Chapter 11

How to create databases, tables, and indexes



Objectives

Applied

- 1. Given the design for a database, write the DDL statements to create the tables, constraints, and indexes that are required.
- 2. Write a script that includes all of the DDL statements for creating the tables of a database.
- 3. Use MySQL Workbench to work with the columns, data, constraints, and indexes for a table.



Objectives (continued)

Knowledge

- 1. Describe how each of these types of constraints restricts the values that can be stored in a table: not null, unique, primary key, and foreign key.
- 2. Describe the difference between a column-level constraint and a table-level constraint.
- 3. Describe the use of an index.
- 4. Describe the use of a script for creating the tables of a database.
- 5. Describe three character sets that are commonly used with MySQL and the pros and cons of each character set.
- 6. Describe how a collation works with a character set.
- 7. Describe two storage engines that are commonly used with MySQL and the pros and cons of each engine.



How to use the CREATE DATABASE statement

Syntax

CREATE DATABASE [IF NOT EXISTS] db_name

Attempt to create a database named AP

CREATE DATABASE ap

Create a database named AP only if it doesn't exist

CREATE DATABASE IF NOT EXISTS ap



How to use the DROP DATABASE statement

Syntax

DROP DATABASE [IF EXISTS] db name

Attempt to drop a database named AP

DROP DATABASE ap

Drop a database named AP only if it exists

DROP DATABASE IF EXISTS ap



How to use the USE statement

Syntax

USE db_name

Select a database named AP

USE ap



The syntax of the CREATE TABLE statement

```
CREATE TABLE [db_name.]table_name
(
   column_name_1 data_type [column_attributes]
   [, column_name_2 data_type [column_attributes]]...
   [, table_level_constraints]
)
```

Common column attributes

```
NOT NULL
UNIQUE
DEFAULT default_value
AUTO_INCREMENT
```



A statement that creates a table without column attributes

```
CREATE TABLE vendors
(
  vendor_id INT,
  vendor_name VARCHAR(50)
)
```

A statement that creates a table with column attributes



Another statement that creates a table with column attributes

```
CREATE TABLE invoices
  invoice id
                  INT
                                 NOT NULL
                                            UNIQUE,
 vendor id
                  INT
                                 NOT NULL,
  invoice number VARCHAR(50)
                                 NOT NULL,
  invoice date DATE,
  invoice total DECIMAL(9,2)
                                 NOT NULL,
                  DECIMAL (9,2)
 payment total
                                            DEFAULT 0
```



The syntax of a column-level primary key constraint

column_name data_type PRIMARY KEY column_attributes

A table with column-level constraints



The syntax of a table-level primary key constraint

```
[CONSTRAINT [constraint_name]]
PRIMARY KEY (column name 1[, column name 2]...)
```

A table with table-level constraints



A table with a two-column primary key constraint



The syntax of a column-level foreign key constraint

```
[CONSTRAINT] REFERENCES table_name (column_name)
[ON DELETE {CASCADE|SET NULL}]
```

A table with a column-level foreign key constraint



The syntax of a table-level foreign key constraint

A table with a table-level foreign key constraint



An INSERT statement that fails because a related row doesn't exist

```
INSERT INTO invoices
VALUES (1, 1, '1')
```

The response from the system

```
Error Code: 1452. Cannot add or update a child row: a
foreign key constraint fails ('ex'.'invoices', CONSTRAINT
'invoices_fk_vendors' FOREIGN KEY ('vendor_id')
REFERENCES 'vendors' ('vendor_id'))
```



A constraint that uses the ON DELETE clause

```
CONSTRAINT invoices_fk_vendors

FOREIGN KEY (vendor_id) REFERENCES vendors (vendor_id)

ON DELETE CASCADE
```



Terms to know about constraints

- Column-level constraint
- Table-level constraint
- Not null constraint
- Unique constraint
- Primary key constraint
- Foreign key constraint



