

Databases: Mysql

1. উপাত্ত (Data) কি?
2. তথ্য (Information) কি?
3. উপাত্ত ও তথ্য এর পার্থক্য কি কি?
4. ডেটাবেজ (Database) কি?
5. ডেটাবেজ ব্যবহারের সুবিধা ও অসুবিধা 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 পরস্পর 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

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



ডেটাবেজ এর বিভিন্ন উপাদানসমূহ

- ডেটাবেজ এর উপাদানসমূহ

1. Field
2. Record
3. Value (data)

A Field = column

Record = row

ফিল্ড ও রেকর্ডের পার্থক্য কি কি?

Field = column
Record = row

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

Student

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

আলোচ্য বিষয়সমূহ

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

কী (Key) ফিল্ড?

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



প্রাইমারী কী (Primary Key) ?

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

কম্পোজিট কী (Composite Key) ?

- ➔ অনেক সময় একটি ফিল্ডের সাহায্যে প্রতিটি রেকর্ডকে পৃথকভাবে সনাক্ত করা যায়না।

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

Composite key

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

student

ফরেন কী (Foreign Key) ?

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

কী কে ফরেন কী বলে।

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

Primary key

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

Primary key Foreign key

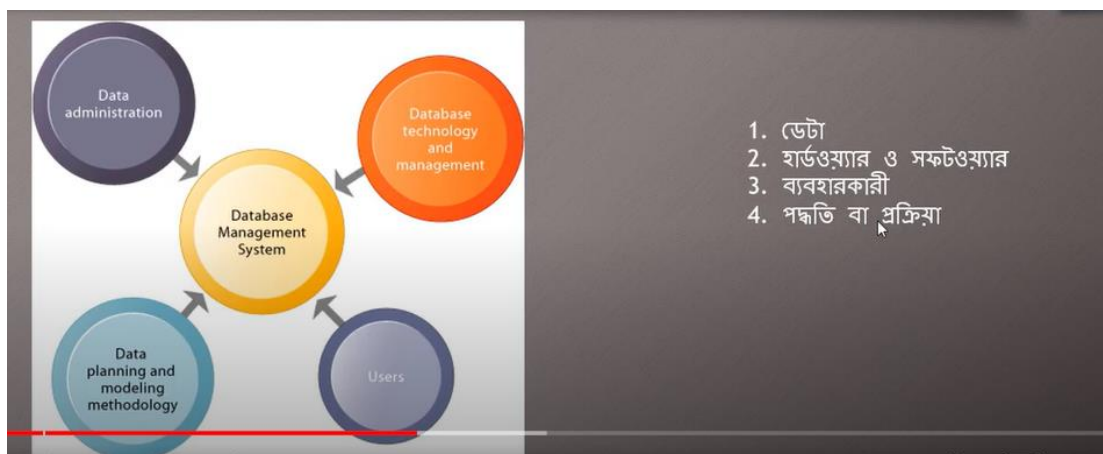
প্রাইমারী কী ও ফরেন কী এর মধ্যে পার্থক্য

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।

Database management topic:

1. ডেটাবেজ ম্যানেজমেন্ট সিস্টেম (DBMS) কী?
2. ডেটাবেজ ম্যানেজমেন্ট সিস্টেমের উপাদান কি কি?
3. ডেটাবেজ ম্যানেজমেন্ট সিস্টেম এর সুবিধা ও অসুবিধা লিখ।
4. ডেটাবেজ ম্যানেজমেন্ট সিস্টেম এর কাজ কি কি?
5. ডেটাবেজ ম্যানেজমেন্ট সিস্টেমের প্রয়োগ
6. DATA View

ডেটাবেজ ম্যানেজমেন্ট সিস্টেমের উপাদান কি কি?



