.
52. 칼럼 뒤에 데이터 유형은 꼭 지정되어야 한다.
53. 테이블명과 컬럼명은 문자,숫자로 시작할 수 있고, 벤더별로 길이에 대한 한계가 있다.
54. 각 칼럼들은 콤마 “,”로 구분되고, 생성문의 끝은 항상 세미콜론 “;”으로 끝난다.
55. **테이블 내의 data를 삭제하는 방법으로는 truncate table과 delete 문이 있다. 두 방법의 차이로 올바르지 않은 것은? (2개)**
56. Truncate table은 DDL, delete 문은 DML이다.
57. Truncate table은 조건에 해당하는 data만 삭제하는 것이 불가능하다.
58. 시스템 활용 측면에서는 delete 문을 사용해 전체 삭제하는 것을 권장한다.
59. 두 방법 중 truncate table을 사용하면 복구가 불가능하다.
60. 실행문의 차이일 뿐 처리하는 방식 자체는 동일하다.
61. **다음 중 트랜잭션의 특성에 해당하지 않는 것은?**
    1. 원자성
    2. 독립성
    3. 일관성
    4. 고립성
    5. 지속성
62. **다음 연산자 중 구분이 다른 하나는?**
63. !=
64. ^=
65. <>
66. IS NOT NULL
67. NOT 칼럼명 =
68. **아래 두 쿼리에 대한 실행결과가 같으면 O, 다르면 X**

SELECT msi.organization\_id, msi.inventory\_item\_id

FROM mtl\_system\_items\_b msi

WHERE 1=1

AND (msi.segment1,msi.organization\_id) IN (('4664005100',262),('46640061',104))

;

SELECT msi.organization\_id, msi.inventory\_item\_id

FROM mtl\_system\_items\_b msi

WHERE 1=1

AND msi.segment1 IN ('4664005100','46640061')

AND msi.organization\_id IN (262,104)

1. **다음 중 데이터 유형에 대한 내용 중 옳지 않은 것은?**
2. 대표적인 데이터 유형으로는 CHARACTER, VARCHAR, NUMERIC, DATETIME이 있다.
3. CHAR 유형은 공백도 하나의 문자로 취급한다.
4. VARCHAR 유형은 40바이트로 지정되더라도 실제 데이터만큼의 공간만 차지한다.
5. 테이블에 정의된 잘못된 데이터유형은 ALTER TABLE로 수정 가능하다.
6. 숫자 타입은 NUMBER뿐이다.
7. **테이블 생성 시 주의사항 중 옳지 않은 것은?**
8. 테이블 생성 시 대/소문자는 구별하지 않는다.
9. DATETIME 데이터 유형은 별도로 크기 지정을 하지 않는다.
10. 문자 데이터 유형은 반드시 가질 수 있는 최대 길이를 표시해야 한다.
11. 테이블명은 반드시 문자로 시작해야 한다.
12. 칼럼에 대한 제약 조건은 최초 생성 시에만 추가할 수 있다.
13. **트랜잭션의 특징 중 옳지 않은 것은?**
14. 일관성(consistency)
15. 원자성(atomicity)
16. 지속성(durability)
17. 중복성(duplication)
18. 고립성(isolation)
19. **ALTER TABLE에 대한 설명 중 옳은 것은?**
20. DROP COLUMN을 이용하여 한 번에 다수의 컬럼을 삭제 할 수 있다.
21. MODIFY COLUMN을 이용하여 컬럼의 크기를 늘릴 수는 있지만 줄일 수는 없다.
22. TRUNCATE TABLE을 통해 테이블 구조를 삭제할 수 있다.
23. RENAME TABLE을 이용하여 테이블명과 컬럼명을 바꿀 수 있다.
24. CREATE TABLE을 할 때 기존 테이블을 이용하기 어렵다.
25. **SQL 연산자 내용 중 옳지 않은 것은?**
26. LIKE ‘비교문자열’ - 비교 문자열과 형태가 일치하면 된다.
27. IS NULL - NULL값인 경우
28. BETWEEN a AND b – a와 b 값 사이에 있는 경우(a 포함)
29. IN (list) – 리스트에 있는 값 중 어느 하나라도 일치하면 된다.
30. **테이블 생성 시 주의 사항으로 옳지 않은 것은?**
31. 테이블 생성시 대/소문자 구분은 하지 않아도 된다.
32. Datetime 데이터 유형은 별도로 크기를 지정하지 않아도 된다.
33. 문자 데이터 유형 입력 시 최대 길이에 대해 표시하지 않아도 된다.
34. 컬럼 사이 구분은 콤마로 하되, 마지막 컬럼은 콤마를 찍지 않는다.
35. 컬럼에 대한 제약조건을 추가하는 경우 CONSTRAINT 를 이용한다.
36. **컬럼 변경 시 주의 사항으로 옳지 않은 것은?**
37. 해당 컬럼의 크기를 늘릴 수는 있지만 줄일 수는 없다.
38. 컬럼이 Null 값만 가지고 있어도 컬럼의 폭을 줄일 수는 없다.
39. 컬럼이 Null 값만 가지고 있으면 데이터 유형을 변경할 수 있다.
40. 해당 컬럼의 Default 값을 바꾸면 변경 작업 이후 발생하는 행 삽입에만 영향을 미친다.
41. 해당 컬럼에 Null 값이 없을 경우에만 Not null 제약조건을 추가할 수 있다.
42. **Commit 이후의 데이터 상태로 옳지 않은 것은?**
43. 데이터에 대한 변경 사항이 데이터 베이스에 반영된다.
44. 이전 데이터는 영원히 잃어버리게 된다.
45. Commit 사용자만 결과를 볼 수 있다.
46. 관련된 행에 대한 잠금이 풀리고 다른 사용자들이 행을 조작할 수 있다.
47. **연산자의 우선 순위를 나열한 것으로 옳은 것은?**