The syntax for modifying the columns of a table



A statement that adds a new column

ALTER TABLE vendors
ADD last_transaction_date DATE

A statement that drops a column

ALTER TABLE vendors
DROP COLUMN last_transaction_date



A statement that changes the length of a column

ALTER TABLE vendors
MODIFY vendor name VARCHAR(100) NOT NULL

A statement that changes the type of a column

ALTER TABLE vendors
MODIFY vendor_name CHAR(100) NOT NULL

A statement that changes the default value

ALTER TABLE vendors

MODIFY vendor_name VARCHAR(100) NOT NULL

DEFAULT 'New Vendor'



A statement that changes the name of a column

ALTER TABLE vendors
RENAME COLUMN vendor name TO v name

A statement that fails because it would lose data

```
ALTER TABLE vendors
MODIFY v name VARCHAR(10) NOT NULL
```

The response from the system

Error Code: 1265. Data truncated for column 'v_name' at row 1



The syntax for modifying the constraints of a table

```
ALTER TABLE [dbname.]table_name
{
ADD PRIMARY KEY constraint_definition |
ADD [CONSTRAINT constraint_name]
    FOREIGN KEY constraint_definition |
DROP PRIMARY KEY |
DROP FOREIGN KEY constraint_name
}
```



A statement that adds a primary key constraint

```
ALTER TABLE vendors
ADD PRIMARY KEY (vendor id)
```

A statement that adds a foreign key constraint

```
ALTER TABLE invoices
ADD CONSTRAINT invoices_fk_vendors
FOREIGN KEY (vendor_id) REFERENCES vendors (vendor_id)
```



A statement that drops a primary key constraint

ALTER TABLE vendors
DROP PRIMARY KEY

A statement that drops a foreign key constraint

ALTER TABLE invoices
DROP FOREIGN KEY invoices_fk_vendors



A statement that renames a table

RENAME TABLE vendors TO vendor

A statement that deletes all data from a table

TRUNCATE TABLE vendor

A statement that deletes a table from the current database

DROP TABLE vendor



A statement that qualifies the table to be deleted

DROP TABLE ex.vendor

A statement that returns an error due to a foreign key reference

DROP TABLE vendors

The response from the system

```
Error Code: 3730. Cannot drop table 'vendors' referenced by a foreign key constraint 'invoices_fk_vendors' on table 'invoices'
```



The syntax of the CREATE INDEX statement

```
CREATE [UNIQUE] INDEX index_name

ON [dbname.]table_name (column_name_1 [ASC|DESC][,

column_name_2 [ASC|DESC]]...)
```

A statement that creates an index based on a single column

```
CREATE INDEX invoices_invoice_date_ix
ON invoices (invoice date)
```

A statement that creates an index based on two columns

```
CREATE INDEX invoices_vendor_id_invoice_number_ix
ON invoices (vendor_id, invoice_number)
```



A statement that creates a unique index

```
CREATE UNIQUE INDEX vendors_vendor_phone_ix
ON vendors (vendor_phone)
```

A statement that creates an index that's sorted in descending order

```
CREATE INDEX invoices_invoice_total_ix ON invoices (invoice_total DESC)
```

A statement that drops an index

DROP INDEX vendors_vendor_phone_ix ON vendors



The script that creates the AP database (part 1)

```
-- create the database
DROP DATABASE IF EXISTS ap;
CREATE DATABASE ap;
-- select the database
USE ap;
-- create the tables
CREATE TABLE general ledger accounts
  account number
                    INT
                                       PRIMARY KEY,
  account description VARCHAR (50)
                                       UNIQUE
);
CREATE TABLE terms
  terms id
                        INT
                                       PRIMARY KEY
                                       AUTO INCREMENT,
  terms description
                       VARCHAR (50)
                                       NOT NULL,
  terms due days
                        INT
                                       NOT NULL
```



