Experiment 3

Aim: Create a database using Data Definition Language (DDL) and apply integrity constraints for the specified System

Hardware and Software Requirement: P-IV and above, Oracle

Theory:

• **Data-definition language** (DDL). The SQL DDL provides commands for defining relation schemas, deleting relations, and modifying relation schemas.

• **Data-manipulation language** (DML). The SQL DML provides the ability to query information from the database and to insert tuples into, delete tuples from, and modify tuples in the database.

SQL Data Definition

The set of relations in a database must be specified to the system by means of a data-definition language (DDL). The SQL DDL allows specification of not only a set of relations, but also information about each relation, including:

- The schema for each relation.
- The types of values associated with each attribute.
- The integrity constraints.
- The set of indices to be maintained for each relation.
- The security and authorization information for each relation.
- The physical storage structure of each relation on disk

Basic Types

The SQL standard supports a variety of built-in types, including:

• **char**(*n*): A fixed-length character string with user-specified length *n*. The full form, **character**, can be used instead.

• **varchar**(*n*): A variable-length character string with user-specified maximum length *n*.The full form, **character varying**, is equivalent.

• **int**: An integer (a finite subset of the integers that is machine dependent). The full form, **integer**, is equivalent.

- **smallint**: A small integer (a machine-dependent subset of the integer type).
- **numeric**(*p*, *d*): A fixed-point number with user-specified precision. The number consists of *p* digits (plus a sign), and *d* of the *p* digits are to the right of the decimal point. Thus, **numeric**(3,1) allows 44.5 to be stored exactly, but neither 444.5 or 0.32 can be stored exactly in a field of this type.
- **real, double precision**: Floating-point and double-precision floating-point numbers with machine-dependent precision.
- **float**(*n*): A floating-point number, with precision of at least *n* digits.

Each type may include a special value called the **null** value. A null value indicates an absent value that may exist but be unknown or that may not exist at all.

Basic Schema Definition

Create Table Construct

An SQL relation is defined using the **create table** command:

create table r ($A_1\,D_1$, $A_2\,D_2$, ..., $A_n\,D_n$,
(integrity-constraint$_1$),
...,
(integrity-constraint$_k$))

r is the name of the relation

each A_i is an attribute name in the schema of relation r

D_i is the data type of values in the domain of attribute A_i

Drop and Alter Table Constructs

- The drop table command deletes all information about the dropped relation from the database.
- The alter table command is used to add attributes to an existing relation:
  - alter table *r* add *A D*

where *A* is the name of the attribute to be added to relation *r* and *D* is the domain of *A*.

- All tuples in the relation are assigned *null* as the value for the new attribute.

The alter table command can also be used to drop attributes of a relation:

- alter table *r* drop *A*

where *A* is the name of an attribute of relation *r*

TRUNCATE: Remove all records from table ,including spaces allocated for the records are removed.

Syntax:

```
TRUNCATE TABLE <TABLE NAME>;
```

Integrity Constraints

Integrity constraints guard against accidental damage to the database, by ensuring that authorized changes to the database do not result in a loss of data consistency.

A checking account must have a balance greater than $10,000.00

A salary of a bank employee must be at least $4.00 an hour

A customer must have a (non-null) phone number

Constraints on a Single Relation

- **not null**
- **primary key**
- **unique**
- **check** (*P*), where *P* is a predicate

Not Null Constraint

Declare *branch_name* for *branch* is **not null**

branch_name **char**(15) **not null**

The Unique Constraint

unique (A_1, A_2, …, A_m)

The unique specification states that the attributes

*A*1, *A*2, … *Am*
form a candidate key.

Candidate keys are permitted to be null (in contrast to primary keys).

The check clause

check (*P*), where *P* is a predicate

Example: Declare *branch_name* as the primary key for *branch* and ensure that the values of *assets* are non-negative.

```
create table branch
            (branch_name    char(15),
             branch_city          char(30),
             assets          integer,
             primary key (branch_name),
             check (assets >= 0))
```

Referential Integrity

Ensures that a value that appears in one relation for a given set of attributes also appears for a certain set of attributes in another relation.

Example: If "Perryridge" is a branch name appearing in one of the tuples in the *account* relation, then there exists a tuple in the *branch* relation for branch "Perryridge".

Primary and candidate keys and foreign keys can be specified as part of the SQL **create table** statement:

The primary key clause lists attributes that comprise the primary key.

