

- Database = Data + base
= উপাত্ত + ষাঁটি

ডেটা (Data) ও তথ্য (Information) কি ?

4

- Latin শব্দ Dataum এর বহুবচন হল Data.
- Datum শব্দের অর্থ হচ্ছে তথ্যের উপাদান।



Roll	Name	Gender	Age	GPA	City
101	Rahim	Male	18	3.44	Sylhet
102	Hasina	Female	17	4.50	Dhaka

সাজানো ও অর্থবোধক উপাত্তকে তথ্য বলে।
Processed or meaningful data = information

সাজানো ও অর্থবোধক নয় এমন কিছু fact/ঘটনাকে উপাত্ত বলে।
unprocessed or meaningless fact = data

© 2014
Digitized by srujanika@gmail.com

উপাত্ত / Data	তথ্য / Information
১. সাজানো ও অর্থবোধক নয় এমন কিছু fact/ঘটনাকে উপাত্ত বলে।	১. সাজানো ও অর্থবোধক উপাত্তকে তথ্য বলে।
২. উপাত্ত কোন বিষয় সম্পর্কে আংশিক ধারণা।	২. তথ্য কোন বিষয় সম্পর্কে সম্পূর্ণ ধারণা।
৩. উপাত্তকে সরাসরি ব্যবহার করা যায়না।	৩. তথ্যকে সরাসরি ব্যবহার করা যায়।
৪. উদাহরণ :	৪. উদাহরণ :

ডেটাবেজ কি ?

- Database = Data + base
= উপাত্ত + স্টোর
- পরম্পর সম্পর্কযুক্ত এক বা একাধিক ডেটা টেবিল বা ফাইলের সমষ্টি হচ্ছে ডেটাবেজ।



ডেটাবেজ ব্যবহার এর সুবিধা :

1. দ্রুত ডেটা উন্মত্তাপন করা যায়।
2. সহজে ডেটা বা রেকর্ড খুঁজে পাওয়া যায়।
3. সহজে ডেটা আপডেট করা যায়।
4. ডেটাকে নিরাপদভাবে সংরক্ষণ করে রাখে।
5. একসাথে অনেকেই ব্যবহার করতে পারেন।

ডেটাবেজ ব্যবহার এর অসুবিধা :

1. ডেটাবেজ এর নিরাপত্তা ব্যবস্থা দুর্বল হলে আর্থিক ক্ষতি হতে পারে।
2. ডেটাবেজ পরিচালনার জন্য দক্ষ জনশক্তির দরকার।

আলোচ্য বিষয়সমূহ

- ডেটাবেজ এর প্রকারভেদ সম্পর্কে আলোচনা কর।
- ডেটাবেজ এর বিভিন্ন উপাদানসমূহ সম্পর্কে আলোচনা কর।
- ফিল্ড ও রেকর্ডের পার্থক্য কি কি ?

ডেটাবেজ এর প্রকারভেদ সম্পর্কে আলোচনা কর।

- গঠনগত দিক থেকে ডেটাবেজ ২ প্রকার।
- ১. সাধারণ ডেটাবেজ
- ২. রিলেশনাল ডেটাবেজ

১. সাধারণ ডেটাবেজ

- শুধুমাত্র একটি ফাইল বা পরম্পর সম্পর্কহীন একাধিক ফাইলের সাহায্যে যে ডেটাবেজ গঠন করা হয়, তাকে সাধারণ ডেটাবেজ বলে।

Roll	Name	Gender	Age	GPA	City
101	Rahim	Male	18	3.44	Sylhet
102	Hasina	Female	17	4.50	Dhaka

২. রিলেশনাল ডেটাবেজ

- পরস্পর সম্পর্কযুক্ত একাধিক ফাইলের / টেবিলের সাহায্যে যে ডেটাবেজ গঠন করা হয়, তাকে রিলেশনাল ডেটাবেজ বলে।

Student				ExamResult	
Roll	Name	Gender	Age	Roll	GPA
101	Rahim	Male	18	101	3.44
102	Hasina	Female	17	102	4.50

ডেটাবেজ এর বিভিন্ন উপাদানসমূহ

- ডেটাবেজ এর উপাদানসমূহ
- 1. Field
- 2. Record
- 3. Value (data)

Field = column
Record = row

Roll	Name	Gender	Age
101	Rahim	Male	18
102	Hasina	Female	17

Student

Field = column

Record = row

Roll	Name	Gender	Age
101	Rahim	Male	18 ↗
102	Hasina	Female	17

ফিল্ড	রেকর্ড
১. ফিল্ড হল টেবিলের কলাম।	১. রেকর্ড হল টেবিলের সারি।
২. ফিল্ড দ্বারা পরিপূর্ণ তথ্য পাওয়া যায়না।	২. রেকর্ড দ্বারা পরিপূর্ণ তথ্য পাওয়া যায়।
৩. একটি ফিল্ডে একই ধরণের ডাটা থাকে।	৩. একটি রেকর্ডে বিভিন্ন ধরণের ডাটা থাকে।

Student

আলোচ্য বিষয়সমূহ

- কী ফিল্ড (key field) কি ?
- কী সমূহ এর প্রকারভেদ সম্পর্কে আলোচনা কর।
- প্রাইমারী কী ও ফরেন কী এর মধ্যে পার্থক্য কি কি ?

Roll	Name	Gender	Age↑
101	Rahim	Male	18
102	Farjana	Female	17
103	Mahfuz	Male	18
104	Farjana	Female	17

কী (Key) ফিল্ড ?

যে ফিল্ডের উপর ভিত্তি করে ডেটাবেজের রেকর্ড
সনাক্ত, অনুসন্ধান, সম্পর্ক তৈরি করা যায় তাকে কী
ফিল্ড বলে।

student

কী সমূহ এর প্রকারভেদ

- Key ও প্রকার ।



Primary Key

Primary key

Roll	Name	Gender	Age
101	Rahim	Male	18
102	Farjana	Female	17
103	Mahfuz	Male	18
104	Farjana	Female	17

student

প্রাইমারী কী (Primary Key) ?

যে ফিল্ডের সাহায্যে প্রতিটি রেকর্ডকে পৃথকভাবে
সনাক্ত করা যায়, তাকে প্রাইমারী কী বলে ।

Composite Key

Composite key

Roll	Name	Gender	Section
101	Rahim	Male	A
102	Farjana	Female	A
101	Mahfuz	Male	B
102	Farjana	Female	B

student

কম্পোজিট কী (Composite Key) ?

অনেক সময় একটি ফিল্ডের সাহায্যে প্রতিটি রেকর্ডকে
পৃথকভাবে সনাক্ত করা যায়ন।

তখন দুই বা ততোধিক কী ফিল্ডের সাহায্যে
কম্পোজিট কী তৈরি করে প্রতিটি রেকর্ডকে পৃথকভাবে
সনাক্ত করা যায়।

Foreign Key

ফরেন কী (Foreign Key) ?