The script that creates the AP database (part 2)

```
CREATE TABLE vendors
 vendor id
                                 INT
                                                PRIMARY KEY
                                                AUTO INCREMENT,
                                VARCHAR (50)
                                                NOT NULL
 vendor name
                                                UNIQUE,
 vendor address1
                                VARCHAR (50),
 vendor address2
                                VARCHAR (50),
 vendor city
                                VARCHAR (50) NOT NULL,
 vendor state
                                CHAR(2)
                                               NOT NULL,
                                VARCHAR (20)
 vendor zip code
                                               NOT NULL,
 vendor phone
                                VARCHAR (50),
 vendor contact last name
                                VARCHAR (50),
 vendor contact first name
                                VARCHAR (50),
  default terms id
                                INT
                                                NOT NULL,
  default account number
                                INT
                                               NOT NULL,
  CONSTRAINT vendors fk terms
    FOREIGN KEY (default terms id)
    REFERENCES terms (terms id),
  CONSTRAINT vendors fk accounts
    FOREIGN KEY (default account number)
    REFERENCES general ledger accounts (account number)
);
```



The script that creates the AP database (part 3)

```
CREATE TABLE invoices
  invoice id
                        INT
                                       PRIMARY KEY
                                       AUTO INCREMENT,
 vendor id
                                       NOT NULL,
                        INT
  invoice number
                        VARCHAR (50)
                                       NOT NULL,
  invoice date
                                       NOT NULL,
                        DATE
  invoice total
                       DECIMAL(9,2)
                                       NOT NULL,
 payment total
                        DECIMAL(9,2)
                                       NOT NULL
                                                     DEFAULT 0,
  credit total
                        DECIMAL (9,2)
                                       NOT NULL
                                                     DEFAULT 0,
  terms id
                        INT
                                       NOT NULL,
                                       NOT NULL,
  invoice due date
                        DATE
 payment date
                        DATE,
  CONSTRAINT invoices fk vendors
    FOREIGN KEY (vendor id)
    REFERENCES vendors (vendor id),
  CONSTRAINT invoices fk terms
    FOREIGN KEY (terms id)
    REFERENCES terms (terms id)
);
```

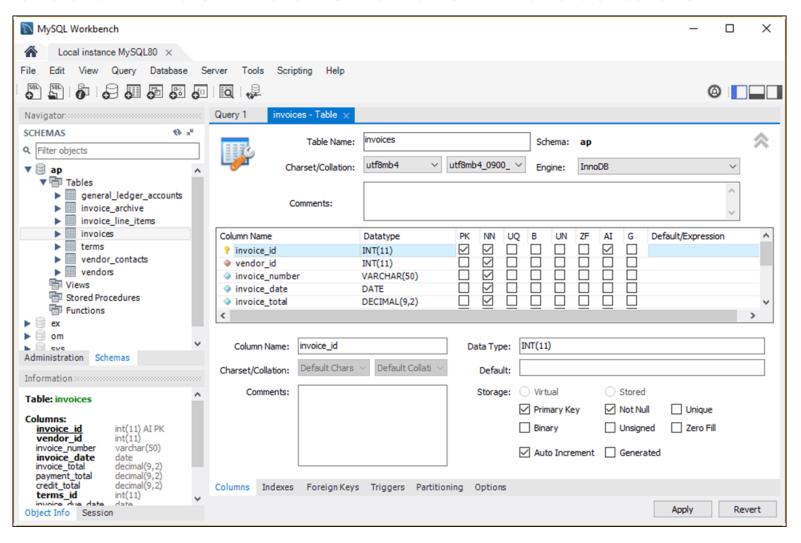


The script that creates the AP database (part 4)

```
CREATE TABLE invoice line items
  invoice id
                         INT
                                        NOT NULL,
  invoice sequence
                       INT
                                        NOT NULL,
  account number
                        INT
                                        NOT NULL,
  line item amount DECIMAL(9,2) NOT NULL,
  line item description VARCHAR(100) NOT NULL,
  CONSTRAINT line items pk
   PRIMARY KEY (invoice id, invoice sequence),
  CONSTRAINT line items fk invoices
   FOREIGN KEY (invoice id)
   REFERENCES invoices (invoice id),
 CONSTRAINT line items fk acounts
   FOREIGN KEY (account number)
   REFERENCES general ledger accounts (account number)
);
-- create an index
CREATE INDEX invoices invoice date ix
 ON invoices (invoice date DESC);
```