ডেটাবেজ ম্যানেজমেন্ট সিস্টেম এর সুবিধা ও অসুবিধা লিখ

সুবিধা	অসুবিধা
১। সুন্দরভাবে ডেটা নিয়ন্ত্রণ করা যায় ।	১। দক্ষ জনশক্তির প্রয়োজন ।
২। ডেটা খুঁজে পাওয়া সহজ হয়।	২। বাস্তবায়ন ও পরিচালনা করা বেশ ব্যয়বহুল ।
৩। ডেটাবেজের নিরাপত্তা প্রদান করে ।	
৪। বহু ব্যবহারকারী একসাথে ব্যবহার করতে পারে ।	

ডেটাবেজ ম্যানেজমেন্ট সিস্টেম এর কাজ কি কি?

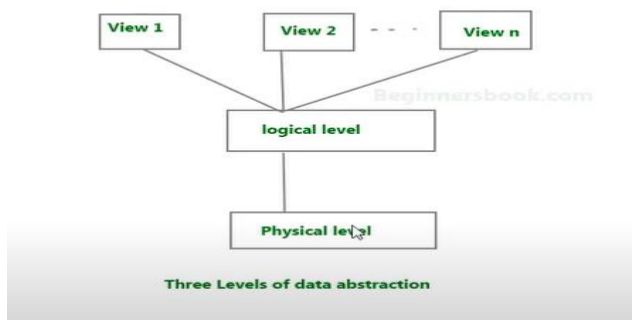
1. CREATE DATABASE
2. UPDATE DATABASE
3. PRINT DATABASE
4. BACKUP AND RECOVERY DATABASE
5. SAVE DATA
6. RETRIEVE DATA
7. DELETE DATA
8. SECURE DATA etc.

How use in database example:

ডেটাবেজ ম্যানেজমেন্ট সিস্টেমের প্রয়োগ

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

Database model:



Importance discussion:

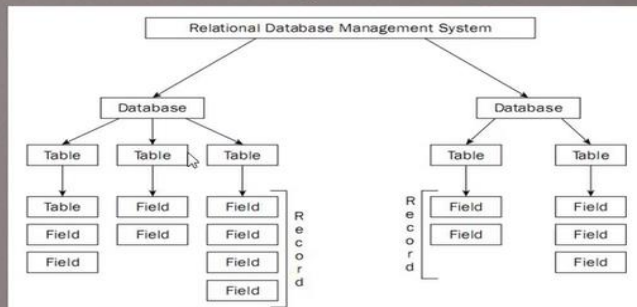
আলোচ্য বিষয়সমূহ

1. ডেটাবেজ রিলেশন কি?
2. ডেটাবেজ রিলেশন এর প্রকারভেদ ।
3. সার্টিং VS ইন্ডেক্সিং কি ? সুবিধা ও অসুবিধা
4. কর্পোরেট ডেটাবেজ কি?
5. ডেটা সিকিউরিটি ? লাইফ সাইকেল
6. ডেটা এনক্রিপশন

রিলেশনাল ডেটাবেজ ম্যানেজমেন্ট সিস্টেম (RDBMS) কী?

RDBMS = Relational DataBase Management System

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



Example:

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

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 ।

Use :

ডেটাবেজ ম্যানেজমেন্ট সিস্টেমের প্রয়োগ

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.

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।

1. One-To-One
2. One-To-Many
3. Many-To-Many

ডেটাবেজ রিলেশন তৈরির শর্ত:

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

Roll	Name	Group	Reg No.	Roll	GPA	Section
101	Ayesha	Science	100101	101	4.5	A
102	Tamim	Science	100102	102	4.5	A
103	Ayesha	Arts	100103	103	5	B
104	Lipu	Science	100104	104	4.7	A
105	Rabeya	Arts	100105	105	5	B

Student Table Result Table

One-To-One Database Relation:

১। কোন একটি টেবিলের একটি রেকর্ডের সাথে যখন অন্য টেবিলের কেবল একটি রেকর্ডের সম্পর্ক সুপন 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।

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

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

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

১। কোন একটি টেবিলের একটি রেকর্ডের সাথে যখন 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 ধরনের সম্পর্ক তৈরি হয়।

Roll	Name
101	Ayesha
102	Tamim
103	Ayesha

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

Many-To-Many Database Relation:

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

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

৩। দুটি টেবিলের প্রাইমারী কি জংশন টেবিলের ফরেন কি হিসাবে ব্যবহার করা হয় এই ধরনের সম্পর্কে।

The diagram illustrates a Many-to-Many relationship between two tables: Teacher Table and Course Table. The Teacher Table has columns Teacher_ID and Teacher_Name. The Course Table has columns Course_ID and Subject_Name. The Routine Table acts as a junction table with columns Teacher_ID, Course_ID, and Day. Arrows indicate the relationships between the tables.

