

12 ஆம் வகுப்பு – கணினி அறிவியல்

பாடம் 12.
வினவல் அமைப்பு மொழி

Prepared by,

J. KAVITHA, B.Sc,B.Ed,M.C.A,M.Phil.,
Computer Instructor Gr – I,
GHSS, S.S.KULAM,
Coimbatore.

கற்றலின் நோக்கங்கள்

- SQL வினவல் மொழியின் செயலாக்க திறன்.
- SQL-ன் கூறுகள்.
- புலங்கள் மற்றும் பதிவுகளை குறிப்பிட்டு ஒரு அட்டவணையை உருவாக்குதல்.
- ஒரு அட்டவணையில் பதிவுகளை சேர்த்தல், மேம்படுத்துதல் (புதுப்பித்தல்) மற்றும் நீக்குதல் போன்ற பல்வேறு கையாளுதலை செயற்படுத்துதல்.
- பல்வேறு நிபந்தனைகளையும் அவற்றை அட்டவணைகளில் பயன்படுத்தும் வழிமுறைகளையும் தெரிந்து கொள்ளுதல்.
- பல்வேறு உட்பிரிவுகளைப் பயன்படுத்துவதன் மூலம் அட்டவணையில் வினவல்களை உருவாக்குதல்.
- ஒரு அட்டவணை மற்றும் பதிவுகளை நீக்குவதற்கான கட்டளைகளையும், செயல்படுத்தப்பட்ட கட்டளைகளை திரும்ப பெறும் கட்டளைகள் பற்றி அறிந்து கொள்ளுதல்.

SQL அறிமுகம்

- வினவல் அமைப்பு மொழி (Structured Query Language) என்பது தரவு தளங்களை அணுகுவதற்கும், கையாளுவதற்கும் பயன்படுத்தப்படும் ஒரு வினவல் மொழியாகும்.
- SQL, பயனருக்கு தரவுதளங்களை உருவாக்குதல், திரும்ப எடுத்தல், மாற்றி அமைத்தல் மற்றும் தகவல்களை பரிமாற்றம் செயதல் போன்ற செயல்பாடுகளை செயல்படுத்த அனுமதிக்கிறது.
- பொதுவாக, தரவுத்தளம் என்பது தொடர்புடைய தரவுகளைக் கொண்ட அட்டவணையின் தொகுப்பைக் கொண்ட ஒரு களஞ்சியம் ஆகும்.

உறவுநிலை தரவுத்தள மேலாண்மை அமைப்பில் SQL – ன் பங்கு

- உறவுநிலை தரவுத்தள மேலாண்மை (RDBMS) என்பது தொடர்புடைய தரவு கூறுகளை இணைக்கும் வரிசை அடிப்படையிலான அட்டவணை வடிவமைப்பைக் கொண்ட ஒரு வகை தரவுத்தள மேலாண்மை அமைப்பாகும்.
- Oracle, MySQL, MS SQL Server, IBM DB2 மற்றும் Microsoft Access போன்றவை உறவுநிலை தரவுத்தள மேலாண்மை அமைப்பிற்கான (RDBMS) மென்பாருள் தொகுப்புகளாகும்.

உறவுநிலை தரவுத்தள மேலாண்மை அமைப்பில் SQL - ன் பங்கு

- இத்தகைய தரவுத்தளங்களுடன் தொடர்பு கொண்டு தரவுகளை கையாள SQL மொழி பயன்படுகிறது.
- Create, Read, Update மற்றும் Delete போன்ற செயற்பாடுகள் ஒட்டுமொத்தமாக CRUD எனப்படும்.
- RDBMS - ல் உள்ள தரவுகள், தரவுத்தள பொருட்களாக அட்டவணைகளில் (Tables) சேமிக்கப்படும்.

உறவுநிலை தரவுத்தள மேலாண்மை அமைப்பில் SQL – ன் பங்கு

- ஒரு அட்டவணையின் நெடுவரிசை புலம் (field) என்று குறிப்பிடப்படுகிறது.

