

Relational Database Concepts

© IBM Corporation. All rights reserved.

Relational Model

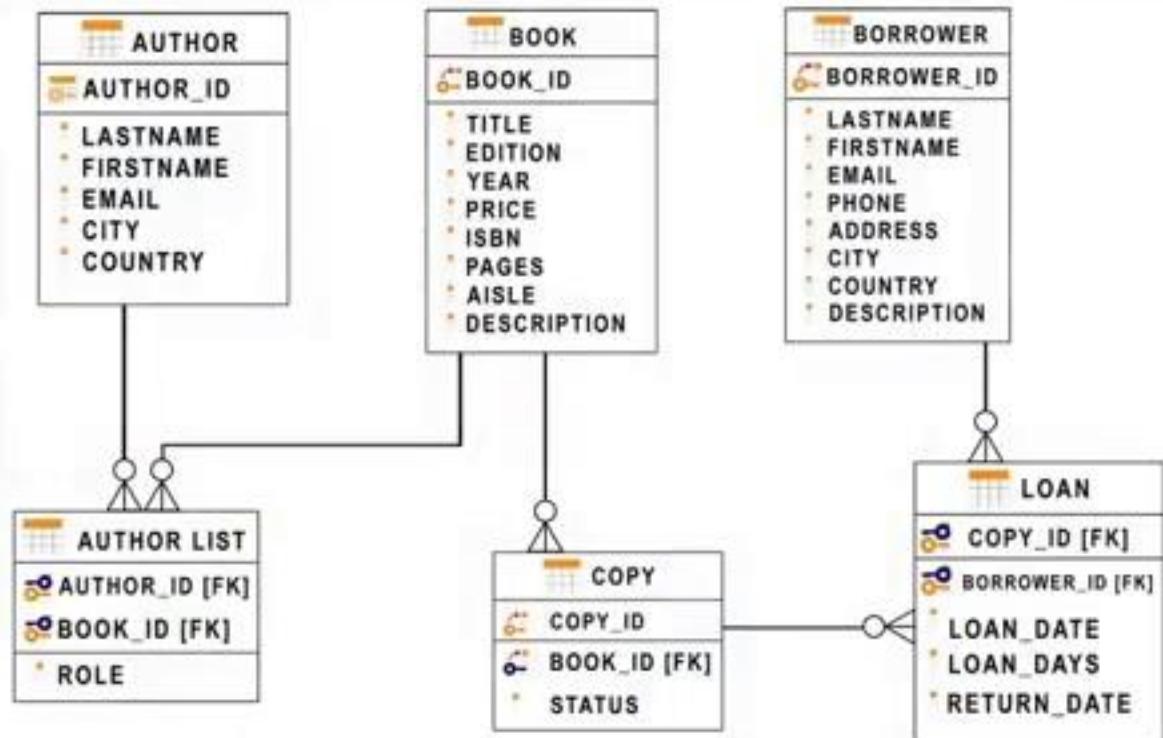
- Most used data model
- Allows for data independence

Relational Model

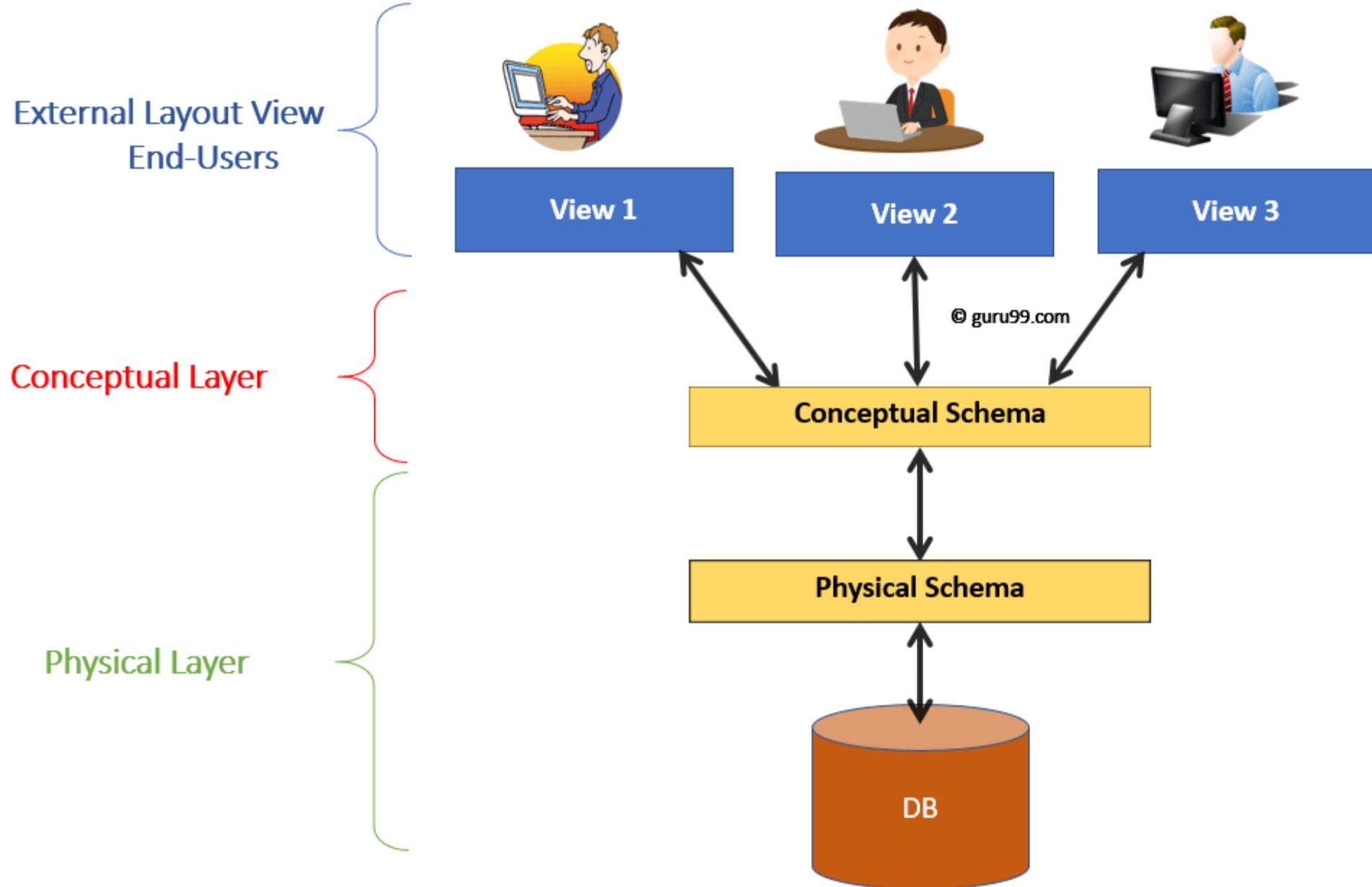
- Most used data model
- Allows for data independence
- Data is stored in tables

Relational Model

- Most used data model
- Allows for data independence
- Data is stored in tables



logical data independence - physical data independence - physical storage independence