Teacher_ID	Teacher_Name
T01	Rumel
T02	Sohid
T03	Alak

Course_ID	Subject_Name
C01	English
C02	Bangla
C03	Math

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

New topic : Sorting

- Sorting অর্থ সাজানো।
- Sorting ২ ধরনের।
- Ascending (উর্ধ্বক্রম)
- Descending (নিম্নক্রম)

Shain, anis, bijoy, zakir

Anis, bijoy, shain, zakir

মেমো, হাইপারলিঙ্ক, OLE ফিল্ডে সর্টিং হয়না।

The diagram shows three tables representing the same data sorted in different ways. The first table is the original data. The second table is sorted in ascending order (উর্ধ্বক্রম) by GPA. The third table is sorted in descending order (অধঃক্রম) by GPA.

Roll	Name	GPA
101	Shewly	4.5
102	Karim	2.5
103	Lipu	3.7
104	Ayesha	4.90

Roll	Name	GPA
102	Karim	2.5
103	Lipu	3.7
101	Shewly	4.5
104	Ayesha	4.90

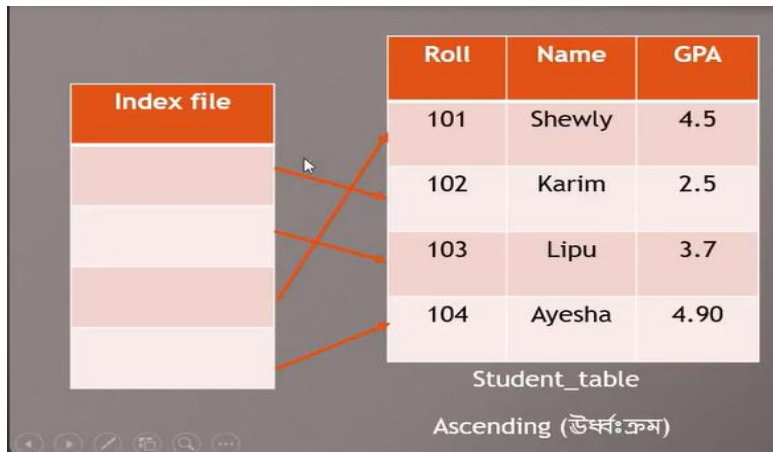
Roll	Name	GPA
104	Ayesha	4.90
101	Shewly	4.5
103	Lipu	3.7
102	Karim	2.5

১। সর্টিং হল ডেটা টেবিলের রেকর্ডগুলোকে কোন নির্ধারিত ফিল্ড অনুসারে সাজানো।

- ২। সার্টিং এর ফলে ডেটা খুঁজে (searching) বের করা সহজ হয়।
- ৩। ডেটাবেজে আলাদা সর্টেড ফাইল তৈরি হয়।
- ৪। সার্টিং এর ফলে মূল ডেটা ফাইলে রেকর্ডের ক্রমিক নং পরিবর্তন হয়।
- ৫। ডেটাবেজে কোন রেকর্ড সংশোধন / সংযোজন করলে সট করা ফাইল আপডেট হয়না।
- ৬। রেকর্ডগুলো সাজানোর জন্য তুলনামূলক বেশী সময়ের প্রয়োজন হয়।

Indexing:

- ১। ইন্ডেক্সিং হল মূল টেবিল অপরিবর্তিত রেখে ডেটা টেবিলের রেকর্ডগুলোকে কোন নির্ধারিত ফিল্ড অনুসারে সাজানো।
- ২। ইনডেক্স ফাইলে কোন রেকর্ড থাকেনা, তবে লজিকাল অর্ডার থাকে।
- ৪। ইন্ডেক্সিং এর ফলে মূল ডেটা ফাইলে রেকর্ডের ক্রমিক নং পরিবর্তন হয় না।
- ৫। ডেটাবেজে কোন রেকর্ড সংশোধন / সংযোজন করলে index করা ফাইল স্বয়ংক্রিয়ভাবে আপডেট হয়।



Sorting vs Indexing

Sorting	Indexing
১। সর্টিং হল ডেটা টেবিলের রেকর্ডগুলোকে কোন নির্ধারিত ফিল্ড অনুসারে সাজানো।	১। ইন্ডেক্সিং হল মূল টেবিল অপরিবর্তিত রেখে ডেটা টেবিলের রেকর্ডগুলোকে কোন নির্ধারিত ফিল্ড অনুসারে সাজানো।
২। সর্টিং 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 সর্টি 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। সর্টিং এ মেমোরি বেশী প্রয়োজন হয়।	৫। ইন্ডেক্সিংএ মেমোরি কম প্রয়োজন হয়।

Security

আলোচ্য বিষয়সমূহ

১. ডেটা সিকিউরিটি
২. ডেটা এনক্রিপশন
৩. প্লেইনটেক্সট
৪. সাইফারটেক্সট

- ডেটা সিকিউরিটি কি?
- অনির্দিষ্ট ব্যক্তির 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-

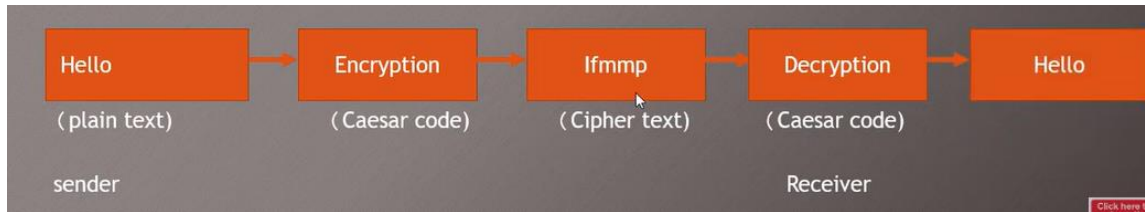
- ১) সিজার কোড
- ২) DES (Data Encryption Standard)
- ৩) IDEA (International Data Encryption Algorithm)

Caesar Cipher

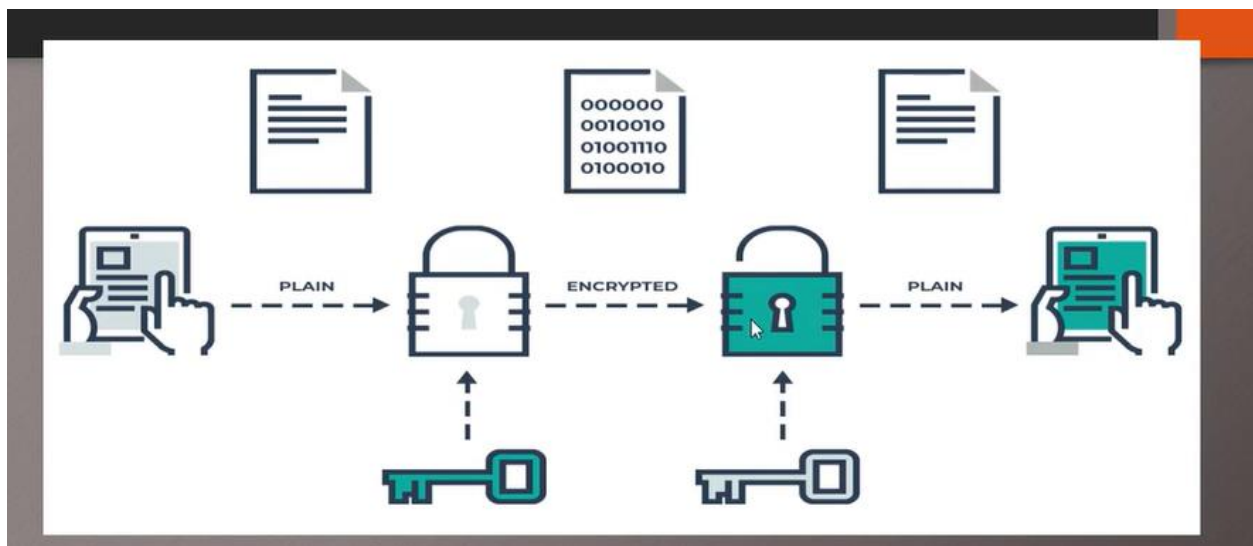
The **Caesar Cipher**, also known as a shift cipher, is one of the oldest and simplest forms of encrypting a message. It is a type of substitution cipher where each letter in the original message (which in cryptography is called the plaintext) is replaced with a letter corresponding to a certain number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