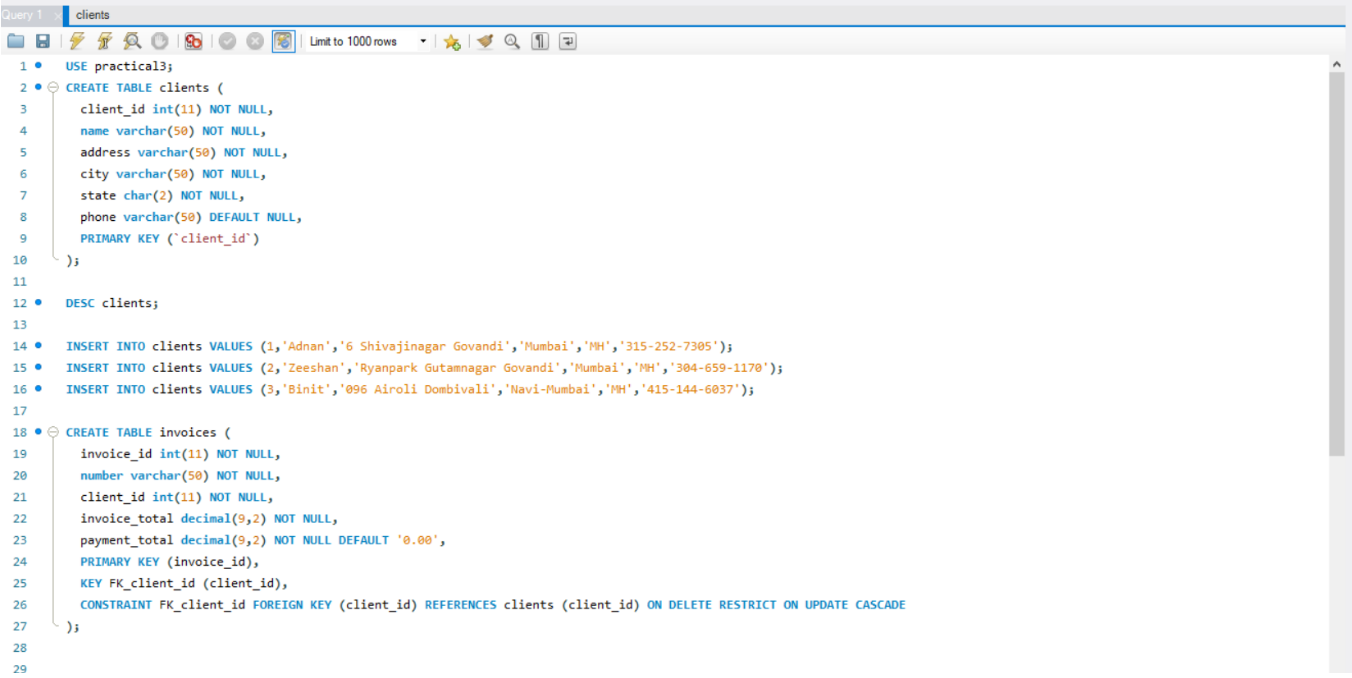
The unique key clause lists attributes that comprise a candidate key.

The foreign key clause lists the attributes that comprise the foreign key and the name of the relation referenced by the foreign key. By default, a foreign key references the primary key attributes of the referenced table.

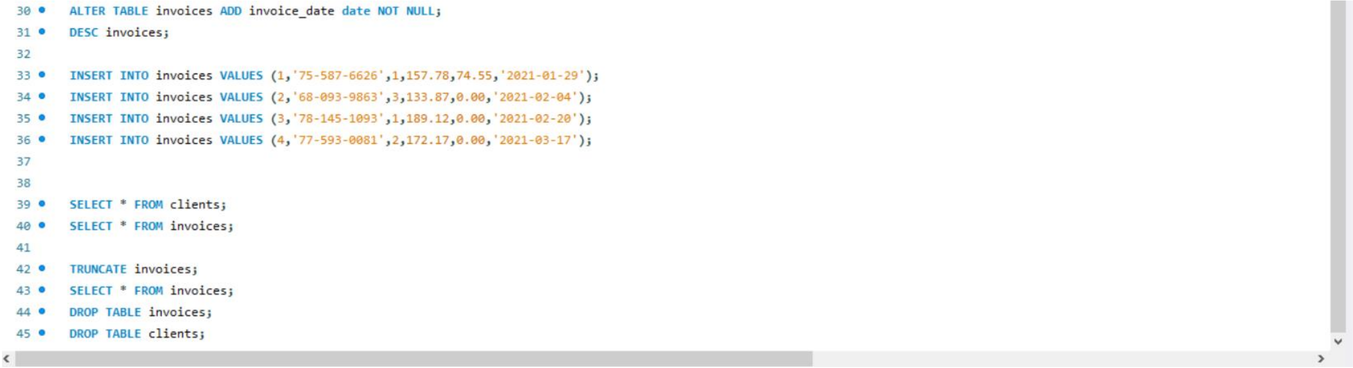
```
create table account
(account_number char(10),
branch_name      char(15),
balance     integer,
primary key (account_number),
foreign key (branch_name) references branch )
```