Entity-Relationship Model

- Used as a tool to design relational databases



Entity-Relationship Model

- Used as a tool to design relational databases



BOOK
BOOK_ID
TITLE
EDITION
YEAR
PRICE
ISBN
PAGES
AISLE
DESCRIPTION

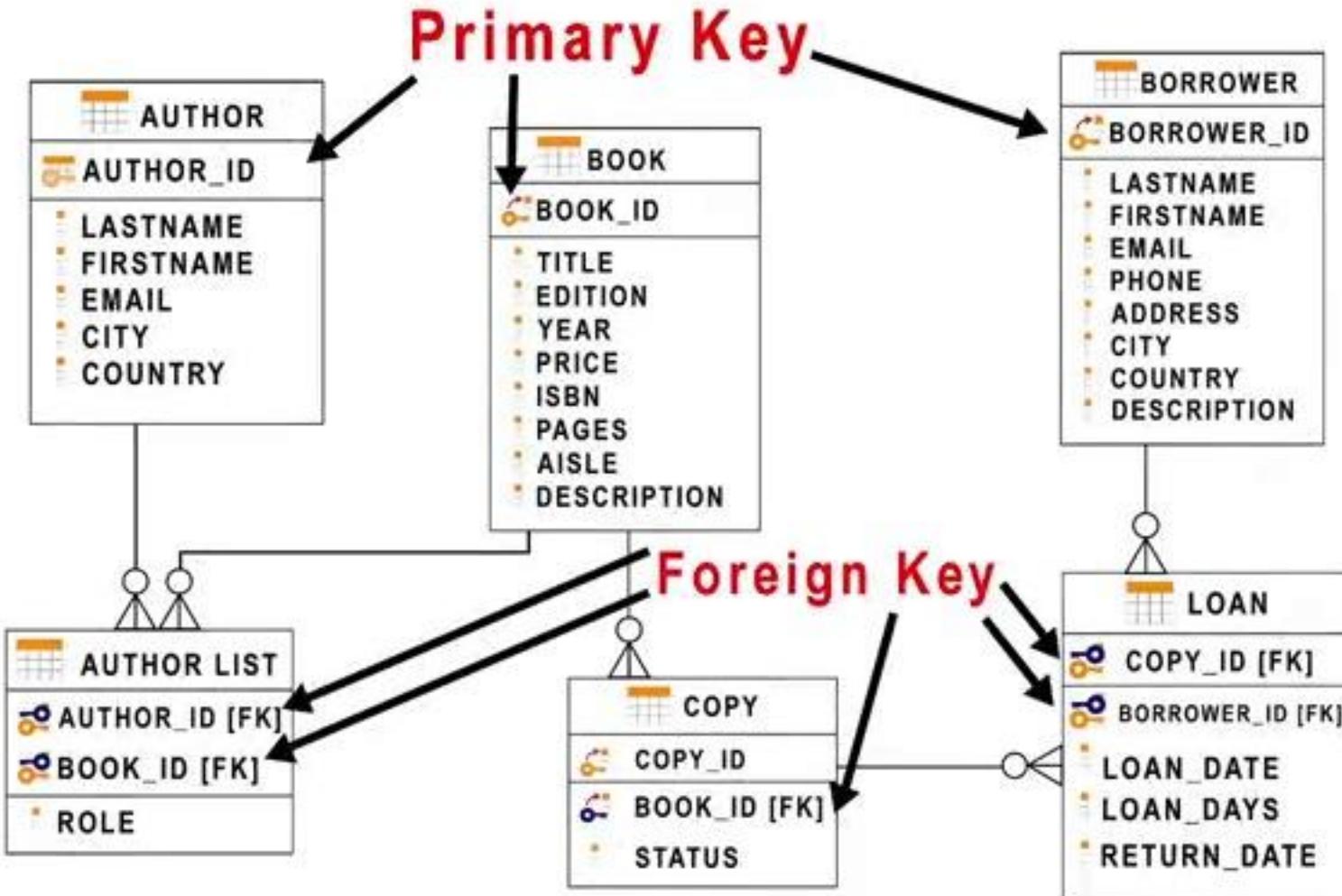
Mapping Entity Diagrams to Tables

- Entities become tables
- Attributes get translated into columns

Table: Book

Title	Edition	Year	Price	ISBN	Pages	Aisle	Description
Database Fundamentals	1	2010	24.99	978-0-9866283-1-1	300	DB-A02	Teaches you the fundamentals of databases
Getting started with DB2 Express-C	1	2010	24.99	978-0-9866283-5-1	280	DB-A01	Teaches you the essentials of DB2 using DB2 Express-C, the free version of DB2

Primary Keys and Foreign Keys



Cloud databases

- ✓ Ease of Use and Access



Cloud databases

- ✓ Ease of Use and Access
 - API
 - Web Interface



Cloud databases

- ✓ Ease of Use and Access
 - API
 - Web Interface
 - Cloud or Remote Applications

- ✓ Scalability & Economics



Cloud databases

- ✓ Ease of Use and Access
 - API
 - Web Interface
 - Cloud or Remote Applications

- ✓ Scalability & Economics
 - Expand/Shrink Storage & Compute Resources



Cloud databases

- ✓ Ease of Use and Access
 - API
 - Web Interface
 - Cloud or Remote Applications

- ✓ Scalability & Economics
 - Expand/Shrink Storage & Compute Resources
 - Pay per use

- ✓ Disaster Recovery



Cloud databases

- ✓ Ease of Use and Access
 - API
 - Web Interface
 - Cloud or Remote Applications

- ✓ Scalability & Economics
 - Expand/Shrink Storage & Compute Resources
 - Pay per use

- ✓ Disaster Recovery
 - Cloud Backups and Geographical Distribution



Examples of Cloud databases

- IBM Db2
- Databases for PostgreSQL



Examples of Cloud databases

- IBM Db2
- Databases for PostgreSQL
- Oracle Database Cloud Service
- Microsoft Azure SQL Database
- Amazon Relational Database Services (RDS)



Examples of Cloud databases

- IBM Db2
- Databases for PostgreSQL
- Oracle Database Cloud Service
- Microsoft Azure SQL Database
- Amazon Relational Database Services (RDS)

Available as:

- VMs or Managed Service



Examples of Cloud databases

- IBM Db2
- Databases for PostgreSQL
- Oracle Database Cloud Service
- Microsoft Azure SQL Database
- Amazon Relational Database Services (RDS)

Available as:

- VMs or Managed Service
- Single or Multi-tenant



Single Tenant



Multi-Tenant



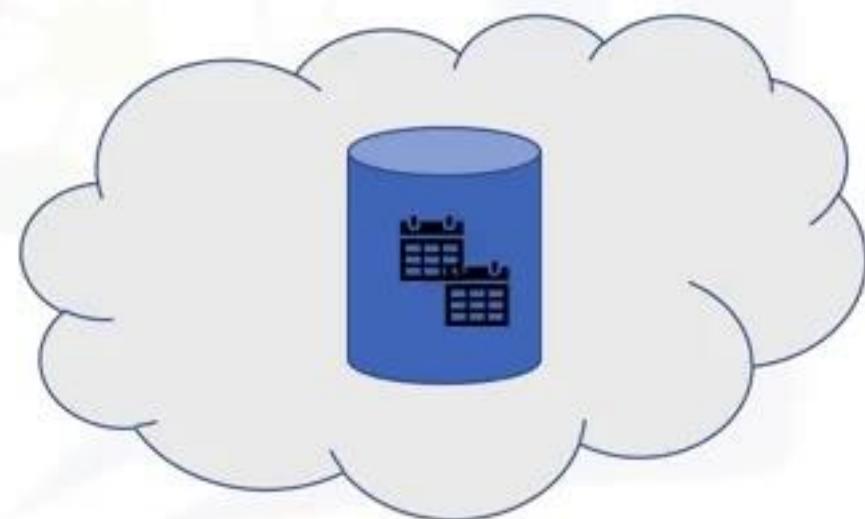
VS.

Database service instances

- DBaaS provides users with access to Database resources in cloud without setting up hardware and installing software.

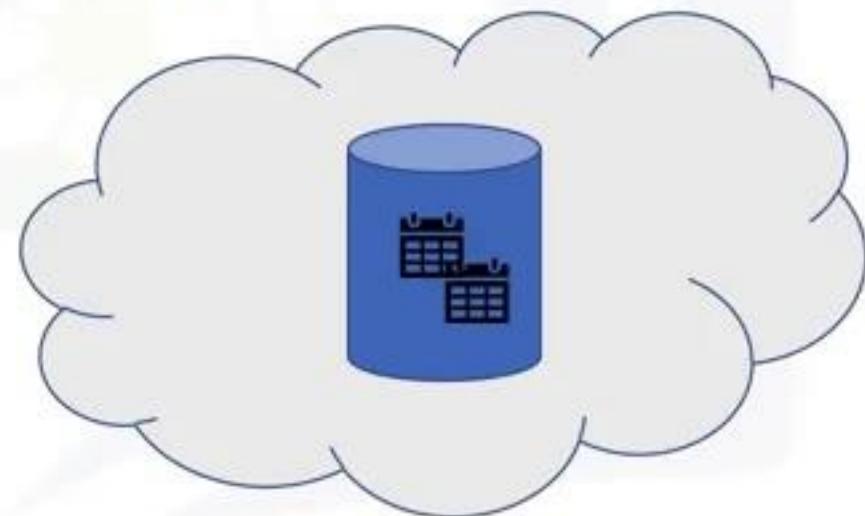
Database service instances

- DBaaS provides users with access to Database resources in cloud without setting up hardware and installing software.
- Database service instance holds data in data objects / tables



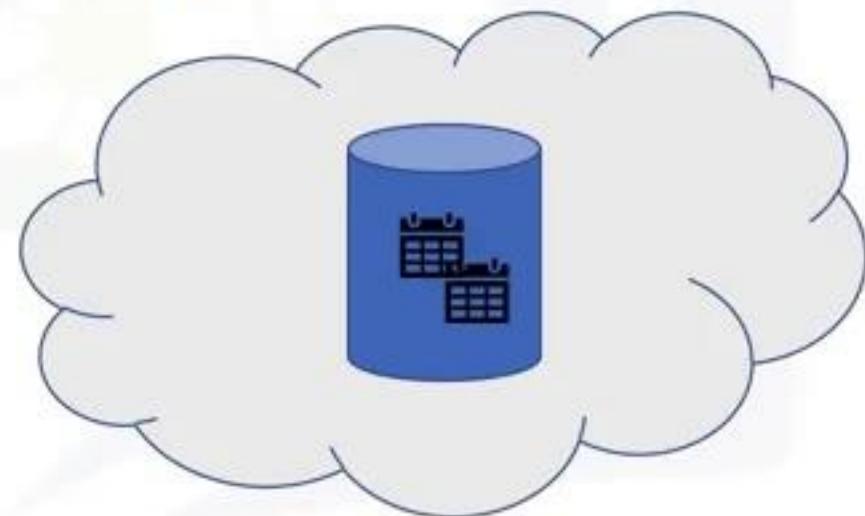
Database service instances

- DBaaS provides users with access to Database resources in cloud without setting up hardware and installing software.
- Database service instance holds data in data objects / tables
- Once data is loaded, it can be queried using web interfaces and applications