একটি টেবিলের প্রাইমেরী কী যদি অন্য টেবিলের
সাধারণ কী হিসাবে ব্যবহার হয়, তখন সেই সাধারণ
কী কে ফরেন কী বলে।

Primary key

Roll	Name	Gender	Age
101	Rahim	Male	18
102	Farjana	Female	17
103	Mahfuz	Male	18
104	Farjana	Female	17

student

Primary key Foreign key

RegNumber	Roll	GPA
101112	101	3.25
102113	102	3.44
103114	101	4
104115	102	5

ExamResult

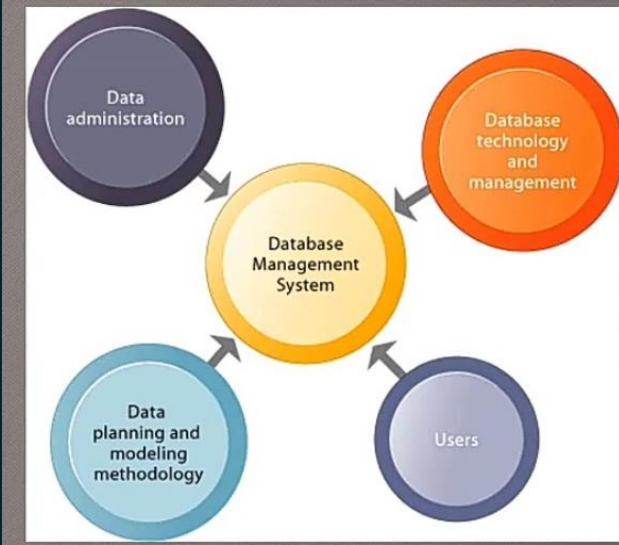
প্রাইমারী কী ও ফরেন কী এর মধ্যে পার্থক্য

প্রাইমারী কী	ফরেন কী
১) যে ফিল্ডের সাহায্যে প্রতিটি রেকর্ডকে পৃথকভাবে সনাক্ত করা যায়, তাকে প্রাইমারী কী বলে ।	১) একটি টেবিলের প্রাইমারী কী যদি অন্য টেবিলের সাধারণ কী হিসাবে ব্যবহার হয়, তখন সেই সাধারণ কী কে ফরেন কী বলে ।
২) ডেটা সনাক্তকরণে ব্যবহার করা হয় ।	২) দুইটি টেবিলের মধ্যে সম্পর্ক স্থাপনের ক্ষেত্রে ব্যবহার করা হয় ।
৩) এটি parent .	৩) এটি child.
৪) একটি টেবিলে একটিমাত্র প্রাইমারী কী থাকে ।	৪) একটি টেবিলে একের অধিক ফরেন কী থাকতে পারে ।

আলোচ্য বিষয়সমূহ

- ডেটাবেজ ম্যানেজমেন্ট সিস্টেম (DBMS) কী?
- ডেটাবেজ ম্যানেজমেন্ট সিস্টেমের উপাদান কি কি?
- ডেটাবেজ ম্যানেজমেন্ট সিস্টেম এর সুবিধা ও অসুবিধা লিখ ।
- ডেটাবেজ ম্যানেজমেন্ট সিস্টেম এর কাজ কি কি ?
- ডেটাবেজ ম্যানেজমেন্ট সিস্টেমের প্রয়োগ
- DATA View

ডেটাবেজ ম্যানেজমেন্ট সিস্টেমের উপাদান কি কি?



1. ডেটা
2. হার্ডওয়্যার ও সফটওয়্যার
3. ব্যবহারকারী
4. পদ্ধতি বা প্রক্রিয়া

ডেটাবেজ ম্যানেজমেন্ট সিস্টেম এর সুবিধা ও অসুবিধা লিখ

সুবিধা

- ১। সুন্দরভাবে ডেটা নিয়ন্ত্রণ করা যায়।
- ২। ডেটা খুঁজে পাওয়া সহজ হয়।
- ৩। ডেটাবেজের নিরাপত্তা প্রদান করে।
- ৪। বহু ব্যবহারকারী একসাথে ব্যবহার করতে পারে।

অসুবিধা

- ১। দক্ষ জনশক্তির প্রয়োজন।
- ২। বাস্তবায়ন ও পরিচালনা করা বেশ ব্যবহৃল।

ডেটাবেজ ম্যানেজমেন্ট সিস্টেম এর কাজ কি কি ?

1. CREATE DATABASE
2. UPDATE DATABASE
3. PRINT DATABASE
4. BACKUP AND RECOVERY DATABASE
5. SAVE DATA
6. RETRIEVE DATA
7. DELETE DATA
8. SECURE DATA etc.

ডেটাবেজ ম্যানেজমেন্ট সিস্টেমের প্রয়োগ

1. বিমান ও রেলওয়ে
2. ব্যাংকিং
3. শিক্ষা প্রতিষ্ঠান
4. মানব সম্পদ
5. ই-কমার্স

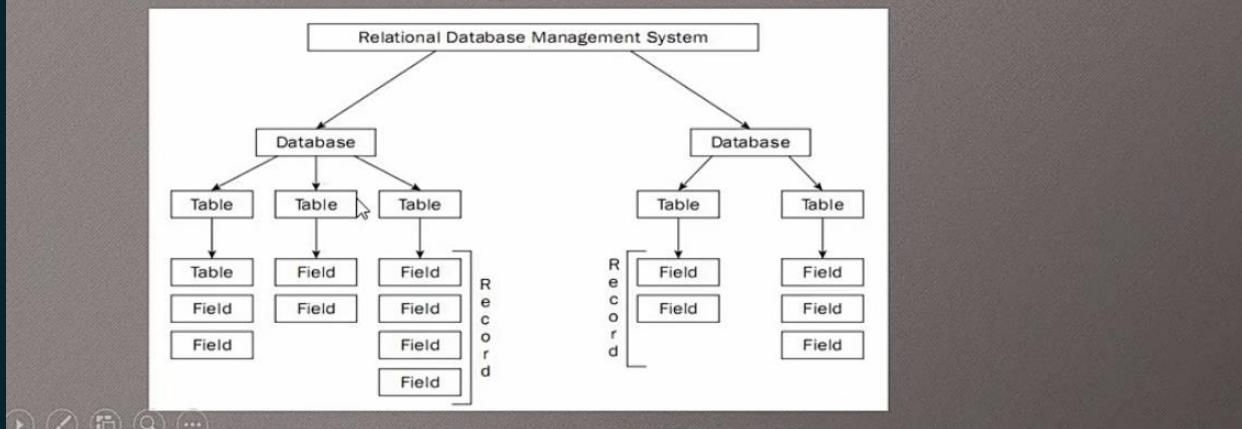
আলোচ্য বিষয়সমূহ

1. রিলেশনাল ডেটাবেজ ম্যানেজমেন্ট সিস্টেম (RDBMS) কী?
2. রিলেশনাল ডেটাবেজ ম্যানেজমেন্ট সিস্টেম বৈশিষ্ট্য কী?
3. রিলেশনাল ডেটাবেজ ম্যানেজমেন্ট সিস্টেম এর ব্যবহার
4. RDBMS VS DBMS

রিলেশনাল ডেটাবেজ ম্যানেজমেন্ট সিস্টেম (RDBMS) কী?

RDBMS = Relational DataBase Management System

1. রিলেশনাল ডেটাবেজ হচ্ছে বহু ব্যবহৃত একটি ডেটাবেজ।
2. এক বা একাধিক সম্পর্কযুক্ত টেবিল থাকে।



রিলেশনাল ডেটাবেজ ম্যানেজমেন্ট সিস্টেম বৈশিষ্ট্য কী?

1. সহজে বিভিন্ন টেবিল তৈরি করে ডাটা রাখা যায়।
2. ডেটাবেজের বিভিন্ন টেবিলের মধ্যে সহজেই সম্পর্ক তৈরি করা যায়।
3. টেবিলে থাকা তথ্য খুব সহজেই খুঁজে বের করা যায়।
4. সহজে এক ডেটাবেজ থেকে অন্য ডেটাবেজে তথ্য আদান-প্রদান করা যায়।

| ↴

রিলেশনাল ডেটাবেজ ম্যানেজমেন্ট সিস্টেম এর ব্যবহার

1. বিমান ও রেলওয়ে
2. ব্যাংকিং
3. শিক্ষা প্রতিষ্ঠান
4. মানব সম্পদ
5. ই-কমার্স

↳

RDBMS VS DBMS

	DBMS	RDBMS
1.	DBMS applications store data as file .	RDBMS applications store data in a tabular form .
2.	In DBMS, data is generally stored in either a hierarchical form or a navigational form.	In RDBMS, the tables have an identifier called primary key and the data values are stored in the form of tables.
3.	Normalization is not present in DBMS.	Normalization is present in RDBMS.
4.	DBMS does not apply any security with regards to data manipulation.	RDBMS defines the integrity constraint for the purpose of ACID (Atomicity, Consistency, Isolation and Durability) property.
5.	DBMS uses file system to store data, so there will be no relation between the tables .	in RDBMS, data values are stored in the form of tables, so a relationship between these data values will be stored in the form of a table as well.
6.	DBMS has to provide some uniform methods to access the stored information.	RDBMS system supports a tabular structure of the data and a relationship between them to access the stored information.
7.	DBMS does not support distributed database .	RDBMS supports distributed database .
8.	DBMS is meant to be for small organization and deal with small data . it supports single user .	RDBMS is designed to handle large amount of data . it supports multiple users .
9.	Data Redundancy is common in this model leading to difficulty in maintaining the data.	Keys and indexes are used in the tables to avoid redundancy.
10.	Example DBMS are dBase, Microsoft Access, LibreOffice Base, FoxPro.	Example RDBMS are SQL Server, Oracle , MySQL, Maria DB, SQLite.