Conclusion: We have Successfully executed DDL command using SQL Live

Code:

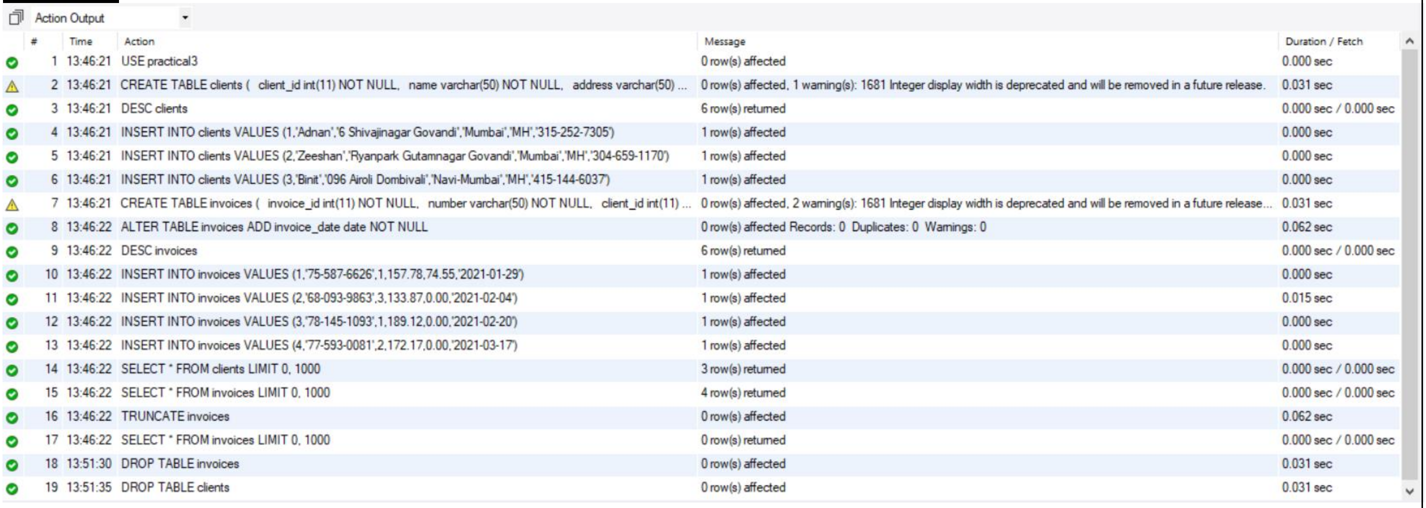


```sql
USE practical3;
CREATE TABLE clients (
  client_id int(11) NOT NULL,
  name varchar(50) NOT NULL,
  address varchar(50) NOT NULL,
  city varchar(50) NOT NULL,
  state char(2) NOT NULL,
  phone varchar(50) DEFAULT NULL,
  PRIMARY KEY (`client_id`)
);

DESC clients;

INSERT INTO clients VALUES (1,'Adnan','6 Shivajinagar Govandi','Mumbai','MH','315-252-7305');
INSERT INTO clients VALUES (2,'Zeeshan','Ryanpark Gutamnagar Govandi','Mumbai','MH','304-659-1170');
INSERT INTO clients VALUES (3,'Binit','096 Airoli Dombivali','Navi-Mumbai','MH','415-144-6037');

CREATE TABLE invoices (
  invoice_id int(11) NOT NULL,
  number varchar(50) NOT NULL,
  client_id int(11) NOT NULL,
  invoice_total decimal(9,2) NOT NULL,
  payment_total decimal(9,2) NOT NULL DEFAULT '0.00',
  PRIMARY KEY (invoice_id),
  KEY FK_client_id (client_id),
  CONSTRAINT FK_client_id FOREIGN KEY (client_id) REFERENCES clients (client_id) ON DELETE RESTRICT ON UPDATE CASCADE
);