Database service instances

- DBaaS provides users with access to Database resources in cloud without setting up hardware and installing software.
- Database service instance holds data in data objects / tables
- Once data is loaded, it can be queried using web interfaces and applications



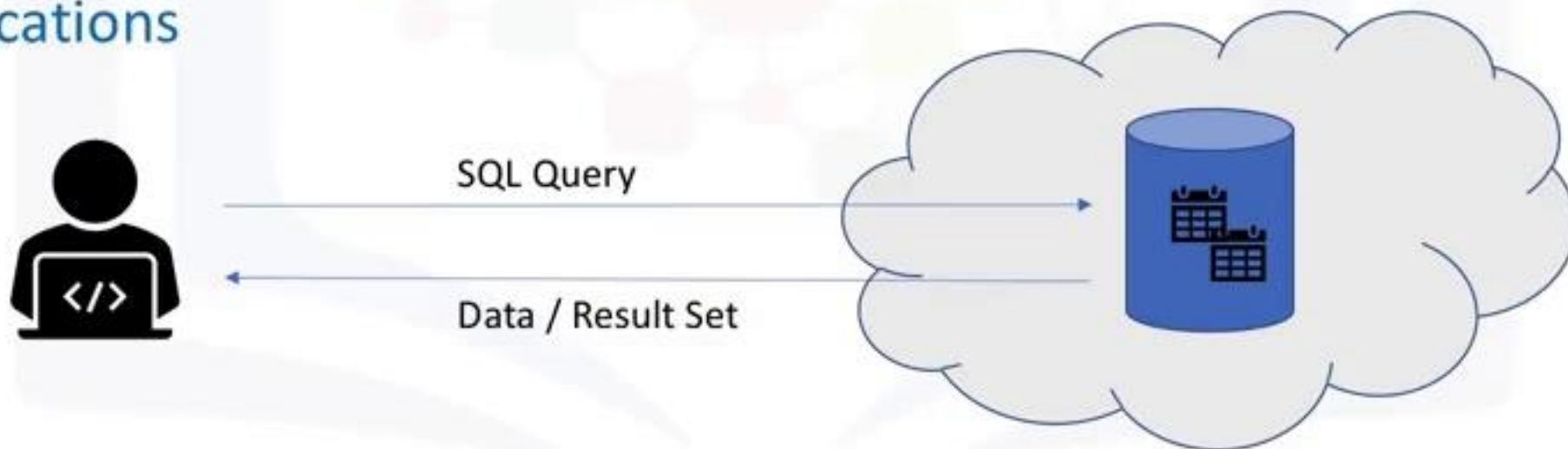
Database service instances

- DBaaS provides users with access to Database resources in cloud without setting up hardware and installing software.
- Database service instance holds data in data objects / tables
- Once data is loaded, it can be queried using web interfaces and applications



Database service instances

- DBaaS provides users with access to Database resources in cloud without setting up hardware and installing software.
- Database service instance holds data in data objects / tables
- Once data is loaded, it can be queried using web interfaces and applications



Creating a database instance on IBM Db2 on Cloud



IBM Db2 on Cloud



Deploy an instance of Db2 on Cloud Service

The screenshot shows the IBM Cloud Catalog interface. At the top, there's a navigation bar with links for Catalog, Docs, a profile icon, Log in, and Sign up. Below the navigation is a search bar with the placeholder "Search the catalog...". On the left, a sidebar titled "Category" lists several service types with checkboxes: Compute, Containers, Networking, Storage, AI, and Analytics. The main area displays three service cards. The first card is for "Compose for RethinkDB", which is described as a JSON document-based, distributed database. The second card is for "Compose for ScyllaDB", described as a highly performant, in-place replacement for Cassandra. The third card, for "Db2", is highlighted with a green oval border. It is described as a next-generation SQL database, formerly known as dashDB For Transactions. The Db2 card also indicates it is "Lite • Free • IAM-enabled".

Deploy an instance of Db2 on Cloud Service

The screenshot shows the IBM Cloud Catalog interface. The left sidebar has a 'Category' filter with checkboxes for Compute, Containers, Networking, Storage, AI, Analytics, Databases (which is checked), Developer Tools, Integration, Internet of Things, Security and Identity, and Web and Mobile. The main area shows several service cards:

- Compose for RethinkDB**: IBM + Analytics • Databases. Description: RethinkDB is a JSON document based, distributed database with an integrated administration and exploration console.
- Compose for ScyllaDB**: IBM + Analytics • Databases. Description: ScyllaDB is a highly performant, in-place replacement for the Cassandra wide-column distributed database. Status: Beta.
- Db2**: IBM + Analytics • Databases. Description: A next generation SQL database. Formerly dashDB For Transactions. Status: Lite • Free • IAM-enabled. This card is highlighted with a green oval.
- Db2 Hosted**: IBM + Databases. Description: Offers customers the rich features of an on-premises Db2 deployment without the cost, complexity, and management overhead of maintaining their own hardware and software. Status: Free. This card is highlighted with a red box.
- Db2 Warehouse**: IBM + Analytics + Data. Description: Db2 Warehouse on Cloud provides a flexible and powerful data warehouse for enterprise-level analytics. Status: Dedicated.
- GEO Web Services**: Third party + Databases. Description: Adding geo-intelligence to your business. Status: Free.

Deploy an instance of Db2 on Cloud Service

The screenshot shows the IBM Cloud Catalog interface. On the left, a sidebar lists categories: Compute, Containers, Networking, Storage, AI, Analytics, **Databases** (which is checked), Developer Tools, Integration, Internet of Things, Security and Identity, and Web and Mobile. The main area displays a search bar and a grid of service cards. The 'Db2' card is highlighted with a green border and a red circle around its 'Lite + Free + IAM-enabled' badge. It also has a large red 'X' over it. Other cards include 'Compose for RethinkDB', 'Compose for ScyllaDB', 'Db2 Hosted' (with a large red 'X'), 'Db2 Warehouse' (with a large red 'X'), and 'GEO Web Services'. The 'Db2' card contains the following text:

Db2
IBM + Analytics + Databases
A next generation SQL database. Formerly dashDB For Transactions.
Lite + Free + IAM-enabled

Create a new service

The screenshot shows the IBM Cloud interface for creating a new Db2 service. The top navigation bar includes 'IBM Cloud', 'Search', 'Catalog', 'Docs', 'Support', 'Manage', and user icons. The main content area is titled 'Db2' and shows the following details:

- Author:** IBM • Date of last update: 04/29/2020 • Docs • API docs
- Create** (button) • **About** (button)
- Select a region**: A dropdown menu currently set to 'Dallas'.
- Select a pricing plan**: A table showing the 'Lite' plan details:

Plan	Features	Pricing
Lite	200 MB of data storage 5 simultaneous connections Shared multitenant system	Free

A note below states: 'The Free plan provides a free Db2 service for development and evaluation. The plan has a set amount of limitations as shown. You can continue using the free plan for as long as needed, however, users are asked to re-extend their free account every 90 days by email. If you do not re-extend, your free account is cleaned out a further 90 days later. This helps provide free resources for everyone.'

Create (button) • **Add to estimate** (button)

View the newly created service

Dashboard

Resource summary

View all

4 Resources

Services (3)

Storage

Service	Type
Db2-tq-01	Default
Watson Studio-8w	Default
watson-vision-combined-ey	Default

The screenshot shows the IBM Cloud dashboard. The 'Services' tab is selected and highlighted with a red box. Under the 'Services' section, there are three entries: 'Db2-tq-01', 'Watson Studio-8w', and 'watson-vision-combined-ey'. The first entry, 'Db2-tq-01', is also highlighted with a red box.



Manage the database instance

Resource list /

Db2-tq-01 Active Add tags

Details Actions...

Getting started

Manage Selected

Service credentials

Plan

Connections

Open Console

Getting Started

Where can I find my credentials?
Get your username and password by clicking the "Service Credentials" link to the left and selecting "New Credentials".

Getting Started

Need Help?

Use IBM dW Answers to view recently asked questions or ask your own. Still unable to find an answer? Submit a Bluemix Support Ticket to our team.

IBM dW Answers **Support Ticket**

Create new service credentials

Resource list /

Db2-tq-01 • Active Add tags

Details Actions...

Getting started

Manage

Service credentials

You can generate a new set of credentials for cases where you want to manually connect an app or external consumer to an IBM Cloud™ service. [Learn more](#)

Search credentials...

New credential

Key name	Date created
No service credentials	

Credentials are provided in JSON format. The JSON snippet lists credentials, such as the API key and secret, as well as connection information for the service.

Service credentials

Resource list / Db2-tq-01 Active Add tags 

Details Actions... 