எ.கா: ஒரு மாணவர் அட்டவணையில் புலங்கள் AdmNo, studName, studAge, studclass, place ஆகிய புலங்களைக் கொண்டிருக்கலாம்.

- ஒரு அட்டவணையில் உள்ள தொடர்புடைய புலங்கள் அல்லது நெடுவரிசைகளின் தொகுப்பைக் கொண்ட ஒரு வரிசை, **பதிவு (Record)** எனப்படும்.

எ.கா: ஒரு பதிவு என்பது மாணவர் அட்டவணையில் உள்ள ஒரு குறிப்பிட்ட மாணவரின் விவரங்களைக் கொண்டிருக்கும் ஒரு வரிசையாகும்.

SQL மற்றும் MySQL க்கு உள்ள வேறுபாடுகள்

SQL	MySQL
வினவல் அமைப்பு மொழி (SQL) என்பது தரவு தளங்களை அணுகுதலுக்கான ஆகும்.	MySQL என்பது உறவுநிலை தரவுதள மேலாண்மை அமைப்பிற்கான மென்பொருளாகும்.
உறவுநிலை தரவுதளங்களை உருவாக்கவும், செயல்படுத்தவும் பயன்படும் மொழியாகும்.	தரவுதளங்களை வினவ SQL – ஐ பயன்படுத்துகிறது.

SQL கூறுகள்

SQL கட்டளைகள் 5 பிரிவுகளாகப் பிரிக்கப்பட்டுள்ளது.

- DML - தரவு கையாளுதல் மொழி
- DDL - தரவு வரையறை மொழி
- DCL - தரவு கட்டுப்பாட்டு மொழி
- TCL - பரிவர்த்தனை கட்டுப்பாட்டு மொழி
- DQL - தரவு வினவல் மொழி

SQL கூறுகள்

DDL (Data Definition Language)
தரவு வரையறை மொழி

தரவுத்தள அமைப்பு அல்லது திட்ட வடிவமைப்பினை வரையறுக்கும் SQL கூற்றுகளைக் கொண்டிருக்கும்.

DML (Data Manipulation Language)
தரவு கையாளுதல் மொழி

ஒரு தரவுத்தளத்தில் உள்ள தரவுகளை சேர்த்தல், அழித்தல் மற்றும் மாற்றியமைக்க பயன்படும் போன்ற SQL கூற்றுகளைக் கொண்டிருக்கும்.

DCL (Data Control Language)
தரவு கட்டுப்பாட்டு மொழி

தரவுத்தளத்தில் சேமித்து வைக்கப்படும் தரவுகளின் அணுகுதலை கட்டுப்படுத்தும் SQL கூற்றுகளைக் கொண்டிருக்கும்.

SQL கூறுகள்

TCL (Transaction Control Language) பரிவர்த்தனை கட்டுப்பாட்டு மொழி

தரவுதளத்தில் உள்ள பரிவர்த்தனைகளை நிர்வகிக்க பயன்படுகிறது.

DQL (Data Query Language) தரவு வினவல் மொழி

தரவுதளத்தில் உள்ள தரவுகளை வினவுதலுக்கும், மீட்டெடுப்பதற்குமான கட்டளைகளைக் கொண்டுள்ளது.

தரவு வகைகள்

- தரவுகளத்தில் தரவுகள் மதிப்புகளின் அடிப்படையில் சேமிக்கப்படும். இது தரவு வகையாக அடையாளப் படுத்தப்படுகிறது.
- SQL - ல் char, varchar, dec, numeric, int, smallint, float, real, double போன்ற தரவு வகைகள் பயன்படுத்தப்படுகின்றன.

SQL கட்டளைகளும், அதன் செயல்பாடுகளும்

SQL, தரவுதளத்துடன் செயலாற்றுவதற்கான முன் - வரையறுக்கப்பட்ட கட்டளைகளின் தொகுதியை வழங்குகிறது.