```



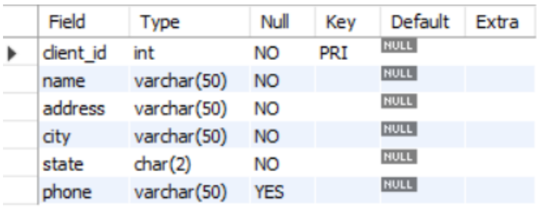
```sql
ALTER TABLE invoices ADD invoice_date date NOT NULL;
DESC invoices;

INSERT INTO invoices VALUES (1,'75-587-6626',1,157.78,74.55,'2021-01-29');
INSERT INTO invoices VALUES (2,'68-093-9863',3,133.87,0.00,'2021-02-04');
INSERT INTO invoices VALUES (3,'78-145-1093',1,189.12,0.00,'2021-02-20');
INSERT INTO invoices VALUES (4,'77-593-0081',2,172.17,0.00,'2021-03-17');


SELECT * FROM clients;
SELECT * FROM invoices;

TRUNCATE invoices;
SELECT * FROM invoices;
DROP TABLE invoices;
DROP TABLE clients;
```

Output:

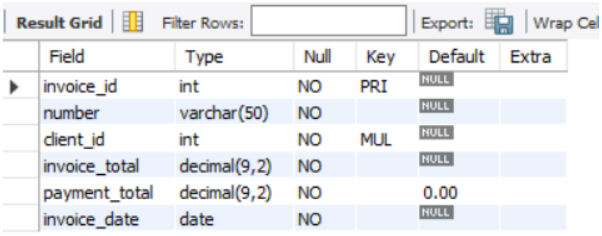


Action Output

	#	Time	Action	Message	Duration / Fetch
✓	1	13:46:21	USE practical3	0 row(s) affected	0.000 sec
⚠	2	13:46:21	CREATE TABLE clients (client_id int(11) NOT NULL, name varchar(50) NOT NULL, address varchar(50) ...	0 row(s) affected, 1 warning(s): 1681 Integer display width is deprecated and will be removed in a future release.	0.031 sec
✓	3	13:46:21	DESC clients	6 row(s) returned	0.000 sec / 0.000 sec
✓	4	13:46:21	INSERT INTO clients VALUES (1,'Adnan','6 Shivajinagar Govandi','Mumbai','MH','315-252-7305')	1 row(s) affected	0.000 sec
✓	5	13:46:21	INSERT INTO clients VALUES (2,'Zeeshan','Ryanpark Gutamnagar Govandi','Mumbai','MH','304-659-1170')	1 row(s) affected	0.000 sec
✓	6	13:46:21	INSERT INTO clients VALUES (3,'Binit','096 Airoli Dombivali','Navi-Mumbai','MH','415-144-6037')	1 row(s) affected	0.000 sec
⚠	7	13:46:21	CREATE TABLE invoices (invoice_id int(11) NOT NULL, number varchar(50) NOT NULL, client_id int(11) ...	0 row(s) affected, 2 warning(s): 1681 Integer display width is deprecated and will be removed in a future release...	0.031 sec
✓	8	13:46:22	ALTER TABLE invoices ADD invoice_date date NOT NULL	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.062 sec
✓	9	13:46:22	DESC invoices	6 row(s) returned	0.000 sec / 0.000 sec
✓	10	13:46:22	INSERT INTO invoices VALUES (1,'75-587-6626',1,157.78,74.55,'2021-01-29')	1 row(s) affected	0.000 sec
✓	11	13:46:22	INSERT INTO invoices VALUES (2,'68-093-9863',3,133.87,0.00,'2021-02-04')	1 row(s) affected	0.015 sec
✓	12	13:46:22	INSERT INTO invoices VALUES (3,'78-145-1093',1,189.12,0.00,'2021-02-20')	1 row(s) affected	0.000 sec
✓	13	13:46:22	INSERT INTO invoices VALUES (4,'77-593-0081',2,172.17,0.00,'2021-03-17')	1 row(s) affected	0.000 sec
✓	14	13:46:22	SELECT * FROM clients LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
✓	15	13:46:22	SELECT * FROM invoices LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
✓	16	13:46:22	TRUNCATE invoices	0 row(s) affected	0.062 sec
✓	17	13:46:22	SELECT * FROM invoices LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec
✓	18	13:51:30	DROP TABLE invoices	0 row(s) affected	0.031 sec
✓	19	13:51:35	DROP TABLE clients	0 row(s) affected	0.031 sec