Getting started You can generate a new set of credentials for cases where you want to manually connect an app or external consumer to an IBM Cloud™ service. [Learn more](#)

Manage

Service credentials

Search credentials... 

Key name Date created

Key name	Date created	Actions
Service credentials-1	MAY 4, 2020 · 04:29:29 PM	 

```
[{"db": "BLUDB", "dsn": "DATABASE=BLUDB;HOSTNAME=dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net;PORT=50000;PROTOCOL=TCPIP;UID=lct12330;PWD=zgvrlm1mbzv+pgg;", "host": "dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net", "hostname": "dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net", "https_url": "https://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net", "jdbcurl": "jdbc:db2://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50000/BLUDB", "parameters": {}, "password": "REDACTED", "port": 50000, "ssldsn": "DATABASE=BLUDB;HOSTNAME=dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net;PORT=50001;PROTOCOL=TCPIP;UID=lct12330;PWD=zgvrlm1mbzv+pgg;Security=SSL;", "ssljdbcurl": "jdbc:db2://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50001/BLUDB:sslConnection=true;", "uri": "db2://lct12330:zgvrlm1mbzv%2Bpgg@dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50000/BLUDB", "username": "lct12330"}]
```

Service credentials

Key name	Date created	
Service credentials-1	MAY 4, 2020 - 04:29:29 PM	

```
{  
  "db": "BLUDB",  
  "dsn": "DATABASE=BLUDB;HOSTNAME=dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net;PORT=50000;PROTOCOL=TCPIP;UID=lct12330;PWD=zgzvrlmlmbzv+pgg;",  
  "host": "dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net",  
  "hostname": "dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net",  
  "https_url": "https://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net",  
  "jdbcurl": "jdbc:db2://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50000/BLUDB",  
  "parameters": {},  
  "password": "████████",  
  "port": 50000,  
  "ssldash": "DATABASE=BLUDB;HOSTNAME=dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net;PORT=50001;PROTOCOL=TCPIP;UID=lct12330;PWD=zgzvrlmlmbzv+pgg;Security=SSL;",  
  "ssljdbcurl": "jdbc:db2://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50001/BLUDB:sslConnection=true;",  
  "uri": "db2://lct12330:zgzvrlmlmbzv%2Bpgg@dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50000/BLUDB",  
  "username": "lct12330"  
}
```

- Database: BLUDB
- Port: 50000

Service credentials

Key name	Date created
Service credentials-1	MAY 4, 2020 - 04:29:29 PM
{	
"db": "BLUDB", "dsn": "DATABASE=BLUDB;HOSTNAME=dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net;PORT=50000;PROTOCOL=TCPPIP;UID=lct12330;PWD=zgzvrlmlmbzv+pgg;", "host": "dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net", "hostname": "dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net", "https_uri": "https://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net", "jdbcurl": "jdbc:db2://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50000/BLUDB", "parameters": {}, "password": "████████", "port": 50000, "ssldsn": "DATABASE=BLUDB;HOSTNAME=dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net;PORT=50001;PROTOCOL=TCPPIP;UID=lct12330;PWD=zgzvrlmlmbzv+pgg;Security=SSL;", "ssljdbcurl": "jdbc:db2://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50001/BLUDB:sslConnection=true;", "uri": "db2://lct12330:zgzvrlmlmbzv%2Bpgg@dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50000/BLUDB", "username": "lct12330" }	

- **Database:** BLUDB
- **Port:** 50000
- **Hostname:** dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net

Service credentials

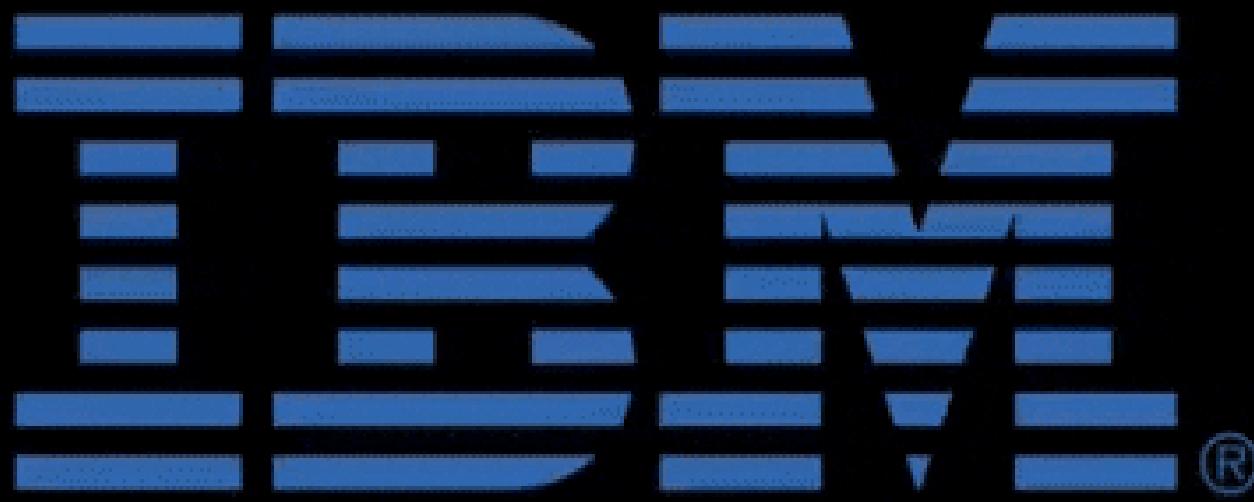
Key name	Date created	
Service credentials-1	MAY 4, 2020 - 04:29:29 PM	
{		
"db": "BLUDB", "dsn": "DATABASE=BLUDB;HOSTNAME=dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net;PORT=50000;PROTOCOL=TCPIP;UID=lct12330;PWD=zgzvrlmlmbzv+pgg;", "host": "dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net", "hostname": "dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net", "https_url": "https://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net", "jdbcurl": "jdbc:db2://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50000/BLUDB", "parameters": {}, "password": "████████", "port": 50000, "ssldsn": "DATABASE=BLUDB;HOSTNAME=dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net;PORT=50001;PROTOCOL=TCPIP;UID=lct12330;PWD=zgzvrlmlmbzv+pgg;Security=SSL;", "ssljdbcurl": "jdbc:db2://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50001/BLUDB:sslConnection=true;", "uri": "db2://lct12330:zgzvrlmlmbzv%2Bpgg@dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50000/BLUDB", "username": "lct12330"		

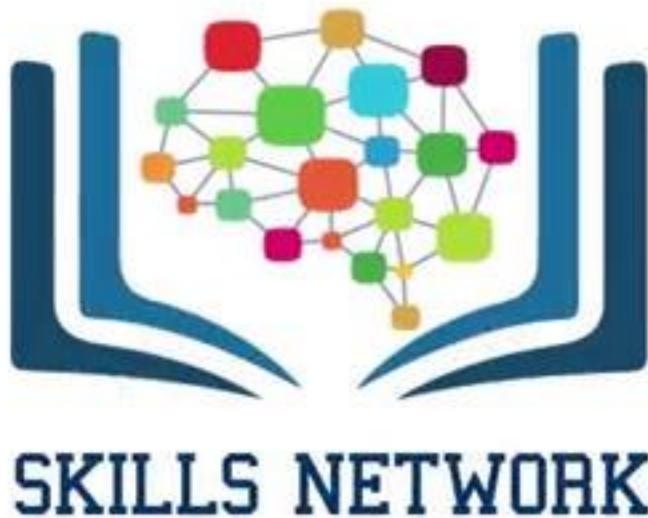
- Database: BLUDB
- Port: 50000
- Hostname: dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net
- Username: lct12330

Service credentials

Key name	Date created
Service credentials-1	MAY 4, 2020 - 04:29:29 PM
{	
"db": "BLUDB", "dsn": "DATABASE=BLUDB;HOSTNAME=dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net;PORT=50000;PROTOCOL=TCPIP;UID=lct12330;PWD=zgzvrlmlmbzv+pgg;", "host": "dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net", "hostname": "dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net", "https_url": "https://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net", "jdbcurl": "jdbc:db2://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50000/BLUDB", "parameters": {}, "password": "*****", "port": 50000, "ssldsn": "DATABASE=BLUDB;HOSTNAME=dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net;PORT=50001;PROTOCOL=TCPIP;UID=lct12330;PWD=zgzvrlmlmbzv+pgg;Security=SSL;", "ssljdbcurl": "jdbc:db2://dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50001/BLUDB:sslConnection=true;", "uri": "db2://lct12330:zgzvrlmlmbzv%2Bpgg@dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net:50000/BLUDB", "username": "lct12330" }	

- Database: BLUDB
- Port: 50000
- Hostname: dashdb-txn-sbox-yp-dal09-04.services.dal.bluemix.net
- Username: lct12330
- Password: *****





Types of SQL statements (DDL vs. DML)

IBM Developer

Types of SQL statements

DDL vs. DML

© IBM Corporation. All rights reserved.