SQL கட்டளைகள்	செயல்பாடுகள்
சிறப்புச் சொற்கள்	SQL - ல் இவை சிறப்பு பொருளைக் கொண்ட கட்டளைகளாக அறியப்படுகிறது.
கட்டளைகள்	கூற்றுகள் என அறியப்படும் இவை பயனரால் தரவுதளத்திற்கு கொடுக்கப்படும் அறிவுறுத்தல்களைக் கொண்டிருக்கும்.
Clauses	இவை சிறப்புச் சொல்லுடன் தொடங்கி, சிறப்புச் சொல்லையும், செயலுருபையும் கொண்டிருக்கும்
செயலுருபுகள்	இவை Clauses ஐ நிறைவுப் பெற செய்ய கொடுக்கப்படும் மதிப்புகளைக் குறிக்கும்.

DDL (Data Definition Language) தரவு வரையறை மொழி: கட்டளைகள்

CREATE	தரவுதளத்தில் அட்டவணைகளை உருவாக்கும்.
ALTER	தரவுதளத்தின் வடிவமைப்பை மாற்றியமைக்கும்.
DROP	தரவுதளத்தில் உள்ள அட்டவணைகளை நீக்கும்.
TRUNCATE	ஒரு அட்டவணையில் உள்ள அனைத்து பதிவுகளையும் அழிக்கும்.

CREATE - கட்டளை

- **CREATE** என்ற கட்டளையை பயன்படுத்தி தரவுகளத்தில் ஒரு அட்டவணையை உருவாக்கலாம்.

தொடரியல்:

```
CREATE TABLE <table-name>  
(<column name><data type>[<size>]  
<column name><data type>[<size>].....);
```

எடுத்துக்காட்டு

```
CREATE TABLE Student  
(Admno integer,  
Name char(20),  
Gender char(1),  
Age integer,  
Place char(10),  
);
```

ALTER கட்டளை

- ALTER கட்டளை தரவுகளத்தின் வடிவமைப்பை மாற்றியமைக்க பயன்படுகிறது.

தொடரியல் 1:

```
ALTER TABLE <table-name> ADD <column-name><data type><size>;
```

எ.கா:

```
ALTER TABLE Student ADD Address char;
```

தொடரியல் 2:

```
ALTER <table-name> MODIFY<column-name><data type><size>;
```

எ.கா:

```
ALTER TABLE Student MODIFY Address char (25);
```

தொடரியல் 3:

```
ALTER <table-name> RENAME old-column-name TO new-column-name;
```

எ.கா:

```
ALTER TABLE Student RENAME Address TO City;
```

DROP கட்டளை

- DROP கட்டளை தரவுதளத்திலிருந்து ஒரு அட்டவணையை நீக்க பயன்படுகிறது.

தொடரியல்:

```
DROP TABLE table-name;
```

எ.கா:

```
DROP TABLE Student;
```


TRUNCATE - கட்டளை

- **TRUNCATE** – ஒரு அட்டவணையில் உள்ள அனைத்து பதிவுகளையும் அழிக்கும். மேலும் அவற்றிற்கு ஒதுக்கப்பட்ட நினைவக இடத்தையும் விடுவிக்கும்.

தொடரியல்:

```
TRUNCATE TABLE table-name;
```

எ.கா:

```
TRUNCATE TABLE Student;
```

DML (Data Manipulation Language) தரவு கையாளுதல் மொழி: கட்டளைகள்

INSERT	ஒரு அட்டவணையில் தரவுகளைச் சேர்த்தல்
UPDATE	அட்டவணையில் ஏற்கனவே உள்ள தரவுகளைப் புதுப்பித்தல்
DELETE	அட்டவணையில் உள்ள அனைத்து பதிவுகளையும் நீக்க

INSERT கட்டளை

INSERT கட்டளை தரவுகளத்தில் புதிய தரவை சேர்ப்பதற்கும் அல்லது ஒரு அட்டவணையில் புதிய பதிவுகளை சேர்ப்பதற்கும் பயன்படுகிறது

தொடரியல்:

```
INSERT INTO <table-name> [column-list] VALUES (values);
```

எ.கா:

```
INSERT INTO Student (Admno, Name, Gender, Age, Place)
```

```
VALUES (100, 'Ashish', 'M', 17, 'Chennai');
```

```
INSERT INTO Student (Admno, Name, Gender, Age, Place)
```

```
VALUES (102, 'Adarsh', 'M', 18, 'Delhi');
```

UPDATE கட்டளை

UPDATE கட்டளை அட்டவணையில் உள்ள சில அல்லது அனைத்து தரவுகளையும் புதுப்பிக்கப் பயன்படுகிறது.