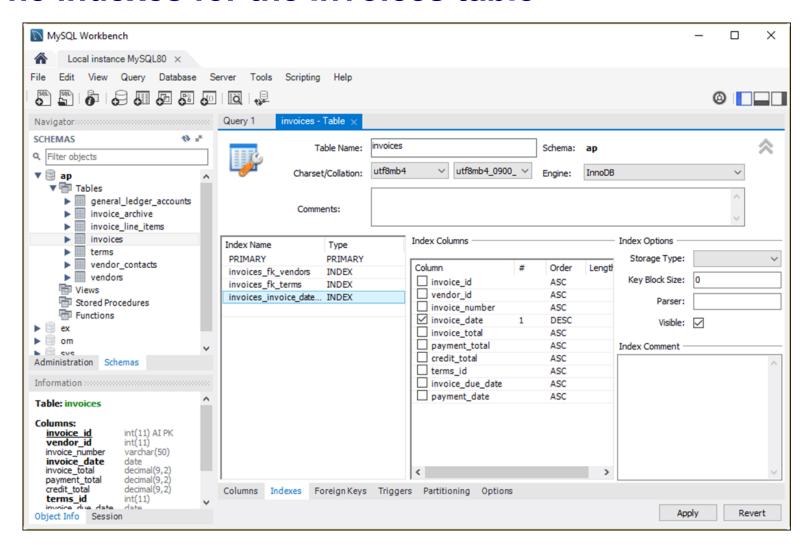


The column definitions for the Invoices table



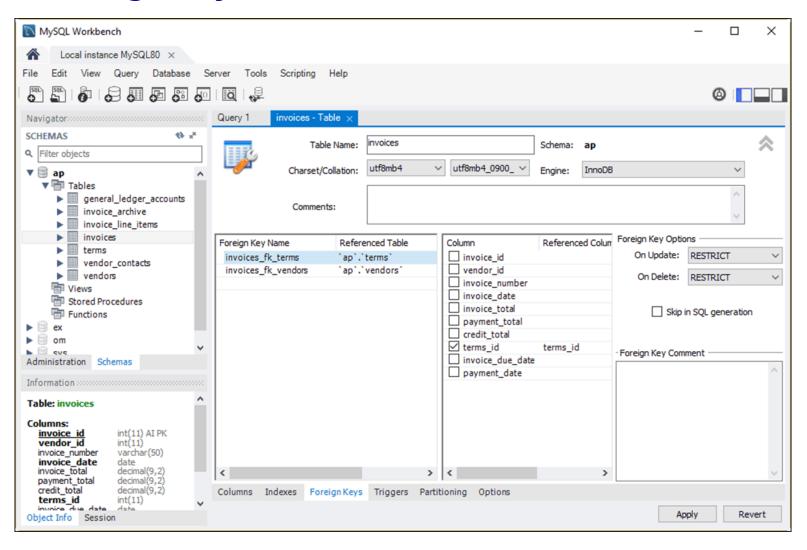


The indexes for the Invoices table





The foreign keys for the Invoices table





Three commonly used character sets

- latin1
- utf8mb3
- utf8mb4



Four collations for the latin1 character set

- latin1_swedish_ci
- latin1_general_ci
- latin1_general_cs
- latin1_bin



Four collations for the utf8mb3 character set

- utf8_general_ci
- utf8_unicode_ci
- utf8_spanish_ci
- utf8_bin



Three collations for the utf8mb4 character set

- utf8mb4_0900_ai_ci
- utf8mb4_0900_as_cs
- utf8mb4_bin



Collation names

- If the name ends with ci, the collation is case-insensitive.
- If the name ends with cs, the collation is case-sensitive.
- If the name includes ai, the collation is accent-insensitive.
- If the name includes as, the collation is accent-sensitive.
- If the name ends with bin, the collation is binary.