Types of SQL Statements - DDL

- SQL Statement types: DDL and DML
- DDL (Data Definition Language) statements:
 - Define, change, or drop data

Types of SQL Statements - DDL

- SQL Statement types: DDL and DML
- DDL (Data Definition Language) statements:
 - Define, change, or drop data
- Common DDL:
 - CREATE

Types of SQL Statements - DDL

- SQL Statement types: DDL and DML
- DDL (Data Definition Language) statements:
 - Define, change, or drop data
- Common DDL:
 - CREATE
 - ALTER

Types of SQL Statements - DDL

- SQL Statement types: DDL and DML
- DDL (Data Definition Language) statements:
 - Define, change, or drop data
- Common DDL:
 - CREATE
 - ALTER
 - TRUNCATE

Types of SQL Statements - DDL

- SQL Statement types: DDL and DML
- DDL (Data Definition Language) statements:
 - Define, change, or drop data
- Common DDL:
 - CREATE
 - ALTER
 - TRUNCATE
 - DROP

Types of SQL Statements - DML

- DML (Data Manipulation Language) statements:
 - Read and modify data

Types of SQL Statements - DML

- DML (Data Manipulation Language) statements:
 - Read and modify data
 - CRUD operations (Create, Read, Update & Delete rows)

Types of SQL Statements - DML

- DML (Data Manipulation Language) statements:
 - Read and modify data
 - CRUD operations (Create, Read, Update & Delete rows)
- Common DML:

Types of SQL Statements - DML

- DML (Data Manipulation Language) statements:
 - Read and modify data
 - CRUD operations (Create, Read, Update & Delete rows)
- Common DML:
 - INSERT

Types of SQL Statements - DML

- DML (Data Manipulation Language) statements:
 - Read and modify data
 - CRUD operations (Create, Read, Update & Delete rows)
- Common DML:
 - INSERT
 - SELECT

Types of SQL Statements - DML

- DML (Data Manipulation Language) statements:
 - Read and modify data
 - CRUD operations (Create, Read, Update & Delete rows)
- Common DML:
 - INSERT
 - SELECT
 - UPDATE

Types of SQL Statements - DML

- DML (Data Manipulation Language) statements:
 - Read and modify data
 - CRUD operations (Create, Read, Update & Delete rows)
- Common DML:
 - INSERT
 - SELECT
 - UPDATE
 - DELETE

Summary

Now you know that:

- DDL used for defining objects (tables)
- DML used for manipulating data in tables



CREATE TABLE Statement

IBM Developer

CREATE TABLE Statement

© IBM Corporation. All rights reserved.

CREATE table

- **Syntax:**

```
CREATE TABLE table_name
(
    column_name_1 datatype optional_parameters,
    column_name_2 datatype,
    ...
    column_name_n datatype
)
```

EXAMPLE

- Create a table for Canadian provinces

```
CREATE TABLE provinces(  
    id char(2) PRIMARY KEY NOT NULL,  
    name varchar(24)  
)
```

EXAMPLE

- Create a table for Canadian provinces

```
CREATE TABLE provinces(  
    id char(2) PRIMARY KEY NOT NULL,  
    name varchar(24)  
)
```

id
char(2)

EXAMPLE

- Create a table for Canadian provinces

```
CREATE TABLE provinces(  
    id char(2) PRIMARY KEY NOT NULL,  
    name varchar(24)  
)
```

id
char(2)

AB

BC

...

EXAMPLE

- Create a table for Canadian provinces

```
CREATE TABLE provinces(  
    id char(2) PRIMARY KEY NOT NULL,  
    name varchar(24)  
)
```

id <i>char(2)</i>	name <i>varchar(24)</i>
AB	
BC	
...	

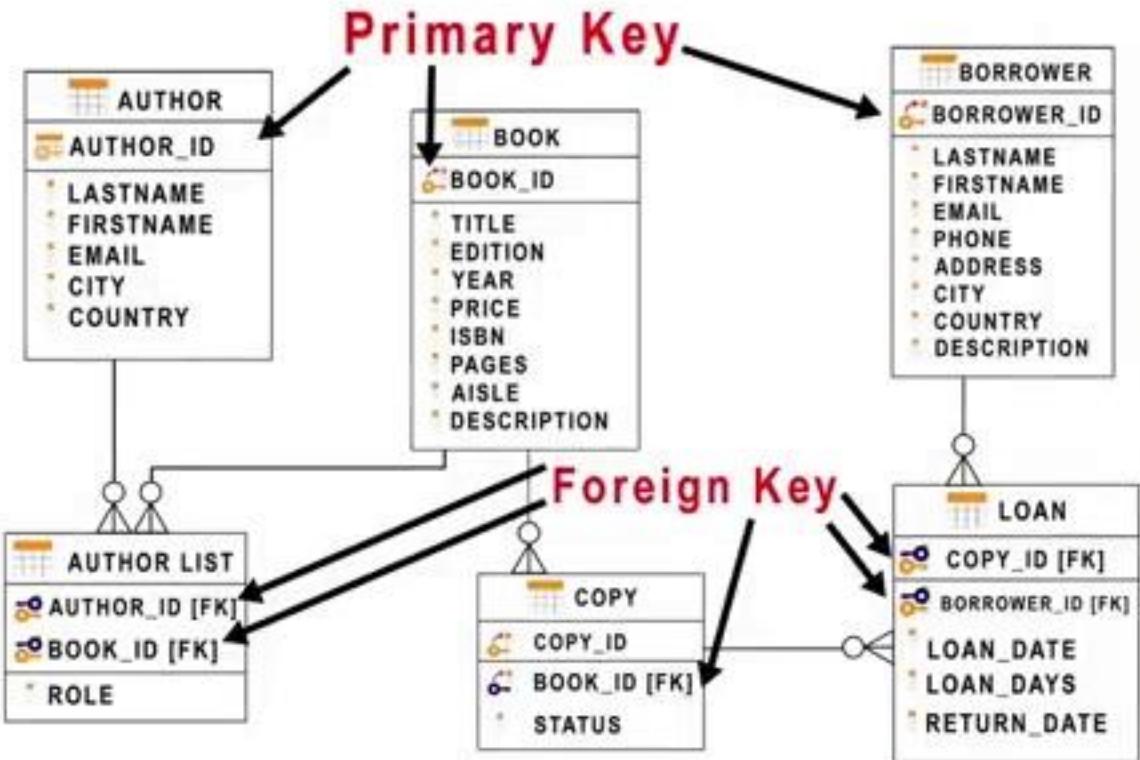
EXAMPLE

- Create a table for Canadian provinces

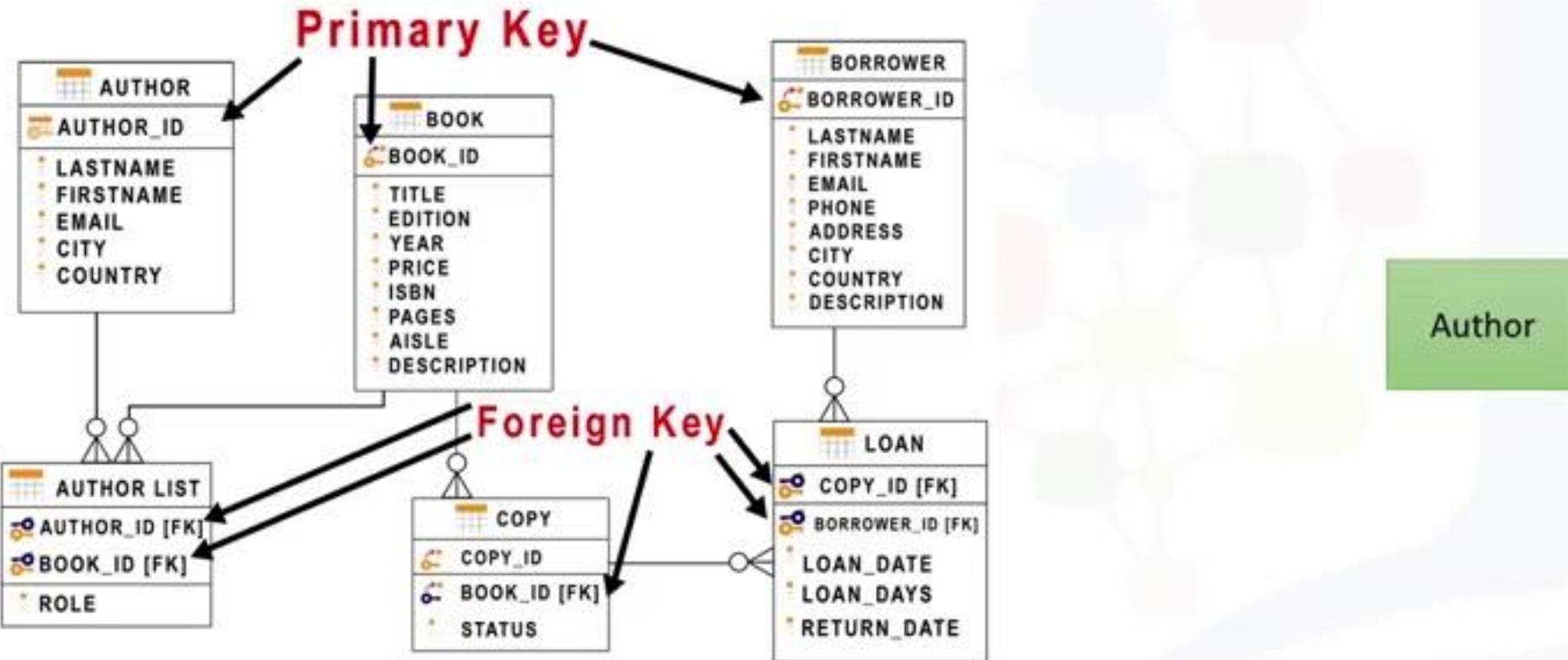
```
CREATE TABLE provinces(  
    id char(2) PRIMARY KEY NOT NULL,  
    name varchar(24)  
)
```

id <i>char(2)</i>	name <i>varchar(24)</i>
AB	ALBERTA
BC	BRITISH COLUMBIA
...	...