தொடரியல்:

```
UPDATE <table-name> SET column-name = value,  
column-name = value,... WHERE condition;
```

எ.கா:

```
UPDATE Student SET Age = 20 WHERE Place = "Bangalore";
```

DELETE - கட்டளை

DELETE - அட்டவணையில் உள்ள அனைத்து பதிவுகளையும் நீக்கும். ஆனால் அவற்றிற்கு ஒதுக்கப்பட்ட நினைவக பகுதியை விடுவிக்காது.

தொடரியல்:

```
DELETE FROM table-name WHERE condition;
```

எ.கா:

```
DELETE FROM Student WHERE Admno=104;
```

```
DELETE * FROM Student;
```

அட்டவணையின் அனைத்து வரிசைகளும் நீக்கப்பட்ட பின்பு, DROP TABLE என்ற கட்டளையை பயன்படுத்தி முழு அட்டவணையையும் நீக்க முடியும்.

DCL (Data Control Language) தரவு கட்டுப்பாட்டு மொழி : கட்டளைகள்

GRANT	ஒரு குறிப்பிட்ட பணியை செய்ய ஒன்று அல்லது அதற்கு மேற்பட்ட பயனர்களுக்கு அனுமதி வழங்கும்.
REVOKE	GRANT ஆல் வழங்கப்பட்ட அணுகல் அனுமதியை திரும்பப் பெறும்.

TCL (Transaction Control Language) பரிவர்த்தனை கட்டுப்பாட்டு மொழி: கட்டளைகள்

COMMIT	தரவுத்தள பரிவர்த்தனையை நிரந்தரமாக சேமிக்க
ROLL BACK	ஒரு தரவுத்தளத்தை முந்தைய COMMIT நிலைவரை மீட்டெடுக்கும்.
SAVE POINT	ROLL BACK செய்வதற்கு ஏதுவாக பரிவர்த்தனையை தற்காலிகமாக சேமிக்கும்.

COMMIT கட்டளை

- COMMIT கட்டளை தரவுத்தள பரிவர்த்தனையை நிரந்தரமாக சேமிக்க பயன்படுகிறது.
- INSERT, UPDATE, DELETE போன்ற DML கட்டளைகளை பயன்படுத்தி செய்யப்பட்ட மாற்றங்கள் COMMIT கட்டளை கொடுத்த பிறகுதான் நிரந்தரமாக சேமிக்கப்படுகிறது.
- COMMIT கட்டளை கொடுத்துவிட்டால் செய்யப்பட்ட மாற்றங்கள் திரும்ப பெற இயலாது.

தொடரியல்:

```
COMMIT;
```


ROLL BACK கட்டளை

- **ROLL BACK** கட்டளை ஒரு தரவுதளத்தை முந்தைய COMMIT நிலைவரை திரும்பப் பெறும்.
- **SAVE POINT** கட்டளையும் பயன்படுத்தி ஒரு குறிப்பிட்ட savepoint இடத்திற்கு கொண்டு செல்ல முடியும்.

தொடரியல்:

```
ROLL BACK TO savepoint name;
```

SAVE POINT - கட்டளை

- SAVE POINT கட்டளை ROLL BACK செய்வதற்கு ஏதுவாக பரிவர்த்தனையை தற்காலிகமாக சேமித்து வைப்பதால் நாம் தேவைப்படும் சமயங்களில் திரும்பவும் அந்த நிலைக்கு செல்லலாம்.