ㄱ. 괄호 () ㄴ. Not 연산자

ㄷ. 비교 연산자 ㄹ. AND ㅁ. OR

1. ㄱ – ㄴ – ㄷ – ㄹ – ㅁ
2. ㄴ – ㄱ – ㄷ – ㄹ – ㅁ
3. ㄴ – ㄱ – ㄷ – ㅁ – ㄹ
4. ㄱ – ㄴ - ㄷ – ㅁ – ㄹ
5. **Null 에 대한 설명으로 틀린 것은?**
6. 값이 존재하지 않거나 확정되지 않은 값을 의미한다.
7. Null 과의 수치연산은 Null 값을 리턴한다.
8. Null 과의 비교연산은 False(거짓)를 리턴한다.
9. 공백과 같은 ASCII 값을 가진다.
10. Is null, is not null이라는 정해진 문구만을 사용해야 한다.
11. **다음 SQL 문장들의 종류와 다른 것은?**
12. SELECT / INSERT / UPDATE / DELETE는 DCL이다
13. 테이블과 같은 데이터 구조를 정의하는데 사용되는 명령어를 DDL이라고 한다.
14. TCL은 트랜잭션 제어어로 COMMIT / ROLLBACK을 의미한다.
15. 데이터 베이스에 접근하여 객체들을 사용하도록 권한을 주고 회수하는 명령어는 DCL이다.
16. **다음 중 의미하는 용어가 무엇인가요?**

사용자가 원하는 조건의 데이터만 유지하기 위한 것이며

데이터의 무결성을 유지하기 위한 데이터베이스의 보편적인 방법으로

종류로는 PK, FK, Unique Key등이 있다.