How to view all available character sets for a server

SHOW CHARSET

Charset	Description	Default collation	Maxlen
utf16	UTF-16 Unicode	utf16_general_ci	4
utf16le	UTF-16LE Unicode	utf16le_general_ci	4
utf32	UTF-32 Unicode	utf32_general_ci	4
utf8	UTF-8 Unicode	utf8_general_ci	3
utf8mb4	UTF-8 Unicode	utf8mb4_0900_ai_ci	4

How to view a specific character set

SHOW CHARSET LIKE 'utf8mb4'



How to view all available collations for a server

SHOW CHARSET

Collation	Charset	Id	Default	Compiled	Sortlen	Pad_attribute
utf8mb4_0900_ai_ci	utf8mb4	255	Yes	Yes	0	NO PAD
utf8mb4_0900_as_ci	utf8mb4	305		Yes	0	NO PAD
utf8mb4_0900_as_cs	utf8mb4	278		Yes	0	NO PAD
utf8mb4_bin	utf8mb4	46		Yes	1	PAD SPACE
utf8mb4_croatian_ci	utf8mb4	245		Yes	8	PAD SPACE
utf8mb4_cs_0900_ai_ci	utf8mb4	266		Yes	0	NO PAD
utf8mb4_cs_0900_as_cs	utf8mb4	289		Yes	0	NO PAD
utf8mb4_czech_ci	utf8mb4	234		Yes	8	PAD SPACE
utf8mb4_danish_ci	utf8mb4	235		Yes	8	PAD SPACE

How to view all available collations for a specific character set

SHOW CHARSET LIKE 'utf8mb4'



How to view the default character set for a server

SHOW VARIABLES LIKE 'character set server'

How to view the default collation for a server

SHOW VARIABLES LIKE 'collation_server'

How to view the default character set for a database

SHOW VARIABLES LIKE 'character_set_database'

How to view the default collation for a database

SHOW VARIABLES LIKE 'collation_database'



How to view the character set and collation for all the tables in a database

```
SELECT table_name, table_collation
FROM information_schema.tables
WHERE table_schema = 'ap'
```

TABLE_NAME	TABLE_COLLATION
invoice_line_items	utf8mb4_0900_ai_ci
invoices	utf8mb4_0900_ai_ci
terms	utf8mb4_0900_ai_ci



The clauses used to specify a character set and collation

[CHARSET character set] [COLLATE collation]

How to specify a character set and collation at the database level

For a new database

CREATE DATABASE ar CHARSET latin1

COLLATE latin1 general ci

For an existing database

ALTER DATABASE ar CHARSET utf8mb4 COLLATE utf8mb4_0900_ai_ci

For an existing database using the CHARSET clause only

ALTER DATABASE ar CHARSET utf8mb4

For an existing database using the COLLATE clause only

ALTER DATABASE ar COLLATE utf8mb4 0900 ai ci



How to specify a character set and collation at the table level

For a new table

For an existing table

```
ALTER TABLE employees
CHARSET utf8mb4 COLLATE utf8mb4 0900 ai ci
```



How to specify a character set and collation at the column level

For a column in a new table

For a column in an existing table

```
ALTER TABLE employees

MODIFY emp_name VARCHAR(25) CHARSET utf8mb4

COLLATE utf8mb4_0900_ai_ci
```



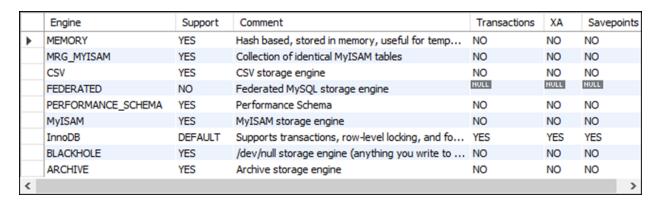
Two commonly used storage engines

- InnoDB
- MyISAM



How to view all storage engines for a server

SHOW ENGINES



How to view the default storage engine for a server

SHOW VARIABLES LIKE 'default storage engine'



How to view the storage engine for all the tables in a database

```
SELECT table_name, engine
FROM information_schema.tables
WHERE table_schema = 'ap'
```

TABLE_NAME	ENGINE
invoice_line_items	InnoDB
invoices	InnoDB
terms	InnoDB



The clause used to specify a storage engine

```
ENGINE = engine_name
```

How to specify a storage engine for a table

For a new table

For an existing table

ALTER TABLE product descriptions ENGINE = InnoDB

How to set the default storage engine for the current session

SET SESSION default_storage_engine = InnoDB