தொடரியல்:

```
SAVEPOINT savepoint_name;
```

DQL (Data Query Language) தரவு வினவல் மொழி: கட்டளைகள்

SELECT - அட்டவணையில் உள்ள பதிவுகளை வெளிகாட்டும்.

தொடரியல்:

```
SELECT <column-list>FROM<table-name>;
```

எ.கா:

```
SELECT Admno, Name FROM Student;
```

சிறப்புச் சொற்கள்

DISTINCT சிறப்புச் சொல்லுடன் SELECT கட்டளையை பயன்படுத்தி ஒரு அட்டவணையில் இரட்டிப்பு வரிசைகள் உருவாகாமல் தடுக்கலாம்.

எ. கா:

```
SELECT * DISTINCT place FROM student;
```

ALL என்ற சிறப்புச் சொல் அட்டவணையின் குறிப்பிட்ட அனைத்து வரிசைகளையும் வெளிக்காட்டும்.

எ.கா:

```
SELECT ALL Place FROM Student;
```

சிறப்புச் சொற்கள்

BETWEEN என்ற சிறப்புச் சொல்லை பயன்படுத்தி அட்டவணையில் ஒரு குறிப்பிட்ட நிபந்தனை கொடுத்து அவற்றின் பதிவுகளை பெறலாம்.

எ.கா:

```
SELECT Admno, Name, Age, Gender FROM Student WHERE Age BETWEEN 18 AND 19;
```

NOT BETWEEN நிபந்தனை நிறைவு செய்யாத பதிவுகளை வெளிக்காட்டும்.

எ.கா:

```
SELECT Admno, Name, Age FROM Student WHERE Age NOT BETWEEN 18 AND 19;
```

சிறப்புச் சொற்கள்

IN சிறப்புச் சொல் பட்டியலுடன் பொருந்தக்கூடிய
மதிப்புகளைக் கொண்ட பதிவுகளை குறிப்பிடப்
பயன்படுகிறது.

எ.கா:

```
SELECT Admno, Name, Place FROM Student WHERE Place IN  
("Chennai", "Delhi");
```

NOT IN சிறப்புச் சொல் பட்டியலுடன் பொருந்தாத
மதிப்புகளைக் கொண்ட பதிவுகளை மட்டுமே
வெளிக்காட்டும்.

எ.கா:

```
SELECT Admno, Name, Place FROM Student WHERE Place NOT IN  
("Chennai", "Delhi");
```

CLAUSES

- ORDER BY clause பயன்படுத்தி தரவுகளை ஒன்று அல்லது அதற்கு மேற்பட்ட நெடுவரிசைகளின் அடிப்படையில் ஏறுவரிசையிலோ அல்லது இறங்கு வரிசையிலோ வரிசைப்படுத்த பயன்படுகிறது.

தொடரியல்:

```
SELECT <column-name> FROM <table-name> ORDER BY  
<column1>,<column2>,...ASC| DESC ;
```

எ.கா:

```
SELECT * FROM Student ORDER BY Name;
```

CLAUSES

- WHERE clause பயன்படுத்தி கொடுக்கப்பட்ட நிபந்தனையை நிறைவு செய்யும் பதிவுகளை மட்டும் பிரித்தெடுக்க முடியும்.

எ.கா:

```
SELECT * FROM Student WHERE Age >= 18 ORDER BY Name;
```


CLAUSES

- GROUP BY clause பயன்படுத்தி வரிசைகள் அல்லது நெடுவரிசைகளில் ஒரே மாதிரியான மதிப்புகளைக் கொண்ட மாணவர்களை ஒரு தொகுதியாக பிரிக்கலாம்.

தொடரியல்:

```
SELECT <column-names> FROM <table-name> GROUP BY  
<column-name>HAVING [condition];
```

எ.கா:

```
SELECT Gender FROM Student GROUP BY Gender;
```

CLAUSES

- **HAVING** clause ஐ **GROUP BY** clause உடன் சேர்த்து பயன்படுத்தி குழுக்களுக்கு நிபந்தனையை இருத்தலாம்.