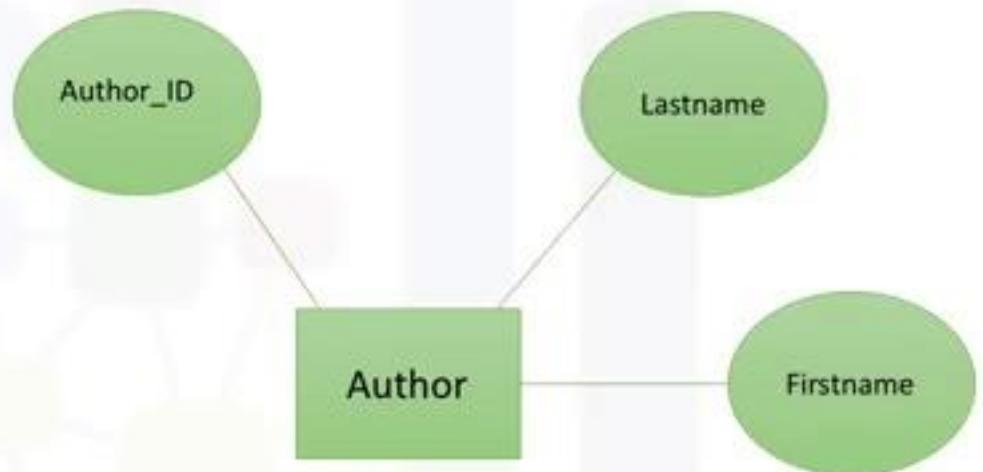
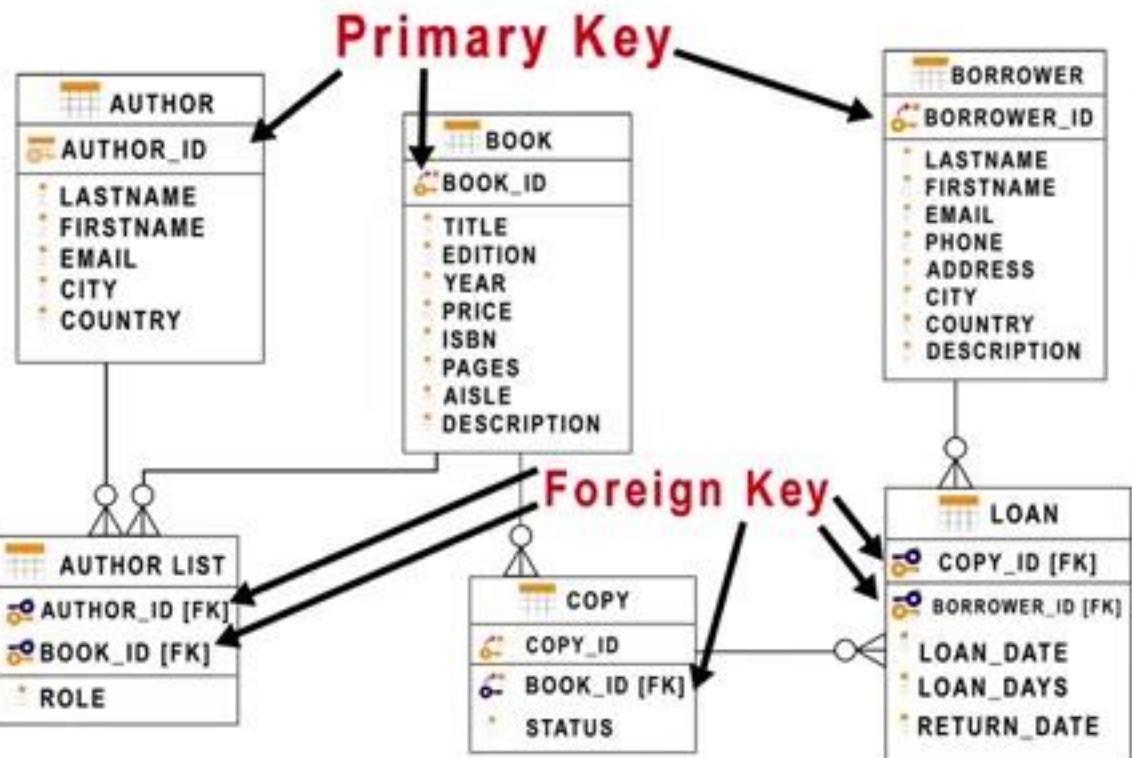
Create a table