আলোচ্য বিষয়সমূহ

- ডেটাবেজ রিলেশন কী?
- ডেটাবেজ রিলেশন এর প্রকারভেদ।

ডেটাবেজ রিলেশন কী?

- দুই বা ততোধিক টেবিলের মধ্যে যে পদ্ধতিতে সম্পর্ক স্থাপন করে টেবিলের উপর বিভিন্ন কুয়েরি চালানো হয় তাকে ডেটাবেজ রিলেশন বলে।
- টেবিলগুলোর মধ্যে কিভাবে রেকর্ডগুলো সম্পর্কযুক্ত হবে তার উপর নির্ভর করে ডেটাবেজ রিলেশনকে ৩ ভাগে ভাগ করা যায়।
 - One-To-One
 - One-To-Many
 - Many-To-Many

ডেটাবেজ রিলেশন তৈরির শর্ত

- রিলেশনাল ডেটা টেবিলগুলির মধ্যে কমপক্ষে একটি কমন ফিল্ড থাকতে হবে এবং কমন ফিল্ডের ডেটা টাইপ, ফিল্ড সাইজ ইত্যাদি একই হতে হবে।
- রিলেশনাল টেবিলগুলোর মধ্যে অন্তত একটি টেবিলে অবশ্যই প্রাইমারী কি ফিল্ড থাকতে হবে।

Roll	Name	Group
101	Ayesha	Science
102	Tamim	Science
103	Ayesha	Arts
104	Lipu	Science
105	Rabeya	Arts

⌚ ⏱️ 📑 Student Table

Reg No.	Roll	GPA	Section
100101	101	4.5	A
100102	102	4.5	A
100103	103	5	B
100104	104	4.7	A
100105	105	5	B

Result Table

One-To-One Database Relation

Table - A

Roll	Name	Group
101	Ayesha	Science
102	Tamim	Science
103	Ayesha	Arts

Table - B

Roll	GPA	Section
101	4.5	A
102	4.5	A
103	5	B

১। কোন একটি টেবিলের একটি রেকর্ডের সাথে যথন অন্য টেবিলের কেবল একটি রেকর্ডের সম্পর্ক স্থাপন করা হয়, তখন তাকে One-To-One Relation বলে ।

২। খুব বেশী ব্যবহার করা হয়না ।

৩। প্রাইমারী কি এর সাথে প্রাইমারী কি এর সম্পর্ক স্থাপন করা হয় ।

৪। টেবিলের ডেটার গোপনীয়তা রক্ষা করা যায় ।



One-To-Many (Many-To-One) Database Relation

Table - A

Roll	Name
101	Ayesha
102	Tamim
103	Ayesha

১। কোন একটি টেবিলের একটি রেকর্ডের সাথে যথন অন্য টেবিলের একাধিক রেকর্ডের সম্পর্ক স্থাপন করা হয়, তখন তাকে One-To-Many Relation বলে ।

২। সবচেয়ে বেশী ব্যবহার করা হয়।

৩। একটি টেবিলের প্রাইমারী কি এর সাথে অন্য টেবিলের ফরেন কি এর সম্পর্ক হলে এই ধরণের সম্পর্ক তৈরি হয় ।

Roll	Subject_ID	Subject_Title	Credit
101	C1001	English	3
101	C1002	Bangla	3
102	C1003	Math	3
103	C1004	ICT	3

Table - B



Many-To-Many Database Relation

Teacher Table		Course Table	
Teacher_ID	Teacher_Name	Course_ID	Subject_Name
T01	Rumel	C01	English
T02	Sohid	C02	Bangla
T03	Alak	C03	Math

Teacher_ID	Course_ID	Day
T01	C01	Mon
T01	C02	Sat
T02	C03	Tue
T03	C01	Mon

Routine Table

1। দুটি টেবিলের মধ্যে যদি একাধিক রেকর্ডের ম্যাচিং থাকে, তখন তাকে Many-To-Many Relation বলে।

Table - A  Table - B

2। এই রিলেশন তৈরির জন্য অতিরিক্ত একটি টেবিলের প্রয়োজন হয়, যাকে জংশন টেবিল বলে।

3। দুটি টেবিলের প্রাইমারী কি জংশন টেবিলের ফরেন কি হিসাবে ব্যবহার করা হয় এই ধরণের সম্পর্কে।

Sorting

- Sorting অর্থ সাজানো।
 - Sorting কি ধরণের।
 - Ascending (উর্ধ্বক্রম)
 - Descending (নিম্নক্রম)
- Shain, anis, bijoy, zakir
- Anis, bijoy, shain, zakir
- Zakir, Shain, bijoy, anis

মেমো, হাইপারলিঙ্ক, OLE ফিল্ডে সর্টিং হয়না।



Sorting

Ascending (উর্ধ্ব:ক্রম)			Descending (অধঃক্রম)		
Roll	Name	GPA	Roll	Name	GPA
101	Shewly	4.5	102	Karim	2.5
102	Karim	2.5	103	Lipu	3.7
103	Lipu	3.7	101	Shewly	4.5
104	Ayesha	4.90	104	Ayesha	4.90

Student_table Student_table Student_table

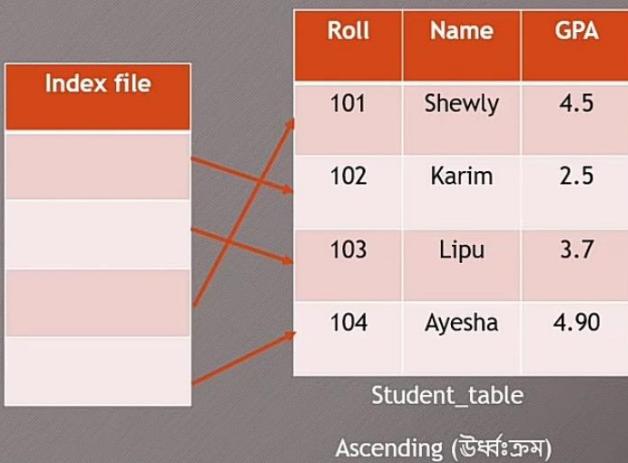
Sorting

Ascending (উর্ধ্ব:ক্রম)			Descending (অধঃক্রম)		
Roll	Name	GPA	Roll	Name	GPA
101	Shewly	4.5	102	Karim	2.5
102	Karim	2.5	103	Lipu	3.7
103	Lipu	3.7	101	Shewly	4.5
104	Ayesha	4.90			