எ.கா:

```
SELECT Gender , count(*) FROM Student GROUP BY  
Gender HAVING Place = 'Chennai';
```

கட்டுப்பாடுகள்

- கட்டுப்பாடுகள் என்பது ஒரு புலம் அல்லது புலங்களின் தொகுதிக்கு பயன்படுத்தக்கூடிய நிபந்தனையை குறிக்கும்.

அட்டவணை கட்டுப்பாடு	தனிப்பட்ட நெடுவரிசைகளுக்கு மட்டுமே பயன்படுத்த முடியும்.
நெடுவரிசை கட்டுப்பாடு	ஒன்று அல்லது அதற்கு மேற்பட்ட நெடுவரிசைகளுக்கு பயன்படுத்த முடியும்.

கட்டுப்பாடுகளின் வகைகள்

கட்டுப்பாடுகள் பல வகைப்படும். அவை,

- Unique கட்டுப்பாடு
- Primary key கட்டுப்பாடு
- Default கட்டுப்பாடு
- Check கட்டுப்பாடு
- Table கட்டுப்பாடு

Unique கட்டுப்பாடு

- குறிப்பிட்ட நெடுவரிசையில் எந்த இரு வரிசைகளும் ஒரே மதிப்பு கொண்டிருக்காது என்பதை உறுதி செய்கிறது.

எ.கா:

Student அட்டவணையில் admno என்ற புலம் திரும்ப இருமுறை வராமல் பார்த்துக் கொள்கிறது.

CREATE TABLE Student

(Admno integer NOT NULL UNIQUE, → Unique constraint
Name char (20) NOT NULL,
Gender char (1),
Age integer,
Place char (10));

- Unique கட்டுப்பாட்டை NOT NULL என்று அறிவிக்கப்பட்ட புலங்களுக்கு மட்டுமே பயன்படுத்த முடியும்.

Primary key கட்டுப்பாடு

- ஒரு பதிவை தனித்தன்மையோடு அடையாளம் காட்ட பயன்படுகிறது.
- வெற்று மதிப்புகளை அனுமதிக்காததால் இந்த கட்டுப்பாடு NOT NULL கொண்டிருக்கும்.

எ.கா:

```
CREATE TABLE Student
```

```
(
```

```
Admno integer NOT NULL PRIMARY KEY, → Primary Key constraint
```

```
Name char(20) NOT NULL,
```

```
Gender char(1),
```

```
Age integer,
```

```
Place char(10),
```

```
);
```

Default கட்டுப்பாடு

- ஒரு புலத்தின் முன்னியல்பு மதிப்புகளை இருத்தி வைக்க பயன்படுகிறது.
- குறிப்பிட்ட புலத்திற்கு மதிப்பு கொடுக்கப்படவில்லை எனில் தானாகவே இருத்தப்படும்.

எ.கா:

```
CREATE TABLE Student
```

```
(
```

```
Admno integer NOT NULL PRIMARY KEY,
```

```
Name char(20) NOT NULL,
```

```
Gender char(1),
```

```
Age integer DEFAULT = "17", → Default Constraint
```

```
Place char(10),
```

```
);
```

Check கட்டுப்பாடு

- ஒப்பீடு மற்றும் தடுக்க செயற்குறிகளின் நிபந்தனையை வரையறுக்க பயன்படுகிறது.
- ஒரு புலத்திற்கு மதிப்பு வரையறுப்பதால் வரையறுக்கப்பட்ட மதிப்புகளை மட்டுமே கொடுக்க முடியும்.

எ.கா:

```
CREATE TABLE Student
```

```
(
```

```
Admno integer NOT NULL PRIMARY KEY
```

```
Name char(20) NOT NULL,
```

```
Gender char(1),
```

```
Age integer (CHECK<=19), → Check Constraint
```

```
Place char(10),
```

```
);
```


Table கட்டுப்பாடு

- ஒரு அட்டவணையில் உள்ள புலங்களின் குழுவிற்கு கட்டுப்பாட்டை பயன்படுத்துவதாகும்.
- Create table student(...) என்பதன் மூலம் ஒரு அட்டவணையை உருவாக்கலாம்.