1. **트랜잭션의 특성 그 의미와 맞는 것은?**
2. 일관성 : 정의된 연산들은 모두 성공적으로 실행되던지 아니면 전혀 실행되지 않은 상태로 남아 있어야 한다.
3. 원자성 : 트랜잭션이 실행되기 전 데이터의 내용이 잘못되면 실행 이후에도 내용이 잘못 되어 있지 않다.
4. 고립성 : 트랜잭션이 실행되는 도중 다른 트랜잭션의 영향을 받아 잘못된 결과를 만들어서는 안된다
5. 연관성 : 트랜잭션이 성공적으로 수행되면 그 트랜잭션이 갱신한 데이터베이스의 내용은 영구적으로 저장된다.
6. **COMMIT 이후의 데이터 상태와 거리가 먼 것은?**
7. 데이터에 대한 변경 사항 데이터베이스에 반영된다
8. 단지 메모리 BUFFER에만 영향을 받았기 때문에 데이터 변경 이전 상태 복구가 가능하다
9. 모든 사용자는 결과를 볼 수 있다
10. 관련된 행에 대한 LOCKING이 풀린다.
11. **다음 설명에 대한 용어는 무엇인가요?**

트랜잭션에 포함된 전체 작업을 Rollback이 아닌 저장 지점까지 작업 롤백이 가능하며 일부 툴에서는 지원이 되지 않는다.

1. **다음 중 NULL의 설명으로 가장 부적절한 것은?**
2. 모르는 값을 의미한다.
3. 값의 부재를 의미한다.
4. 공백문자(Empty String) 혹은 숫자 0을 의미한다.
5. NULL과의 모든 비교는 알 수 없음(Unknown)을 반환한다.
6. **다음 SQL 문장 중 COLUMN1의 값이 널(NULL)이 아닌 경우를 찾아내는 문장으로 가장 적절한 것은? (ANSI 표준 기준)**
7. SELECT \* FROM MYTABLE WHERE COLUMN1 IS NOT NULL
8. SELECT \* FROM MYTABLE WHERE COLUMN1 <> NULL
9. SELECT \* FROM MYTABLE WHERE COLUMN1 != NULL
10. SELECT \* FROM MYTABLE WHERE COLUMN1 NOT NULL
11. **하기 설명에 해당하는 예약어를 쓰시오.( )**

데이터 입력시 컬럼의 값이 지정되어 있지 않을때 입력될 기본값을 지정한다.

1. **제약조건의 설명 중 아닌 것은?**
2. 기본키는 테이블에 저장된 행 데이터를 고유하게 식별하기 위한 키이다.
3. 테이블에 저장된 행 데이터를 고유하게 식별하기 위해 고유키를 정의한다.
4. 기본키 제약 = 고유키 제약 & 외래키 제약을 합쳐놓은 것이다.
5. Not null 은 Null 값의 제약을 금지한다.
6. **CTAS 기법으로 생성한 테이블은 기존 테이블의 제약조건 중에 ( ) 만 새로운 복제 테이블에 적용된다.**
7. **Truncate 와 Delete 의 차이점 1가지는?**
8. **칼럼을 변경할 때 몇 가지 고려사항이다. 이중 틀린 것은?**
9. 해당칼럼의 크기가 줄어들 수 있어도 늘릴수는 없다.
10. Row에 데이터가 있지만, 아무행도 가지고 있지 않으면 컬럼의 폭을 줄일수 있다.
11. Default 값을 바꾸면 변경 작업 이후 발생하는 행 상입에만 영향을 미친다.
12. 해당 컬럼에 Null 값이 없을 경우에만 Not Null 제약조건을 추가할 수 있다.
13. **Drop 명령어를 사용하면 테이블의 모든 구조를 삭제한다. ( ) 옵션은 해당 테이블과 관계가 있었던 참조되는 제약조건에 대해서도 삭제한다는 것을 의미한다.**

1. DBMS

2. 1

3. SQL

4. 3

5. 데이터 무결성 유지

6. 1

7. 5

8. 3

9. 4

10. 3

11. 4

12. 5

13. 4

14. 3,5

15. 2

16. 4

17. X

18. 2,4

19. 5

20. 4

21. 답없음

22. 3

23. 3

24. 1, 2

25. 3

26. 1

27. 4

28. 1

29. 제약조건

30. 3

31. 2

32. Savepoint

33. 3

34. 1

35. Default

36. 3

37. not null

38. truncate는 Auto Commit을 수행

39. 1, 2

40. Cascade constraint