Student_table Student_table

১। সটিং হল ডেটা টেবিলের রেকর্ডগুলোকে কোন নির্ধারিত ফিল্ট অনুসারে সাজানো ।
২। সটিং এর ফলে ডেটা খুঁজে (searching) বের করা সহজ হয় ।
৩। ডেটাবেজে আলাদা স্টেড ফাইল তৈরি হয় ।
৪। সটিং এর ফলে মূল ডেটা ফাইলে রেকর্ডের ক্রমিক নং পরিবর্তন হয় ।
৫। ডেটাবেজে কোন রেকর্ড সংশোধন / সংযোজন করলে সেট করা ফাইল আপডেট হয়না ।
৬। রেকর্ডগুলো সাজানোর জন্য তুলনামূলক ব্রেইন সময়ের প্রয়োজন হয় ।

Indexing



- ১। ইন্ডেক্স হল মূল টেবিল অপরিবর্তিত রেখে ডেটা টেবিলের রেকর্ডগুলোকে কোন নির্ধারিত ফিল্ট অনুসারে সাজানো ।
 - ২। ইন্ডেক্স ফাইলে কোন রেকর্ড থাকেনা, তবে লজিকাল অর্ডার থাকে ।
 - ৩। ইন্ডেক্স এর ফলে মূল ডেটা ফাইলে রেকর্ডের ক্রমিক নং পরিবর্তন হয় না।
 - ৫। ডেটাবেজে কোন রেকর্ড সংশোধন / সংযোজন করলে index করা ফাইল ব্যংক্রিয়ভাবে আপডেট হয়।
 - ৬। রেকর্ডগুলো সাজানোর জন্য তুলনামূলক কম সময়ের প্রয়োজন হয় ।

Sorting vs Indexing

Sorting	Indexing
১। সটিং হল ডেটা টেবিলের রেকর্ডগুলোকে কোন নির্ধারিত ফিল্ড অনুসারে সজানো ।	১। ইন্ডেক্সিং হল মূল টেবিল অপরিবর্তিত রেখে ডেটা টেবিলের রেকর্ডগুলোকে কোন নির্ধারিত ফিল্ড অনুসারে সজানো ।
২। সটিং এর ফলে মূল ডেটা ফাইলে রেকর্ডের ক্রমিক নং পরিবর্তন হয় ।	২। ইন্ডেক্সিং এর ফলে মূল ডেটা ফাইলে রেকর্ডের ক্রমিক নং পরিবর্তন হয় না।
৩। ডেটাবেজে কোন রেকর্ড সংশোধন / সংযোজন করলে সেট করা ফাইল আপডেট হয়না ।	৩। ডেটাবেজে কোন রেকর্ড সংশোধন / সংযোজন করলে index করা ফাইল আপডেট হয়।
৪। রেকর্ডগুলো সাজানোর জন্য তুলনামূলক বেশী সময়ের প্রয়োজন হয় ।	৪। রেকর্ডগুলো সাজানোর জন্য তুলনামূলক কম সময়ের প্রয়োজন হয় ।
৫। সটিং এ মেমোরি বেশী প্রয়োজন হয় ।	৫। ইন্ডেক্সিং এ মেমোরি কম প্রয়োজন হয় ।

আলোচ্য বিষয়সমূহ

- ডেটা সিকিউরিটি
- ডেটা এনক্রিপশন
- প্লেইনটেক্স্ট
- সাইফারটেক্স্ট

- ডেটা সিকিউরিটি কি?
- অনিদিষ্ট ব্যক্তিগুরু হাত থেকে ডেটাকে সুরক্ষা করার পদ্ধতিকে ডেটা সিকিউরিটি বলে।

Encryption

ডেটাকে এক জায়গা থেকে অন্য জায়গায় পাঠানোর পূর্বে মূল ফরম্যাট থেকে অন্য ফরম্যাটে রূপান্তর করার প্রক্রিয়াকে বলে এনক্রিপশন বলে।

- ডেটা এনক্রিপ্ট করার বিভিন্ন পদ্ধতি রয়েছে ।
 - যেমন - ১) সিজার কোড
 - ২) DES (Data Encryption Standard)
 - ৩) IDEA (International Data Encryption Algorithm)

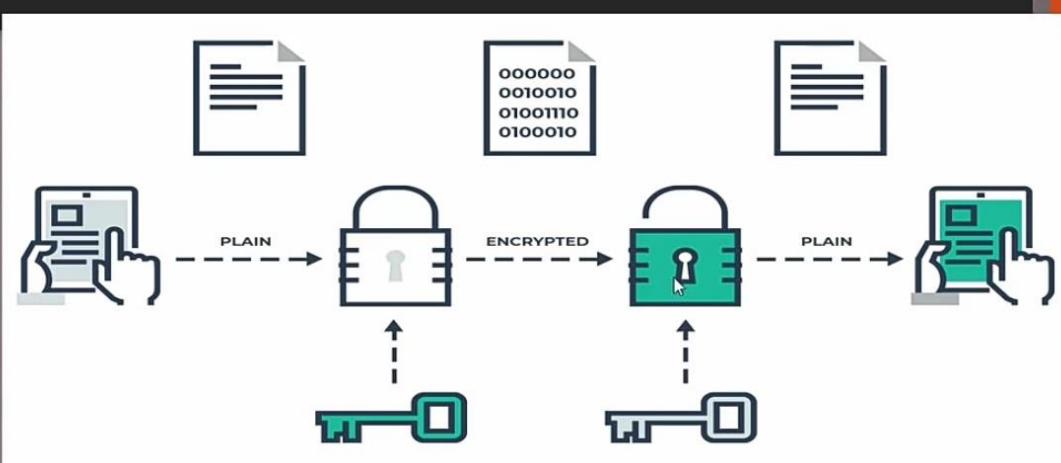
Caesar Cipher

The **Caesar Cipher**, also known as a shift cipher, is one of the oldest and simplest forms of encrypting a message. It is a type of substitution cipher where each letter in the original message (which in cryptography is called the plaintext) is replaced with a letter corresponding to a certain number of letters shifted up or down in the alphabet.

এনক্রিপশন এর উদাহরণ

- ডেটাকে এক জায়গা থেকে অন্য জায়গায় পাঠ্নোর পূর্বে মূল ফরম্যাট থেকে অন্য ফরম্যাটে রূপান্তর করার প্রক্রিয়াকে বলে এনক্রিপশন বলে ।
 - এনক্রিপ্ট করার পূর্বে ডেটা যা পাঠ করা যায়, তাকে প্লেইন টেক্স্ট (plain text) বলে ।
 - এনক্রিপ্ট করার পরের ডেটা যা সহজে মানুষের পাঠ্যযোগ্য রূপে থাকে না, তাকে সাইফার টেক্স্ট (Cipher text) বলে ।
 - কি (key) - গোপন কোড যা এনক্রিপ্ট বা ডিক্রিপ্ট করার জন্য প্রয়োজন ।

Encryption



ଡেটাকे ଏକ ଜୀବନା ଥିଲେ ଅଣ୍ ଜୀବନାଯ ପାଠାନୋର ମୂର୍ବେ ମୂଳ ଫରମ୍ୟାଟ ଥିଲେ ଅଣ୍ ଫରମ୍ୟାଟେ ରମ୍ପାତ୍ର କରାର ପ୍ରକ୍ରିୟାକେ ବଲେ ଏନ୍ଟରିପଶନ ବଲେ ।

এনক্রিপশন এর প্রকারভেদ

- সাধারণত ২ ধরণের এনক্রিপশন দেখা যায় ।
 - 1) গোপন কি এনক্রিপশন / সিমেট্রিক এনক্রিপশন
 - 2) পাবলিক কি এনক্রিপশন / অ্যাসিমেট্রিক এনক্রিপশন

আলোচ্য বিষয়সমূহ

1. কুয়েরি কি? কুয়েরির প্রকারভেদ আলোচনা কর ।
2. কুয়েরি ভাষা
3. SQL স্টেটমেন্ট এর বৈশিষ্ট্য, প্রকারভেদ
4. SQL এর ব্যবহার, সুবিধা ও অসুবিধা