எ.கா:

```
CREATE TABLE Student1
```

```
(
```

```
Admno integer NOT NULL,
```

```
Firstname char(20),
```

```
Lastname char(20),
```

```
Gender char(1),
```

```
Age integer,
```

```
Place char(10),
```

```
PRIMARY KEY (Firstname, Lastname) → Table constraint
```

```
);
```

மதிப்பீடு

1. எந்த கட்டளைகள் அட்டவணை வடிவமைப்பை உருவாக்குதல், உறவுநிலையை நீக்குதல் மற்றும் உறவுநிலை திட்ட வடிவமைப்பை மாற்றித் தர போன்ற செயற்பாடுகளுக்கான வரையறைகளை வழங்குகிறது?

அ) DDL ஆ) DML இ) DCL ஈ) DQL

DDL

2. எந்த கட்டளை அட்டவணையின் வடிவமைப்பை மாற்றி அமைக்க அனுமதிக்கிறது?

அ) SELECT ஆ) ORDER BY இ) MODIFY ஈ) ALTER

ALTER

3. அட்டவணையை நீக்க பயன்படுத்த வேண்டிய கட்டளை

அ) DROP ஆ) DELETE இ) DELETES ALL ஈ) ALTER TABLE

DROP

4. வினவல்களை உருவாக்க பயன்படுவது

அ) SELECT ஆ) ORDER BY இ) MODIFY ஈ) ALTER

SELECT

5. ஒரு தரவுத்தளத்தில் உள்ள தரவை வரிசையாக்கம் செய்ய பயன்படும் clause

அ) SORT BY ஆ) ORDER BY இ) GROUP BY ஈ) SELECT

ORDER BY

முக்கிய வினாக்கள்

1. 18 வயதிற்கும் குறைவாக உள்ள அனைத்து மாணவர்களின் தரவினை வரிசைப்படி தெரிவு செய்யும் ஒரு வினவலை எழுதுக.
2. அட்டவணை கட்டுப்பாட்டிற்கும், நெடுவரிசை கட்டுப்பாட்டிற்கும் உள்ள வேறுபாடுகளை எழுதுக.
3. எந்த SQL கூறு, அட்டவணையை உருவாக்கவும் அவற்றில் மதிப்பை சேர்க்கவும் அனுமதிக்கும்?
4. SQL மற்றும் MySQL க்கு உள்ள வேறுபாடுகள் யாவை?
5. கட்டுப்பாடு என்றால் என்ன? Primary Key கட்டுப்பாடு பற்றி சிறுகுறிப்பு வரைக.
6. ஒரு புதிய புலத்தை சேர்ப்பதன் மூலம் மாணவர் அட்டவணை கட்டமைப்பை மாற்றி அமைக்க ஒரு SQL கூற்றை எழுதுக.
7. ஏதேனும் மூன்று DDL கட்டளைகளை எழுதுக
8. Savepoint கட்டளையின் பயன்பாட்டை ஒரு எடுத்துக்காட்டுடன் எழுதுக.
9. DISTINCT சிறப்புச் சொல்லை பயன்படுத்தி ஒரு SQL கூற்றினை எழுதுக.
10. SQL கூறுகள் யாவை? ஒவ்வொன்றிற்கும் கட்டளைகளை எழுதுக.

நன்றி!

இளமைக் காலத்தில்
கல்வியை புறக்கணித்தவன்
எதிர்கால வாழ்வை
இழந்தவன் ஆகிறான்.
கல்வியின் மேன்மை
உணர்ந்து அக்கல்வியைப்
பெற்று சிறப்போடு வாழ
வாழ்த்துக்கள்.



ஜெ. கவிதா B.Sc, B.Ed, M.C.A, M.Phil.,
கணினி பயிற்றுநர் நிலை - I
அரசு மேல்நிலைப்பள்ளி,
சர்க்கார்சாமக்குளம்,
கோயம்புத்தூர் - 641107.