এনক্রিপশন এর উদাহরণ

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



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



এনক্রিপশন এর প্রকারভেদ

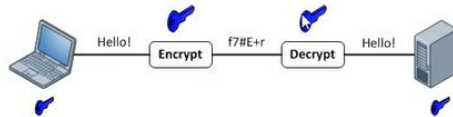
- সাধারণত ২ ধরনের এনক্রিপশন দেখা যায়।

- গোপন কি এনক্রিপশন / সিমিট্রিক এনক্রিপশন
- পাবলিক কি এনক্রিপশন / অ্যাসিমিট্রিক এনক্রিপশন

solution will typically implement all three.

Symmetric Encryption

Symmetric encryption may also be referred to as *shared key* or *shared secret* encryption. In symmetric encryption, a single key is used both to encrypt and decrypt traffic.



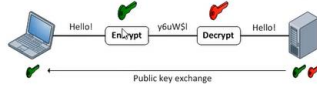
Common symmetric encryption algorithms include DES, 3DES, AES, and RC4. 3DES and AES are commonly used in IPsec and other types of VPNs. RC4 has seen wide deployment on wireless networks as the base encryption used by WEP and WPA version 1.

Symmetric encryption algorithms can be extremely fast, and their relatively low complexity allows for easy implementation in hardware. However, they require that all hosts participating in the encryption have already been

Asymmetric Encryption

Asymmetric encryption is also known as *public-key cryptography*. Asymmetric encryption differs from symmetric encryption primarily in that two keys are used: one for encryption and one for decryption. The most common asymmetric encryption algorithm is RSA.

Compared to symmetric encryption, asymmetric encryption imposes a high computational burden, and tends to be much slower. Thus, it isn't typically employed to protect payload data. Instead, its major strength is its ability to establish a secure channel over a nonsecure medium (for example, the Internet). This is accomplished by the exchange of public keys, which can only be used to encrypt data. The complementary private key, which is never shared, is used to decrypt.



Robust encryption solutions such as IPsec implement the strengths of both symmetric and asymmetric encryption. First, two endpoints exchange public keys, which allows for the setup of a slow but secure channel. Then the two hosts decide on and exchange shared symmetric encryption keys to construct much faster symmetric encryption channels for data.

Query Languages:

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 প্রয়োজনীয় তথ্য খুঁজে বের করাকে কুয়েরি বলে।
 - বিভিন্ন ধরনের 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 ডেটাগুলো 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।
- ➔ 1. Make Table Query
- ➔ 2. Append Query (সংযোজন করা)
- ➔ 3. Delete Query
- ➔ 4. Update Query

কুয়েরি ভাষা:

- যে ভাষার সাহায্যে কুয়েরি করা হয়, তাকে কুয়েরি ভাষা বলে।
- ডেটাবেজে বহুল ব্যবহৃত ৩ টি কুয়েরি ভাষা হল

I. QUEL (Query Language)

II. QBE (Query By Example)

III. SQL (Structured Query Language)

SQL:

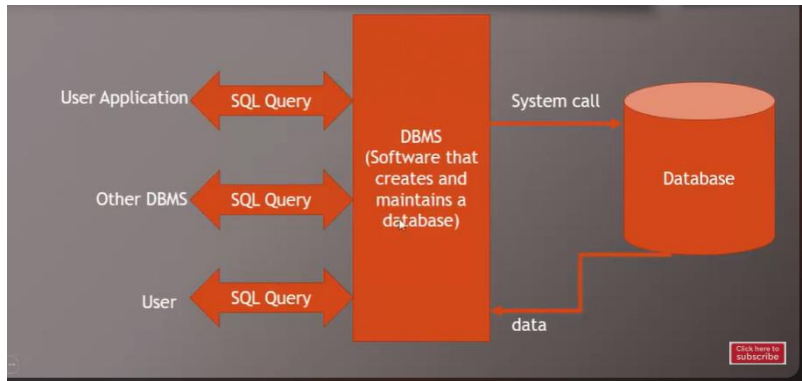
- SQL স্টেটমেন্ট এর বৈশিষ্ট্য
 - I. কেস সেনসিটিভ নয়। SELECT / select
 - II. স্টেটমেন্ট এর শেষে সেমিকোলন (;) দিতে হয়।
 - III. স্টেটমেন্টকে ভেঙ্গে একাধিক লাইনে লেখা যায়।



DBMS full form:

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

- SQL = Structured Query Language

কুয়েরি কি?

1. ডেটাবেজ থেকে রেকর্ড বা রেকর্ডসমূহকে খুঁজে বের করারকে কুয়েরি বলে।
2. ডেটাবেজে 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

XAMPP ডাটাবেইজস কমান্ডঃ

- SHOW DATABASES;

Table create database:

- CREATE DATABASE collage;

Database delete:

DROP DATABASE collage;

How to datatype use:

- Numeric data type

- SMALLINT
- INT
- BIGINT

- FLOAT
- DOUBLE(M,D)
- DECIMAL(M,D)

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

- 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

1. BLOB
2. Logical

Creating a table:

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)

);

Example:

```
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)
```

```
);
```

Renaming Tables:

RENAME command এর মাধ্যমে টেবিলের নাম পরিবর্তন করতে পারবেন।

Syntax

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

Example : RENAME TABLE student TO students;

Data insertion:

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);

- VALUES (value1, value2, value3,...valueN); INSERT INTO student_details (Roll, Name, Gender, Age, GPA, City) VALUES (101, Rahim, 'Male', 18,3.44, 'Sylhet');

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 VALUES (value1, value2, value3,...);

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

INSERT INTO student_details

(Roll, Name, Gender, Age, GPA, City)

VALUES (101, 'Rahim', 'Male', 18, 3.44, 'Sylhet'),

(102, 'Hasina', 'Female', 17, 4.50, 'Dhaka');

Home work (create table):

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

DDL(Data definition language)

DDL -> CREATE, ALTER, DROP

DML(Data manipulation language)

DML> INSERT, SELECT, UPDATE, DELETE

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

Syntax

- SELECT column_list
- FROM table_name;

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

Roll	Name	City
101	Rahim	Sylhet
102	Hasina	Dhaka
103	Sabul	Sylhet
104	Suhan	Khulna
105	Rahim	Barisal
106	Tanvir	Sylhet
107	Hazira	Sylhet
108	Hafiza	Dhaka

SELECT Roll, Name, City FROM student_deatils;

টেবিল সবগুলো ডাটা দেখতে চাই ঃ

SELECT *FROM student_deatils;

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
Sylhet
Dhaka
Khulna
Barisal

SELECT DISTINCT City FROM student_deatils;

DISTINCT key তা duplicate সিটি নামে শো করবে না।

SELECT *FROM student_deatils

LIMIT 5; সুদু ৫ টা ডাটা শো করবে ।

Sorting records with ORDER BY clause:

SELECT Name

FROM student_details

ORDER BY Name; → Ascending সিরিয়াল অনুজায়ি নাম সাজায়।

SELECT Roll, Name, GPA

FROM student details

ORDER BY Name DESC; → Descending (Ascending আর উল্টা)

Arithmetic Operator: +,-,*,/,% and =, !=, <=,<,>,>=, between.