1. কুয়েরি কি? কুয়েরির প্রকারভেদ আলোচনা কর ।

- ডেটাবেজে সংরক্ষিত বিপুল পরিমাণ তথ্য থেকে প্রযোজনীয় তথ্য খুঁজে বের করাকে কুয়োরি বলে ।
- বিভিন্ন ধরণের কুয়েরি রয়েছে ।
 - 1) সিলেক্ট কুয়েরি
 - 2) প্যারামিটার কুয়েরি
 - 3) ক্রসট্যাব কুয়েরি
 - 4) অ্যানম্যাচড কুয়েরি
 - 5) অ্যাকশন কুয়েরি

বিভিন্ন প্রকার কুয়েরি

- ডেটাবেজের ডেটা টেবিল থেকে ফিল্ড বা কলাম নির্বাচন করে যে কুয়েরি করা হয় তাকে সিলেক্ট কুয়েরি বলে।
- কুয়েরিকৃত ডেটাকে সামারি বা গ্রুপ আকারে উপস্থাপনের জন্য যে কুয়েরি পরিচালনা করা হয়, তাকে ক্রসট্যাব কুয়েরি বলে।
- কোন শর্তের উপর ভিত্তি করে একটি টেবিল এর রেকর্ড অন্য ডেটা টেবিলের সাথে সামঞ্জস্য না হলে unmatched ডেটাওলো খুঁজে বের করার জন্য যে কুয়েরি ব্যবহার করা হয়, তাকে Unmatched Query.
- কোন কুয়েরির ফলাফল দিয়ে যখন টেবিলের ডেটাসমূহের কোন পরিবর্তন সাধন করা হয়, তখন তাকে Action Query বলে।
 - Action Query চার প্রকার।
 - Make Table Query
 - Append Query (সংযোজন করা)
 - Delete Query
 - Update Query

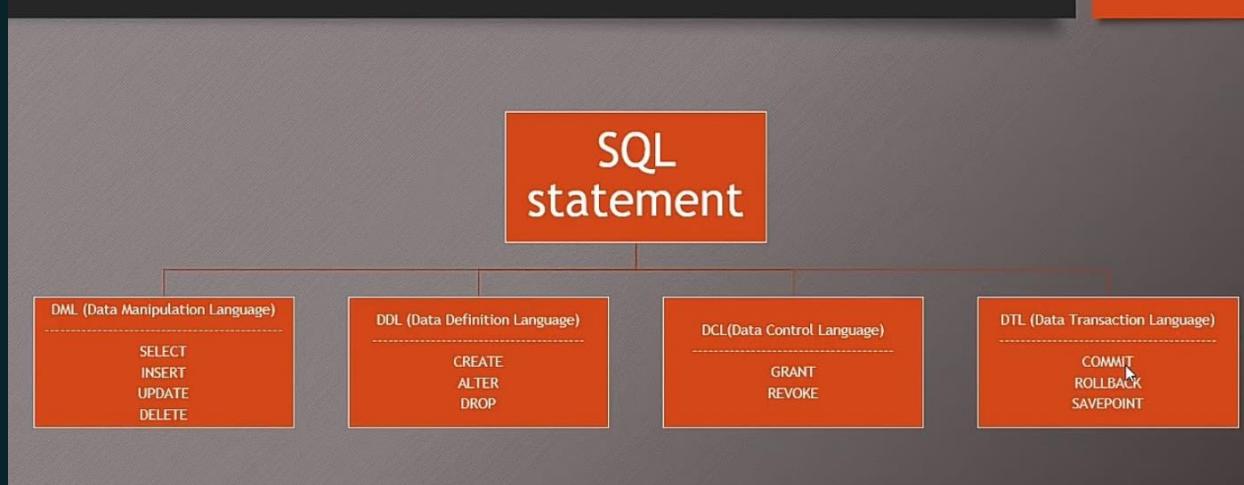
কুয়েরি ভাষা

- যে ভাষার সাহায্যে কুয়েরি করা হয়, তাকে কুয়েরি ভাষা বলে।
- ডেটাবেজ বহুল ব্যবহৃত ৩ টি কুয়েরি ভাষা হল
 - QUEL (Query Language)
 - QBE (Query By Example)
 - SQL (Structured Query Language)

SQL

- SQL স্টেটমেন্ট এর বৈশিষ্ট্য
 - I. কেস সেনসিটিভ নয় । SELECT / select
 - II. স্টেটমেন্ট এর শেষে সেমিকোলন (;) দিতে হয় ।
 - III. স্টেটমেন্টকে ভেঙ্গে একাধিক লাইনে লেখা যায় ।

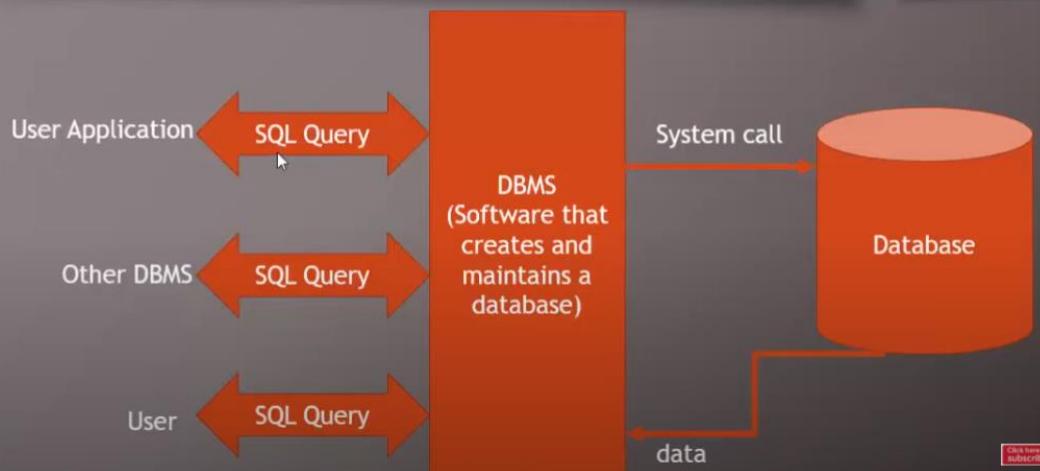
SQL Statement এর প্রকারভেদ



- DBMS = DataBase Management System
- DBMS = DB +MS
- A database management system (DBMS) is a software that enables users to create and maintain a database.
- some examples of popular DBMS are MySql, Oracle, Sybase, Microsoft Access and IBM DB2 etc.

Database Management system

3



SQL

4

- SQL = Structured Query Language
- কুয়েরি কি ?
 1. ডেটাবেজ থেকে রেকর্ড বা রেকর্ডসমূহকে খুঁজে বের করারকে কুয়েরি বলে।
 2. ডেটাবেজ এ ডাটা - Insert, derive, modify, delete করাকেই কুয়েরি বলে।
- কুয়েরি ল্যাঙ্গুয়েজ কি ?
- যে ভাষার সাহায্যে কুয়েরি করা হয়, তাকে কুয়েরি ল্যাঙ্গুয়েজ বলে।

- Sql statements are not case sensitive.
- There must be a semicolon end of a sql command.

- SQL STATEMENTS ->

- 1) DML : SELECT, INSERT, UPDATE, DELETE
- 2) DDL : CREATE, ALTER, DROP

So what we need next ?