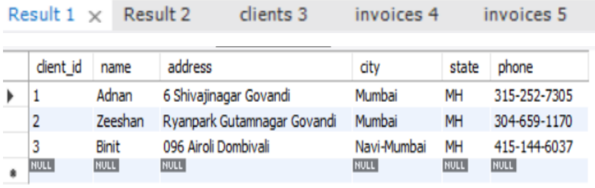


Field	Type	Null	Key	Default	Extra
client_id	int	NO	PRI	NULL	
name	varchar(50)	NO		NULL	
address	varchar(50)	NO		NULL	
city	varchar(50)	NO		NULL	
state	char(2)	NO		NULL	
phone	varchar(50)	YES		NULL	



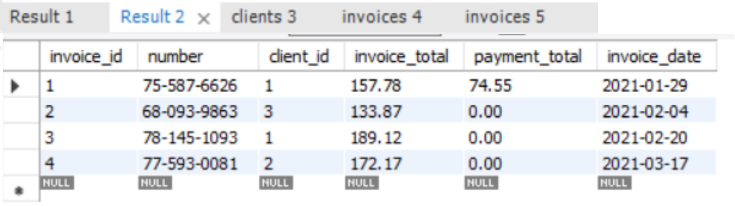
Result Grid | Filter Rows: | Export: | Wrap Cel

Field	Type	Null	Key	Default	Extra
invoice_id	int	NO	PRI	NULL	
number	varchar(50)	NO		NULL	
client_id	int	NO	MUL	NULL	
invoice_total	decimal(9,2)	NO		NULL	
payment_total	decimal(9,2)	NO		0.00	
invoice_date	date	NO		NULL	



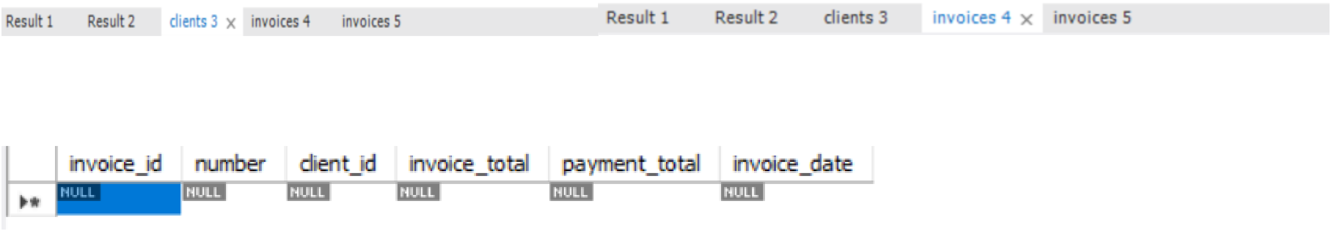
Result 1 × | Result 2 | clients 3 | invoices 4 | invoices 5

client_id	name	address	city	state	phone
1	Adnan	6 Shivajinagar Govandi	Mumbai	MH	315-252-7305
2	Zeeshan	Ryanpark Gutamnagar Govandi	Mumbai	MH	304-659-1170
3	Binit	096 Airoli Dombivali	Navi-Mumbai	MH	415-144-6037
NULL	NULL	NULL	NULL	NULL	NULL



Result 1 | Result 2 × | clients 3 | invoices 4 | invoices 5

invoice_id	number	client_id	invoice_total	payment_total	invoice_date
1	75-587-6626	1	157.78	74.55	2021-01-29
2	68-093-9863	3	133.87	0.00	2021-02-04
3	78-145-1093	1	189.12	0.00	2021-02-20
4	77-593-0081	2	172.17	0.00	2021-03-17
NULL	NULL	NULL	NULL	NULL	NULL



Result 1 | Result 2 | clients 3 × | invoices 4 | invoices 5

Result 1 | Result 2 | clients 3 | invoices 4 × | invoices 5

invoice_id	number	client_id	invoice_total	payment_total	invoice_date
NULL	NULL	NULL	NULL	NULL	NULL