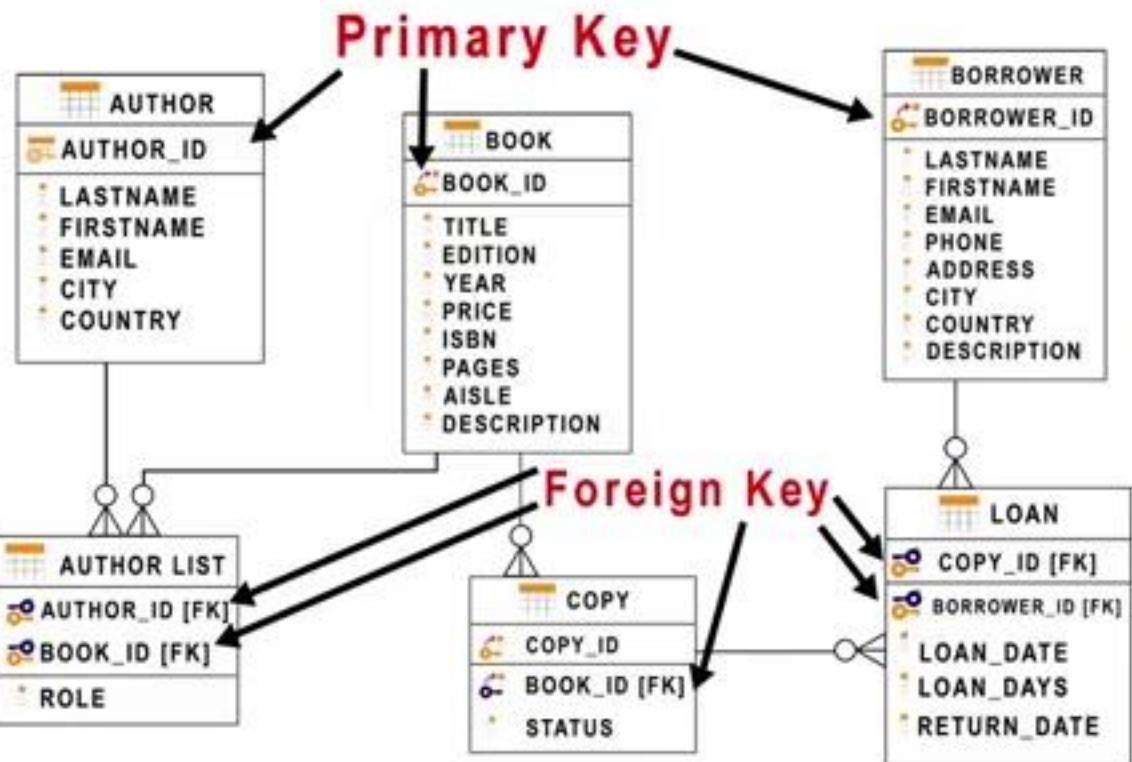
Create a table



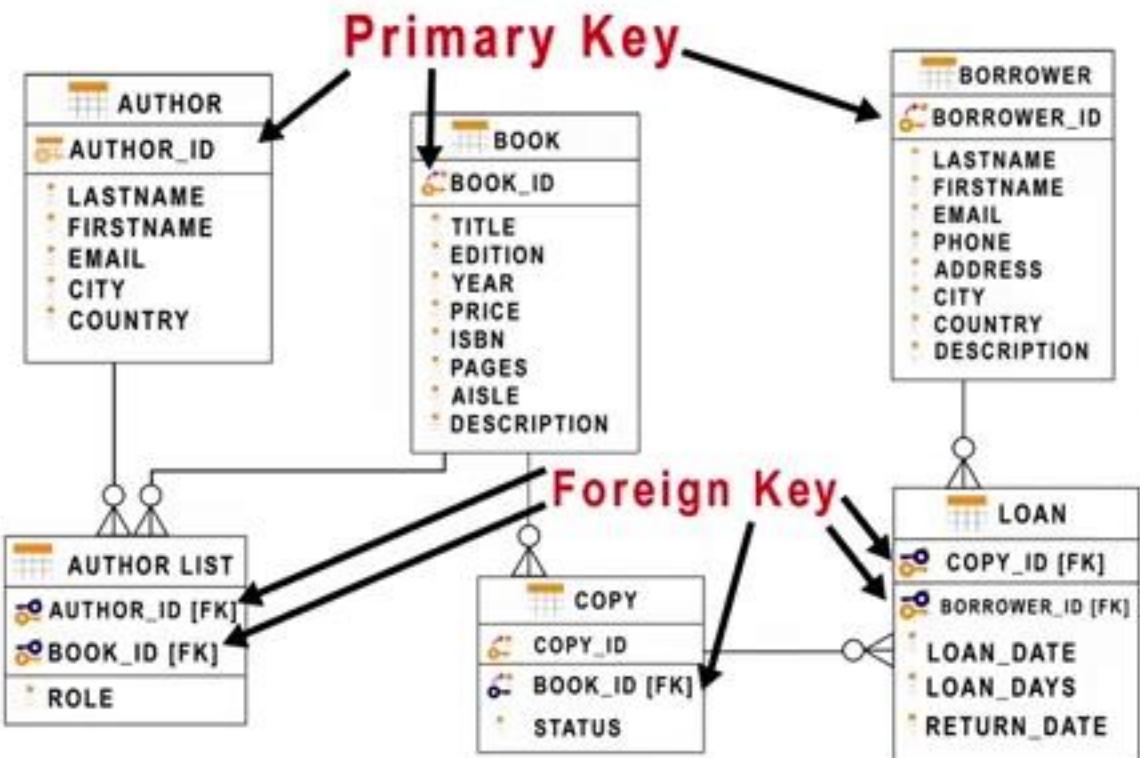
Create a table



Create a table



Create a table



Primary Key: Uniquely Identifies each Row in a Table

CREATE TABLE Statement

To create the Author table, use the following columns and datatypes:

AUTHOR(Author_ID:char, Lastname:varchar, Firstname:varchar, Email:varchar, City:varchar, Country:char)

CREATE TABLE Statement

To create the Author table, use the following columns and datatypes:

AUTHOR(Author_ID:char, Lastname:varchar, Firstname:varchar, Email:varchar, City:varchar, Country:char)

```
CREATE TABLE author (
```

CREATE TABLE Statement

To create the Author table, use the following columns and datatypes:

AUTHOR(Author_ID:char, Lastname:varchar, Firstname:varchar, Email:varchar, City:varchar, Country:char)

```
CREATE TABLE author (
    author_id CHAR(2) PRIMARY KEY NOT NULL,
```

CREATE TABLE Statement

To create the Author table, use the following columns and datatypes:

AUTHOR(Author_ID:char, Lastname:varchar, Firstname:varchar, Email:varchar, City:varchar, Country:char)

```
CREATE TABLE author (
    author_id CHAR(2) PRIMARY KEY NOT NULL,
    lastname VARCHAR(15) NOT NULL,
```

CREATE TABLE Statement

To create the Author table, use the following columns and datatypes:

AUTHOR(Author_ID:char, Lastname:varchar, Firstname:varchar, Email:varchar, City:varchar, Country:char)

```
CREATE TABLE author (
    author_id CHAR(2) PRIMARY KEY NOT NULL,
    lastname VARCHAR(15) NOT NULL,
    firstname VARCHAR(15) NOT NULL,
```

CREATE TABLE Statement

To create the Author table, use the following columns and datatypes:

AUTHOR(Author_ID:char, Lastname:varchar, Firsname:varchar, Email:varchar, City:varchar, Country:char)

```
CREATE TABLE author (
    author_id CHAR(2) PRIMARY KEY NOT NULL,
    lastname VARCHAR(15) NOT NULL,
    firstname VARCHAR(15) NOT NULL,
    email VARCHAR(40),
    city VARCHAR(15),
    country CHAR(2)
```

CREATE TABLE Statement

To create the Author table, use the following columns and datatypes:

AUTHOR(Author_ID:char, Lastname:varchar, Firstname:varchar, Email:varchar, City:varchar, Country:char)

```
CREATE TABLE author (
    author_id CHAR(2) PRIMARY KEY NOT NULL,
    lastname VARCHAR(15) NOT NULL,
    firstname VARCHAR(15) NOT NULL,
    email VARCHAR(40),
    city VARCHAR(15),
    country CHAR(2)
)
```

CREATE TABLE Statement

To create the Author table, use the following columns and datatypes:

AUTHOR(Author_ID:char, Lastname:varchar, Firstname:varchar, Email:varchar, City:varchar, Country:char)

```
CREATE TABLE author (
    author_id CHAR(2) PRIMARY KEY NOT NULL,
    lastname VARCHAR(15) NOT NULL,
    firstname VARCHAR(15) NOT NULL,
    email VARCHAR(40),
    city VARCHAR(15),
    country CHAR(2)
)
```

CREATE TABLE Statement

To create the Author table, use the following columns and datatypes:

AUTHOR(Author_ID:char, Lastname:varchar, Firstname:varchar, Email:varchar, City:varchar, Country:char)

```
CREATE TABLE author (
    author_id CHAR(2) PRIMARY KEY NOT NULL,
    lastname VARCHAR(15) NOT NULL,
    firstname VARCHAR(15) NOT NULL,
    email VARCHAR(40),
    city VARCHAR(15),
    country CHAR(2)
)
```

Summary

Now you know that:

Summary

Now you know that:

- CREATE used for creating entities (tables) in a relational database

Summary

Now you know that:

- CREATE used for creating entities (tables) in a relational database
- CREATE TABLE statement includes definition of attributes (columns):

Summary

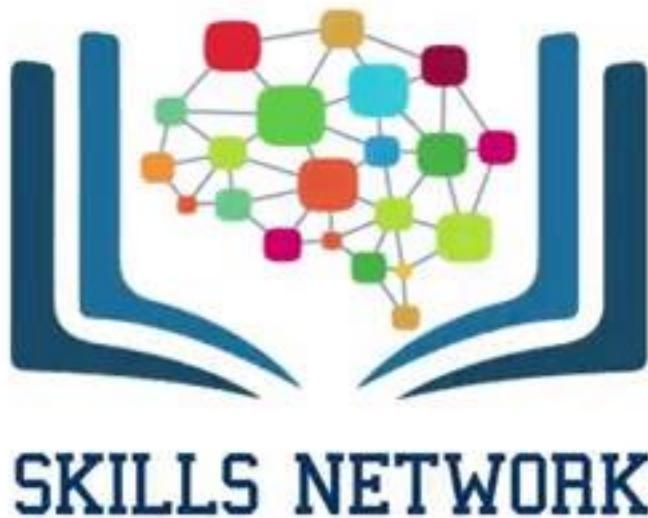
Now you know that:

- CREATE used for creating entities (tables) in a relational database
- CREATE TABLE statement includes definition of attributes (columns):
 - Names of columns

Summary

Now you know that:

- CREATE used for creating entities (tables) in a relational database
- CREATE TABLE statement includes definition of attributes (columns):
 - Names of columns
 - Datatypes of columns
 - Constraints (e.g. Primary Key)



ALTER, DROP, and TRUNCATE Tables

IBM Developer

ALTER, DROP, and TRUNCATE Tables

© IBM Corporation. All rights reserved.

ALTER TABLE ... ADD COLUMN

ALTER TABLE ... ADD COLUMN

- Add or remove columns

ALTER TABLE ... ADD COLUMN

- Add or remove columns
- Modify the data type of columns
- Add or remove keys
- Add or remove constraints

ALTER TABLE ... ADD COLUMN

- Add or remove columns
- Modify the data type of columns
- Add or remove keys
- Add or remove constraints

```
ALTER TABLE <table_name>
    ADD COLUMN <column_name_1> datatype
    ...
    ADD COLUMN <column_name_n> datatype;
```

ALTER TABLE ... ADD COLUMN

- Add or remove columns
- Modify the data type of columns
- Add or remove keys
- Add or remove constraints

```
ALTER TABLE <table_name>
    ADD COLUMN <column_name_1> datatype
    ...
    ADD COLUMN <column_name_n> datatype;
```

ALTER TABLE ... ADD COLUMN

- Add or remove columns
- Modify the data type of columns
- Add or remove keys
- Add or remove constraints

```
ALTER TABLE <table_name>
    ADD COLUMN <column_name_1> datatype
    ...
    ADD COLUMN <column_name_n> datatype;
```