6

- MySQL
- XAMPP is a free and open source cross-platform local web server developed by **Apache Friends**.
- XAMPP stands for **Cross-Platform (X)**, **Apache (A)**, **MySQL (M)**, **PHP (P)** and **Perl (P)**.

```
SHOW DATABASES;
```

```
CREATE DATABASE college;
```

```
DROP DATABASE college;
```

• Numeric data type	Roll	Name	Gender	Age	GPA	City
-----	101	Rahim	Male	18	3.44	Sylhet
• SMALLINT	102	Hasina	Female	17	4.50	Dhaka
• INT						
• BIGINT						
• FLOAT						
• DOUBLE(M,D)						
• DECIMAL(M,D)						
• Character Type	•	• Char(m)	• Varchar(m)	• text	• Data and Time data Type	• -----
-----					• YEAR (YYYY)	
					• DATE (YYYY-MM-DD)	
					• TIME(HH:MM:SS)	
• Some other data type	• -----	• -----	• -----	• -----	• -----	• -----
• -----						
• BLOB						
• Logical						

Creating a table

3

Roll	Name	Gender	Age	GPA	City	Roll	Name	Gender	Age	GPA	City
101	Rahim	Male	18	3.44	Sylhet						
102	Hasina	Female	17	4.50	Dhaka						

- টেবিল তৈরির জন্য CREATE TABLE statement
ব্যবহার করা হয়।

- Basic Syntax

```
CREATE TABLE table_name
(
column_name1 data_type(size),
column_name2 data_type(size),
column_name3 data_type(size),
...
columnN data_type(size)
);
```

```
CREATE TABLE student
(
    Roll int(5),
    Name varchar(20),
    Gender varchar(10),
    Age int(5),
    GPA double(3,2),
    City varchar(15),
    PRIMARY KEY (Roll)
);
```

- **RENAME command** এর মাধ্যমে টেবিলের নাম পরিবর্তন করতে পারবেন।

Syntax

```
RENAME TABLE old_name TO new_name;
```

Example :

```
RENAME TABLE student TO students;
```

```
CREATE TABLE students(
```

```
    Roll int,
```

```
    Name varchar(20),
```

```
    Gender varchar(10),
```

```
    Age int,
```

```
    GPA double(3, 2),
```

```
    City varchar(20),
```

```
    PRIMARY KEY(Roll)
```

```
);
```

Data Insertion

2

Roll	Name	Gender	Age	GPA	City	Roll	Name	Gender	Age	GPA	City
101	Rahim	Male	18	3.44	Sylhet						
102	Hasina	Female	17	4.50	Dhaka						

- টেবিল তৈরির পর Data insert করতে INSERT INTO statement ব্যবহার করা হয়।

- Syntax for data insertion**

```
INSERT INTO table_name (column1, column2, column3, ..., columnN)
VALUES (value1, value2, value3, ..., valueN);
```

```
INSERT INTO student_details (Roll, Name, Gender, Age, GPA, City)
VALUES (101, 'Rahim', 'Male', 18, 3.44, 'Sylhet');
```

Data Insertion

8

Roll	Name	Gender	Age	GPA	City	Roll	Name	Gender	Age	GPA	City
101	Rahim	Male	18	3.44	Sylhet	101	Rahim	Male	18	3.44	Sylhet
102	Hasina	Female	17	4.50	Dhaka	102	Hasina	Female	17	4.50	Dhaka

- টেবিল তৈরির পর Data insert করতে INSERT INTO statement ব্যবহার করা হয়।

- Syntax for data insertion**

```
INSERT INTO table_name
VALUES (value1, value2, value3, ...);
```

```
INSERT INTO student_details
VALUES
```

```
(101, 'Rahim', 'Male', 18, 3.44, 'Sylhet'),
(102, 'Hasina', 'Female', 18, 4.50, 'Dhaka');
```

- DDL -> CREATE, ALTER, DROP
- DML -> INSERT, SELECT, UPDATE, DELETE

- SELECT statement এর সাহায্যে টেবিল থেকে প্রযোজনীয় তথ্য খুঁজে পাওয়া যায়।

Syntax

```
-----  
SELECT column_list  
FROM table_name;
```

```
SHOW DATABASES;  
CREATE DATABASE college;  
DROP DATABASE college;  
RENAME TABLE student TO students;
```

```
//Inserting record into table  
INSERT INTO students  
(Roll, Name, Gender, Age, GPA, City)  
VALUES  
(101, 'Kibria', 'Male', 21, 3.50, 'Dhaka');
```

```
INSERT INTO students  
VALUES  
(103, 'Kibria', 'Male', 21, 3.50, 'Dhaka'),  
(104, 'Kibria', 'Male', 21, 3.50, 'Dhaka');
```

```
//select command  
SELECT Name  
FROM students;
```

```
-----  
SELECT Name, Age  
FROM students;
```

```
-----  
SELECT *  
FROM students;
```

```
-----  
SELECT DISTINCT City //distinct dara double value gulo bar bar dekaby na  
FROM students;
```

```
SELECT *
FROM students
LIMIT 3;
-----
SELECT *
FROM students
LIMIT 2, 2;
first 2 dara bujay first 2 ta row bad.
second 2 dara bujay porer 2 ta row output hisaby dekaby
-----
SELECT Name
FROM students
ORDER BY Name; //ascending akary output dekaby
-----
SELECT Name, Age, Gender
FROM students
ORDER BY Name;
-----
SELECT Name, Age, Gender
FROM students
ORDER BY Name DESC; //descending akary output dekaby
-----
SELECT Name, Age, Gender
FROM students
ORDER BY Name, Age; //multiple column
```

Arithmetic Operator

+

-

*

/

%



SELECT 5 + 2;

Aybaby Korty Hoby.

Comparison operator

=

!=

>

>=

<

<=

BETWEEN

Logical Operator

AND

OR

IN

NOT

3

Roll	Name	Gender	Age	GPA	City
101	Rahim	Male	18	3.44	Sylhet
102	Hasina	Female	17	4.50	Dhaka
103	Sabul	Male	18	3.70	Sylhet
104	Suhan	Male	17	4.50	Khulna
105	Rahim	Male	18	3.44	Barisal
106	Tanvir	Male	17	4.50	Sylhet
107	Hazira	Female	18	3.44	Sylhet
108	Hafiza	Female	17	4.50	Dhaka

Student_details

City
Dhaka
Sylhet
Dhaka

```
SELECT City
FROM student_deatils
WHERE Gender='Female';
```



2

- WHERE clause এর সাহায্যে একটি নির্দিষ্ট শর্ত /condition এর উপর ভিত্তি করে ডাটা খুঁজতে ব্যবহার করা হয়।

```
SELECT column_list
FROM table_name
WHERE condition;
```



```
SELECT City  
FROM students  
WHERE Gender = 'Male';
```

```
SELECT DISTINCT City  
FROM students  
WHERE Gender = 'Male';
```

```
SELECT *  
FROM students  
WHERE Age = 21;
```

Logical Operator

4

AND

OR

IN

NOT

LIKE

```
SELECT Roll  
FROM students  
WHERE Roll BETWEEN 101 AND 104;
```

```
SELECT Roll  
FROM students  
WHERE Roll != 104;
```

```
SELECT *  
FROM students  
WHERE Gender = 'Male' OR GPA >= 3.50;
```

```
SELECT *  
FROM students  
WHERE City = 'Dhaka' AND (Gender = 'Male' OR GPA >= 3.5);
```

```
SELECT *  
FROM students  
WHERE City = 'Dhaka' OR City = 'Cumilla' OR City = 'Bd';
```

```
SELECT *  
FROM students  
WHERE City IN ('Dhaka', 'Bd');
```

```
SELECT *  
FROM students  
WHERE City NOT IN ('Dhaka', 'Bd');
```

```
SELECT *
FROM students
WHERE Name LIKE 'k%';
//% dara bujay k er por ki asy oita niye matha betha nai.

//%k dile ki hoby? tokon nam er sesy jodi k thaki taholy oita print korby.
-----
SELECT *
FROM students
WHERE Name LIKE '%ki%';
-----
SELECT *
FROM students
WHERE Name LIKE '_ i%';
//underscore dara bujay first character ki hoby seta matter na. kintu second a i thakly seta print hoby.
-----
SELECT *
FROM students
WHERE Name LIKE '%i_';
//ata dara bujay seser dik theky suru kory second ta jodi i hoy toby print hoby.
-----
SELECT Roll AS ID, Name AS fName
FROM students;
-----
SELECT Roll AS ID, Name AS 'First Name'
FROM students;
//dui ta word use korychi tai single quotation er moddy rekhychi.
```

SQL Constraints

2

SQL constraints are used to specify rules for table data.