Command : SELECT 5+2;

Logical Operator:

- AND
- OR
- IN
- NOT

WHERE clause:

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

Example:

SELECT column_list

FROM table_name

WHERE condition;

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

SELECT City

FROM student_deatils

WHERE Gender='Femate';

Comparison operator:

→ =

→ !=

→ <=

→ <

→ >

→ > =

→ between

SELECT Roll, Name, GPA

FROM student_details

WHERE Roll BETWEEN 102 AND 105; যে সার্চ করতে হবে অই টুকু সুদু আসবে

Logical Operator:

- AND
- OR
- IN
- NOT

SELECT *

FROM student_details

WHERE Gender Male' OR GPA-3.50;

SELECT *

FROM student_details

WHERE City = 'Sylhet'

AND

(Gender Male' AND GPA-3.50);

SELECT *

FROM student_details

WHERE City='Sylhet'

OR

City = 'Dhaka'

OR

City = 'Barisal';

সবগুলো একসাথে দেখাতে IN operation ইউজ করা হয়।

SELECT *

SELECT *

FROM student details

WHERE Name LIKE 's%'; সুদু ক্যারেঙ্কার খুঁজতে ইউজ করা হয়।

AS Keyword for custom name:

SELECT Roll, Name

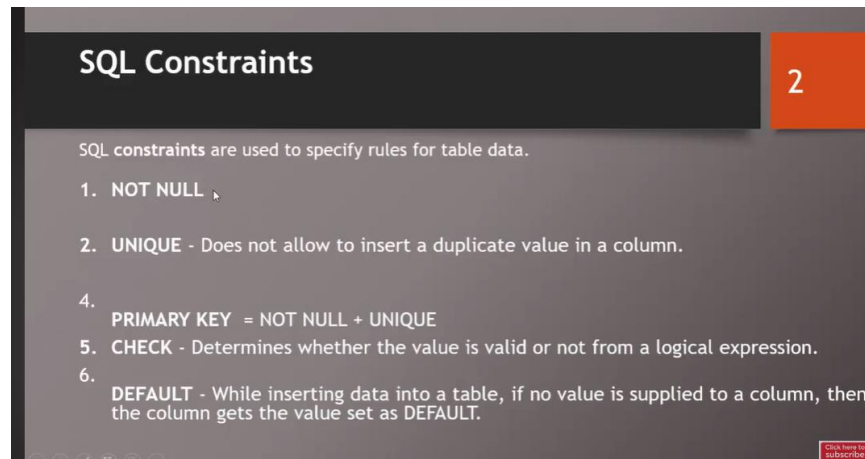
FROM student_details;

আমারা AS keyword use করবো

SELECT Roll AS ID , Name AS 'First Name'

FROM student_details;

Constraint, AUTO INCREMENT SQL commands:



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.

Subscribe

CREATE TABLE teacher

(

ID int,

Name varchar(15)

);

INSERT INTO teacher (ID,Name)

VALUES (1000,'EMON');

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 (1000,'EMON',35000); → সুদূর যোগে ইউজ করবো সেটা ইউজ করবো যেমন(NAME, SALARY).

UPDATE Statement:



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

[Click here to subscribe](#)

UPDATE table_name

SET column1=value1, column2=value2,

WHERE condition;

UPDATE teacher

SET Salary=32000

WHERE ID=1002;

DELETE Statement in SQL:

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



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

Click here to subscribe

এই টেবিল row ডিলিট করবো

```
DELETE FROM table_name
```

```
WHERE condition;
```

```
DELETE FROM teacher
```

```
WHERE ID=1005;
```

```
DELETE FROM teacher
```

```
WHERE ID>1003; → এই তা ইউজ করলে ১০০৩ পর সবগুলো ডাটা ডিলিট হয়ে যাবে।
```

```
UPDATE teacher
```

```
SET Salary = Salary + 5000
```

```
WHERE Salary > 1000;
```

UPPER and LOWER Function:

```
SELECT UPPER('I am emon Khan');
```

```
SELECT LOWER('I AM EMON khan');
```

Functions:

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

```
SELECT CONCAT('Welcome to','mysql database');
```

```
SELECT CONCAT(Name, 'is',age,'years old');
```

```
SELECT CONCAT(Name, 'is',age,'years old') AS student  
FROM student_details;
```

Aggregate Functions in SQL::

- Group functions operate on sets of rows to give one result per group.
- AVG
COUNT
MAX
MIN
SUM

```
DESCRIBE teacher;
```

- SELECT COUNT(*)
FROM student_details;
- SELECT MAX(GPA)
FROM student_details;

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

```
SELECT Roll, Name, Age, City, MIN(GPA)
FROM student_details
WHERE Gender- 'Female';
```

Sub Query in SQL::

```
INSERT INTO teacher (Name, Salary)
VALUES 3 ('Selina', 32000),
('Asad', 38000),
('Sohid', 33500),
('Alin', 33000),
('Dola Barua', 32000);
```

```
SELECT *
FROM teacher
WHERE Salary (SELECT AVG(Salary) FROM teacher);
```

ALTER statement in SQL::

ALTER TABLE

- ALTER TABLE command এর সাহায্যে
 1. টেবিলে নতুন কলাম যুক্ত করা যায় ।
 2. কোন কলামের নাম পরিবর্তন করা যায়।
 3. কোন কলাম delete করা যায় ।

Adding a column (টেবিলে নতুন কলাম যুক্ত করা)

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

#

```
ALTER TABLE teacher
ADD Age int(5);
```

```
ALTER TABLE teacher
ADD Dept varchar(15);
```

Renaming column (কোন কলামের নাম পরিবর্তন করা)

Syntax

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

ALTER TABLE teacher

CHANGE Dept Department varchar(15);

Dropping a column (কলাম delete করা)

• Syntax

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

ALTER TABLE teacher DROP COLUMN Age;

UPDATE Statement (Review):

UPDATE teacher

SET Department='CSE' WHERE ID =1000;

GROUP BY clause::

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

SELECT Department, SUM(Salary)

FROM teacher

GROUP BY Department

ORDER BY SUM(Salary) DESC;

TRUNCATE TABLE Command in SQL::

TRUNCATE TABLE student details; → টেবিলএ ডাটা মুছে ফেলা হয়।

Joining Tables:

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

দুইটা টেবিলকে জয়েন্ট করতে হবে।

Table-1:

```
CREATE TABLE student_details
```

```
(
```

```
Roll int NOT NULL AUTO_INCREMENT,
```

```
Name varchar(15) NOT NULL,
```

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

```
Age int,
```

```
PRIMARY KEY(Roll)
```

```
);
```

INSERT:

```
INSERT INTO student_details
```

```
(Roll, Name, Gender, Age)
```

```
VALUES
```

```
(301, 'Rahim', 'Male", 18);
```

```
INSERT INTO student_details
```

```
(Name, Gender, Age)
```

```
VALUES
```

```
('Rahim', 'Male", 18);
```


('Rahim', 'Male", 18);

('Rahim', 'Male", 18);

('Rahim', 'Male", 18);

Exam result table create :

INSERT INTO exam result (Reg Number, Roll, GPA, Group_Name)

VALUES

(20171,102,3.25, 'Science'),

(20172,101,3.44, 'Arts'),

(20173,103,4, 'Science'),

(28174,184,5, 'Commerce');

জয়েন্ট এর টেবিল করতে কোডঃ

SELECT student_deatils.Roll, Reg Number, Name, Gender, Group Name, GPA

FROM student_details,exam_result

WHERE student details.Roll = exam_result.Roll ;

দুই জয়েন্ট টেবিল আর সম্পর্ক কোডঃ

SELECT student details. Roll, exam_result.Reg Number, student details. Name, student details.Gender,

exam_result.Group_name,exam_result.GPA

FROM student_details, exam_result

WHERE student details. Roll exam_result.Roll;

New table:

SELECT std. Roll, exam. Reg Number,

std.Name, std.Gender,

exam. Group name,exam.GPA

FROM student_details AS std JOIN exam result AS exam

ON std.Roll = exam. Roll;