ALTER TABLE ... ADD COLUMN

```
ALTER TABLE author  
ADD COLUMN telephone_number BIGINT;
```

author_id	lastna me	firstna me	email	city	country
1001	Thomas	John	johnt@...	New York	USA
1002	James	Alice	alicej@...	Seattle	USA
1003	Wells	Steve	stevew:@...	Montreal	Canada
1004	Kumar	Santosh	kumars@...	London	UK

ALTER TABLE ... ADD COLUMN

```
ALTER TABLE author  
ADD COLUMN telephone_number BIGINT;
```

author_id	lastna me	firstna me	email	city	country
1001	Thomas	John	johnt@...	New York	USA
1002	James	Alice	alicej@...	Seattle	USA
1003	Wells	Steve	stevew:@...	Montreal	Canada
1004	Kumar	Santosh	kumars@...	London	UK

ALTER TABLE ... ADD COLUMN

```
ALTER TABLE author  
ADD COLUMN telephone_number BIGINT;
```

author_id	lastna me	firstna me	email	city	country	telepho ne_numb er
1001	Thomas	John	johnt@...	New York	USA	5551111
1002	James	Alice	alicej@...	Seattle	USA	5551112
1003	Wells	Steve	stevew:@...	Montreal	Canada	5552222
1004	Kumar	Santosh	kumars@...	London	UK	5553333

ALTER TABLE ... ADD COLUMN

```
ALTER TABLE author  
ADD COLUMN telephone_number BIGINT;
```

author_id	lastna me	firstna me	email	city	country	telepho ne_numb er
1001	Thomas	John	johnt@...	New York	USA	5551111
1002	James	Alice	alicej@...	Seattle	USA	5551112
1003	Wells	Steve	stevew:@...	Montreal	Canada	5552222
1004	Kumar	Santosh	kumars@...	London	UK	5553333

ALTER TABLE ... ALTER COLUMN

```
ALTER TABLE <table_name>
    ALTER COLUMN <column_name> SET DATA TYPE
    <datatype>;
```

ALTER TABLE ... ALTER COLUMN

```
ALTER TABLE author  
    ALTER COLUMN telephone_number SET DATA TYPE  
CHAR(20);
```

author_id	lastna me	firstna me	email	city	country	telepho ne_numb er
1001	Thomas	John	johnt@...	New York	USA	555-1111
1002	James	Alice	alicej@...	Seattle	USA	555-1112
1003	Wells	Steve	stevew@...	Montreal	Canada	555-2222
1004	Kumar	Santosh	kumars@...	London	UK	555-3333

ALTER TABLE ... ALTER COLUMN

```
ALTER TABLE author
```

```
    ALTER COLUMN telephone_number SET DATA TYPE  
    CHAR(20);
```

author_id	lastna me	firstna me	email	city	country	telepho ne_numb er
1001	Thomas	John	johnt@...	New York	USA	555-1111
1002	James	Alice	alicej@...	Seattle	USA	555-1112
1003	Wells	Steve	stevew@...	Montreal	Canada	555-2222
1004	Kumar	Santosh	kumars@...	London	UK	555-3333

ALTER TABLE ... ALTER COLUMN

```
ALTER TABLE author
```

```
    ALTER COLUMN telephone_number SET DATA TYPE  
    CHAR(20);
```

author_id	lastna me	firstna me	email	city	country	telepho ne_numb er
1001	Thomas	John	johnt@...	New York	USA	555-1111
1002	James	Alice	alicej@...	Seattle	USA	555-1112
1003	Wells	Steve	stevew@...	Montreal	Canada	555-2222
1004	Kumar	Santosh	kumars@...	London	UK	555-3333

ALTER TABLE ... DROP COLUMN

```
ALTER TABLE author  
DROP COLUMN telephone_number;
```

author_id	lastna me	firstna me	email	city	country	telepho ne_numb er
1001	Thomas	John	johnt@...	New York	USA	555-1111
1002	James	Alice	alicej@...	Seattle	USA	555-1112
1003	Wells	Steve	stevew:@...	Montreal	Canada	555-2222
1004	Kumar	Santosh	kumars@...	London	UK	555-3333

ALTER TABLE . . . DROP COLUMN

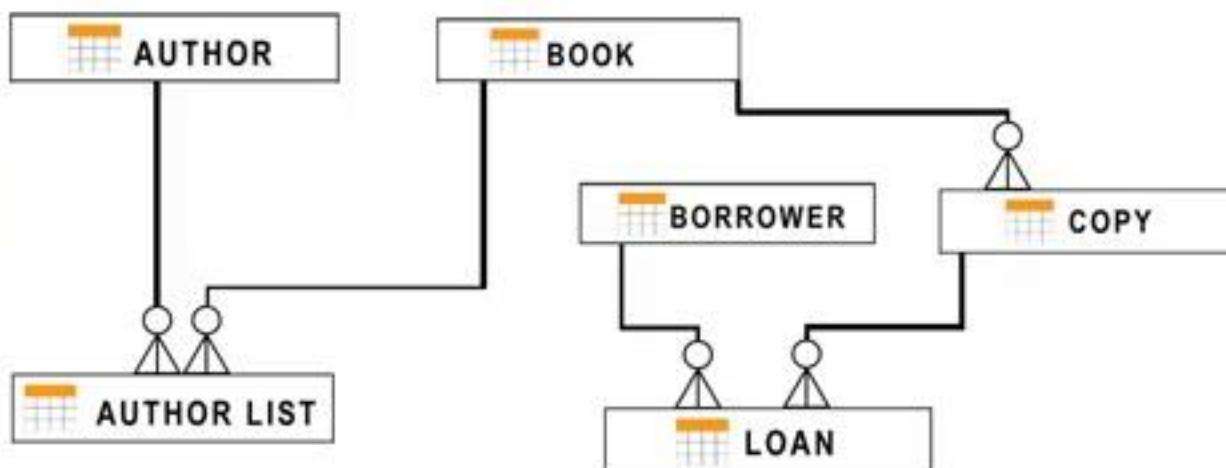
```
ALTER TABLE author  
DROP COLUMN telephone_number;
```

author_id	lastna me	firstna me	email	city	country	telepho ne_numb er
1001	Thomas	John	johnt@...	New York	USA	555-1111
1002	James	Alice	alicej@...	Seattle	USA	555-1112
1003	Wells	Steve	stevew:@...	Montreal	Canada	555-2222
1004	Kumar	Santosh	kumars@...	London	UK	555-3333

DROP TABLE

```
DROP TABLE <table_name>;
```

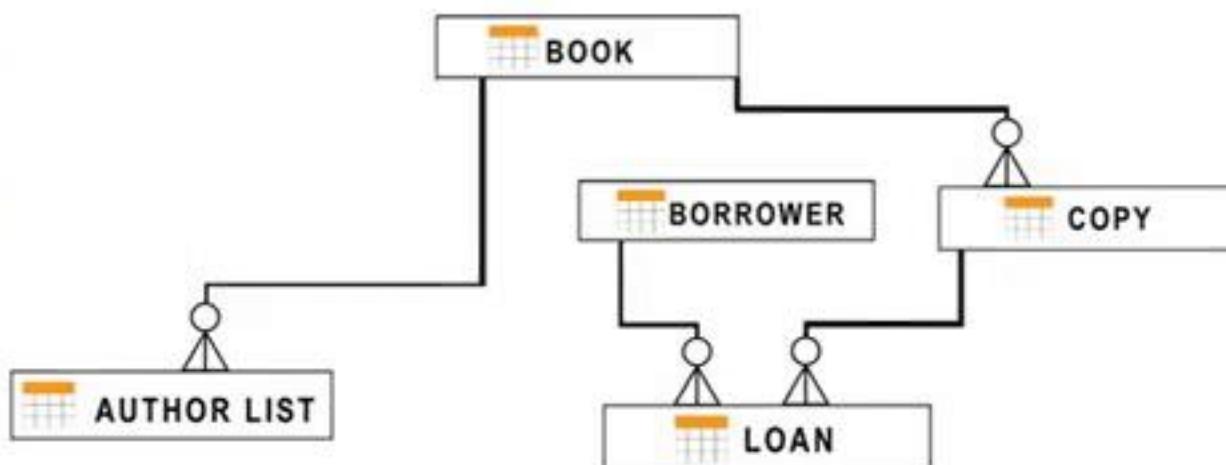
```
DROP TABLE author;
```



DROP TABLE

```
DROP TABLE <table_name>;
```

```
DROP TABLE author;
```



TRUNCATE TABLE

```
TRUNCATE TABLE <table_name>
IMMEDIATE;
```

TRUNCATE TABLE

```
TRUNCATE TABLE <table_name>
IMMEDIATE;
```

TRUNCATE TABLE

```
TRUNCATE TABLE author  
IMMEDIATE;
```

author_id	lastna me	firstna me	email	city	country
1001	Thomas	John	johnt@...	New York	USA
1002	James	Alice	alicej@...	Seattle	USA
1003	Wells	Steve	stevew:@...	Montreal	Canada
1004	Kumar	Santosh	kumars@...	London	UK

TRUNCATE TABLE

```
TRUNCATE TABLE author  
IMMEDIATE;
```

author_id	lastna me	firstna me	email	city	country