1. NOT NULL
2. UNIQUE - Does not allow to insert a duplicate value in a column.
4. PRIMARY KEY = NOT NULL + UNIQUE
5. CHECK - Determines whether the value is valid or not from a logical expression.
6. DEFAULT - While inserting data into a table, if no value is supplied to a column, then the column gets the value set as DEFAULT.



2

```
UPDATE table_name  
SET column1=value1, column2=value2, ...  
WHERE condition;
```

```
UPDATE teacher  
SET Salary=32000  
WHERE ID=1002;
```

ID	Name	Salary
1000	Alak kanti shar	35000.00
1001	Rumel M S PIR	45000.00
1002	Saiful Ambia	NULL
1003	Rubel	9000.00
1004	Rayhan	12000.00
1005	Siraj	19000.00

teacher



3

```
DELETE FROM table_name  
WHERE condition;
```

```
DELETE FROM teacher  
WHERE ID=1005;
```

ID	Name	Salary
1000	Alak kanti shar	35000.00
1001	Rumel M S PIR	45000.00
1002	Saiful Ambia	NULL
1003	Rubel	9000.00
1004	Rayhan	12000.00
1005	Siraj	19000.00

teacher



2

টেবিল থেকে যে কোন একটা রেকর্ড বা সবগুলো রেকর্ড মুছে ফেলার জন্য
DELETE statement ব্যবহার করা হয়।



- UPPER()
- LOWER()
- CONCAT()
- POW()
- GREATEST()
- LEAST()
- LOG()
- LOG10()
- EXP()
- RAND()

```
CREATE TABLE teacher
(
    ID int,
    Name varchar(15)
);
INSERT INTO teacher(ID, Name)
VALUES(10, 'Kibria');
INSERT INTO teacher(Name)
VALUES('Ezaz');
//akhetry idr ghor ta NULL dekaby.
```

```
CREATE TABLE teacher
(
    ID int NOT NULL AUTO_INCREMENT,
    Name varchar(15) NOT NULL,
    Salary double(10, 2),
    PRIMARY KEY(ID)
);
INSERT INTO teacher(ID, Name, Salary)
VALUES(10, 'Kibria', 500);
INSERT INTO teacher(Name, Salary)
VALUES('Ezaz', 600);
//akhetry ID na dileo auto increment hoye jaby.
```

```
UPDATE teacher
SET Salary = 300
WHERE ID = 15;
//aybabu name o update kora jaby.
```

```
DELETE FROM teacher
```

```
WHERE id = 12;
```

```
//WHERE id > 12;
```

```
UPDATE teacher
```

```
SET Salary = Salary + 1000
```

```
WHERE Salary > 500;
```

```
SELECT UPPER('Golam kibria');
```

```
SELECT LOWER('Golam kibria');
```

```
SELECT UPPER(City)
```

```
FROM students;
```

```
//oi j students table er city column taky upper a convert kory diby.
```

```
SELECT UPPER(City) AS Upper_City
```

```
FROM students;
```

```
//AS er maddomy just column er nam ta change korlam.
```

```
SELECT concat('Golam', ' Kibria');
```

```
or
```

```
SELECT concat(Name, ' is ', Age, ' years old')
```

```
FROM students; //students table hoty sober info dekaby.
```

```
or
```

```
SELECT concat(Name, ' is ', Age, ' years old') AS INFO
```

```
FROM students;
```

```
SELECT greatest(1, 2, 12, 21, 23);
```

```
SELECT least(1, 2, 12, 21, 23);
```

SELECT power(2, 3);

SELECT log(2);

SELECT log10(2);

SELECT TRUNCATE(12.1232434, 2);

dosomik er por dui ghor dekaby

SELECT TRUNCATE(log(2), 2);

SELECT rand();

SELECT exp(3); //orthat e to the power 3 er man

- Group functions operate on sets of rows to give one result per group.

AVG

COUNT

MAX

MIN

SUM



ALTER TABLE

- **ALTER TABLE** command এর সাহায্যে

1. টেবিলে নতুন কলাম যুক্ত করা যায়।
2. কোন কলামের নাম পরিবর্তন করা যায়।
3. কোন কুলাম delete করা যায়।



Adding a column (টেবিলে নতুন কলাম যুক্ত করা)

3

- syntax
- **ALTER TABLE table_name**
ADD column_name datatype[size];
- **ALTER TABLE student_details**
ADD Phone text(20);



Renaming column (কোন কলামের নাম পরিবর্তন করা)

4

Syntax

- **ALTER TABLE table_name**
CHANGE oldcolumn_name newcolumn_name dataType(size);
- **ALTER TABLE student_details**
CHANGE phone phone_number text(15);



Dropping a column (কলাম delete করা)

5

- Syntax
- **ALTER TABLE table_name
DROP COLUMN column_name;**
- **ALTER TABLE student_details
DROP COLUMN phone_number;**

```
DESCRIBE teacher; //table a ki ki asy ta describe kory
```

```
-----  
SELECT COUNT(*)  
FROM students;  
//kotogulo row asy seta print kory dekaby.
```

```
-----  
SELECT max(GPA)  
FROM students;  
-----  
SELECT min(GPA)  
FROM students;
```

```
-----  
SELECT SUM(Salary), AVG(Salary)  
FROM teacher;
```

```
-----  
SELECT COUNT(*), SUM(Salary), AVG(Salary), MAX(Salary), MIN(Salary)  
FROM teacher;
```

```
-----  
SELECT MIN(GPA)  
FROM students  
WHERE Gender = 'Male';
```

```
-----  
SELECT Roll, Name, Age, City, MIN(GPA)  
FROM students  
WHERE Gender = 'Male';
```

```
-----  
SELECT *  
FROM teacher  
WHERE Salary > 1180.000000;
```

```
-----  
SELECT *  
FROM teacher  
WHERE Salary > (SELECT AVG(Salary) FROM teacher);  
//atai holo subquery. cuz aktar moddy ar akta query use korychi.
```

```
ALTER TABLE teacher  
ADD Age int;  
or,  
ALTER TABLE teacher  
ADD Dept varchar(15);
```

```
ALTER TABLE teacher  
CHANGE Dept Department varchar(15);  
//rename korar system
```

```
ALTER TABLE teacher  
DROP COLUMN Age;  
or  
ALTER TABLE teacher  
DROP COLUMNS (Age, Department);
```

```
UPDATE teacher  
SET Department = 'CSE'  
WHERE ID = 11;
```

You can divide rows in a table into smaller groups by using the GROUP BY clause.

ID	Name	Salary	Department
1000	Alak kanti shar	35000.00	CSE
1001	Rumel M S PIR	45000.00	EEE
1002	Saiful Ambia	32000.00	CSE
1003	Rubel	9000.00	EEE
1006	Selina	32000.00	CSE
1007	Asad	38000.00	EEE
1008	Sohid	33500.00	BBA
1009	Alin	33000.00	BBA
1010	Dola Barua	32000.00	CSE



Creating Groups of Data: GROUP BY Clause Syntax

```
SELECT      column, group_function(column)
FROM        table
[WHERE      condition]
[GROUP BY  group_by_expression]
[ORDER BY  column];
```

```
SELECT Department,SUM(Salary)
FROM teacher
GROUP BY Department;
```

ID	Name	Salary	Department
1000	Alak kanti shar	35000.00	CSE
1001	Rumel M S PIR	45000.00	EEE
1002	Saiful Ambia	32000.00	CSE
1003	Rubel	9000.00	EEE
1006	Selina	32000.00	CSE
1007	Asad	38000.00	EEE
1008	Sohid	33500.00	BBA
1009	Alin	33000.00	BBA
1010	Dola Barua	32000.00	CSE



Joining tables

2

Roll	Name	Gender	Age
101	Rahim	Male	18
102	Farjana	Female	17
103	Mahfuz	Male	18
104	Farjana	Female	17

Student_details

Reg_Number	Roll	GPA	Group
20171	102	3.25	Science
20172	101	3.44	Arts
20173	103	4	Science
20174	104	5	Commerce

exam_result



```
SELECT Department, SUM(Salary)
FROM teacher
GROUP BY Department;
```

```
SELECT Department, SUM(Salary)
FROM teacher
GROUP BY Department
ORDER BY SUM(Salary) DESC;
```

```
TRUNCATE TABLE teacher;
//ata dara mulotw bujay table er sob delete hoye jaby sudu heading ta cara
```

```
CREATE TABLE Students_Details
(
    Roll int NOT NULL AUTO_INCREMENT,
    Name varchar(15),
    Gender varchar(15),
    Age int,
    PRIMARY KEY(Roll)
);
```

```
INSERT INTO students_details(Roll, Name, Gender, Age)
VALUES
(10, 'Kibria', 'Male', 20),
(11, 'Ezaz', 'Male', 21),
(12, 'Sadia', 'Female', 22);
```

```
CREATE TABLE Exam_Result
(
    RegNo int NOT NULL AUTO_INCREMENT,
    Roll int,
    GPA double(3, 2),
    GroupName varchar(15),
    PRIMARY KEY(RegNo)
);

INSERT INTO exam_result(RegNo, Roll, GPA, GroupName)
VALUES
(101, 17, 3.45, 'Sci'),
(102, 19, 4.56, 'Com'),
(103, 21, 3.56, 'Sci');

SELECT students_details.Roll, RegNo, Name, GPA, GroupName
FROM students_details, exam_result
WHERE students_details.Roll = exam_result.Roll;
//dui ta table combination kory mil thakly oi info gulo dekaby.
```

or, aybabu join keyword er maddomeo kora jay

```
SELECT std.Roll, er.RegNo, std.Name, er.GPA, er.GroupName
FROM students_details AS std JOIN exam_result AS er
ON std.Roll = er.Roll;
```

```
-----  
SELECT std.Roll, er.RegNo, std.Name, er.GPA, er.GroupName  
FROM students_details AS std, exam_result AS er  
WHERE std.Roll = er.Roll;  
  
or,  
  
SELECT std.Roll, er.RegNo, std.Name, er.GPA, er.GroupName  
FROM students_details AS std JOIN exam_result AS er  
ON std.Roll = er.Roll;  
  
or,  
  
SELECT std.Roll, er.RegNo, std.Name, er.GPA, er.GroupName  
FROM students_details AS std INNER JOIN exam_result AS er  
ON std.Roll = er.Roll;  
  
//inner sudumatro dui ta table hoty jeigula match hoby sudu oi row gulo return korby.
```

```
-----  
SELECT std.Roll, er.RegNo, std.Name, er.GPA, er.GroupName  
FROM students_details AS std LEFT JOIN exam_result AS er  
ON std.Roll = er.Roll;  
  
//left join bap paser table k priority diby. many bap paser table er saty na milleo bam paser table er info  
gulo dekaby.
```

```
-----  
SELECT std.Roll, er.RegNo, std.Name, er.GPA, er.GroupName  
FROM students_details AS std RIGHT JOIN exam_result AS er  
ON std.Roll = er.Roll;
```

UNION

3

Roll	Name	Gender	Age
101	Rahim	Male	18
102	Karim	Male	17
103	Sujon	Male	18
104	Rahima	Female	18

Sylhet_tour

```
SELECT Roll,Name,Gender  
FROM Sylhet_tour  
  
UNION  
  
SELECT Roll,Name,Gender  
FROM Dhaka_tour
```

Roll	Name	Gender	Age
104	Rahima	Female	18
105	Farjana	Female	17
106	Mahfuz	Female	18
107	Shakila	Female	17

Dhaka_tour



UNION ALL

4

Roll	Name	Gender	Age
101	Rahim	Male	18
102	Karim	Male	17
103	Sujon	Male	18
104	Rahima	Female	18

Sylhet_tour

```
SELECT Roll,Name,Gender  
FROM Sylhet_tour  
  
UNION ALL  
  
SELECT Roll,Name,Gender  
FROM Dhaka_tour
```

Roll	Name	Gender	Age
104	Rahima	Female	18
105	Farjana	Female	17
106	Mahfuz	Female	18
107	Shakila	Female	17

Dhaka_tour



আগে sylhetTour এবং dhakaTour এর টেবিল তৈরি করে নিতে হবেঃ

```
1 SELECT Roll,Name,Gender  
2 FROM sylhet_tour  
3  
4 UNION  
5  
6 SELECT Roll,Name,Gender  
7 FROM dhaka_tour;
```

```
1 SELECT Roll,Name,Gender  
2 FROM sylhet_tour  
3  
4 UNION ALL  
5  
6 SELECT Roll,Name,Gender  
7 FROM dhaka_tour;
```

6.38 VIEW - PowerPoint

File Home Insert Design Transitions Animations Slide Show Review View Add-ins Tell me what you want to do

From Beginning Current Slide Present Online Custom Slide Show Set Up Slide Show Hide Slide Rehearse Record Slide Timings Show Set Up Monitors

Play Narrations Use Timings Show Media Controls

Monitor: Primary Monitor

Start Slide Show

Views allow us to:

- Structure data in a way that users or classes of users find natural or intuitive.
- Restrict access to the data in such a way that a user can see and (sometimes) modify exactly what they need and no more.
- Summarize data from various tables and use it to generate reports.

Slide 3 of 6 English (United States)

Type here to search

Notes Comments

Desktop 6:18 AM ENG 9/14/2017

The screenshot shows a Microsoft PowerPoint presentation in 'VIEW' mode. The slide content is as follows:

Views allow us to:

- Structure data in a way that users or classes of users find natural or intuitive.
- Restrict access to the data in such a way that a user can see and (sometimes) modify exactly what they need and no more.
- Summarize data from various tables and use it to generate reports.

The slide has a dark grey background with an orange header bar. The number '3' is in the top right corner. The taskbar at the bottom shows various pinned icons and the date/time '9/14/2017 6:18 AM'. The ribbon tabs shown are File, Home, Insert, Design, Transitions, Animations, Slide Show, Review, View, Add-ins, and Tell me what you want to do.

```
/*View in SQL*/
CREATE VIEW student_view AS
SELECT Roll, Name, Gender
FROM students_details;
-----
SELECT *
FROM student_view;
//orthat akta vartual table toiri kory nilam jar moddy sudu roll, name, gender gulo dekaby.
-----
//update view
UPDATE student_view
SET Name = 'Priya'
WHERE Roll = 12;
//insert row into view
INSERT INTO student_view(Roll, Name, Gender)
VALUES(13, 'Saim', 'Male');
//delete row
DELETE FROM student_view
WHERE Roll = 12;
//delete full view
DROP VIEW student_view;
```

```
/*Date and Time*/
SELECT CURRENT_DATE();
SELECT CURRENT_TIME();
SELECT now(); //date time duitai dekaby
SELECT ADDDATE('2021-12-29', INTERVAL 5 DAY); //5 din agy dekaby + month o add kora jaby.
DAY er jayhay MONTH likty hoby.
SELECT SUBDATE('2021-12-29', INTERVAL 5 DAY);
SELECT MAKEDATE(2022, 122); //orthat 2022 saler 122 tomo din er date.
SELECT DAYNAME('2021-12-29'); //ki bar oita return korby.
SELECT MONTHNAME('2021-12-29');
SELECT YEARNAME('2021-12-29');
```