# SɅRɅSWɅTI

## Department of Computer Engineering

**College of Engineering**

SɅRɅSWɅTI

### 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.

# SɅRɅSWɅTI

## Department of Computer Engineering

**College of Engineering**

SɅRɅSWɅTI

* **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, ..., *An Dn,*

(integrity-constraint1),

...,

(integrity-constraintk))

*r* is the name of the relation

each *Ai* is an attribute name in the schema of relation *r Di* is the data type of values in the domain of attribute *Ai*

###### 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*

# SɅRɅSWɅTI

## Department of Computer Engineering

**College of Engineering**

SɅRɅSWɅTI

**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, … *A*m

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

# SɅRɅSWɅTI

## Department of Computer Engineering

**College of Engineering**

SɅRɅSWɅTI

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

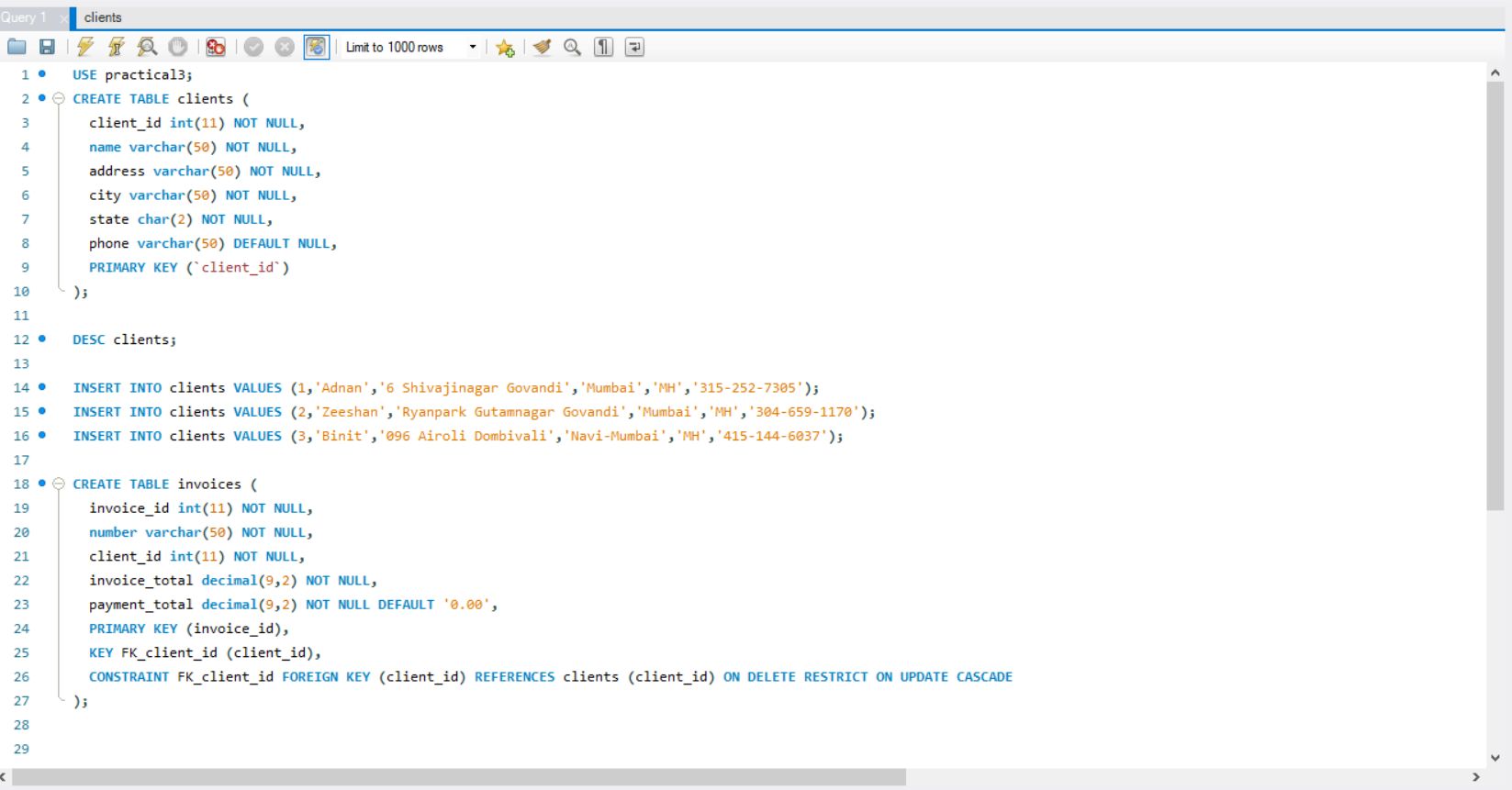
# SɅRɅSWɅTI

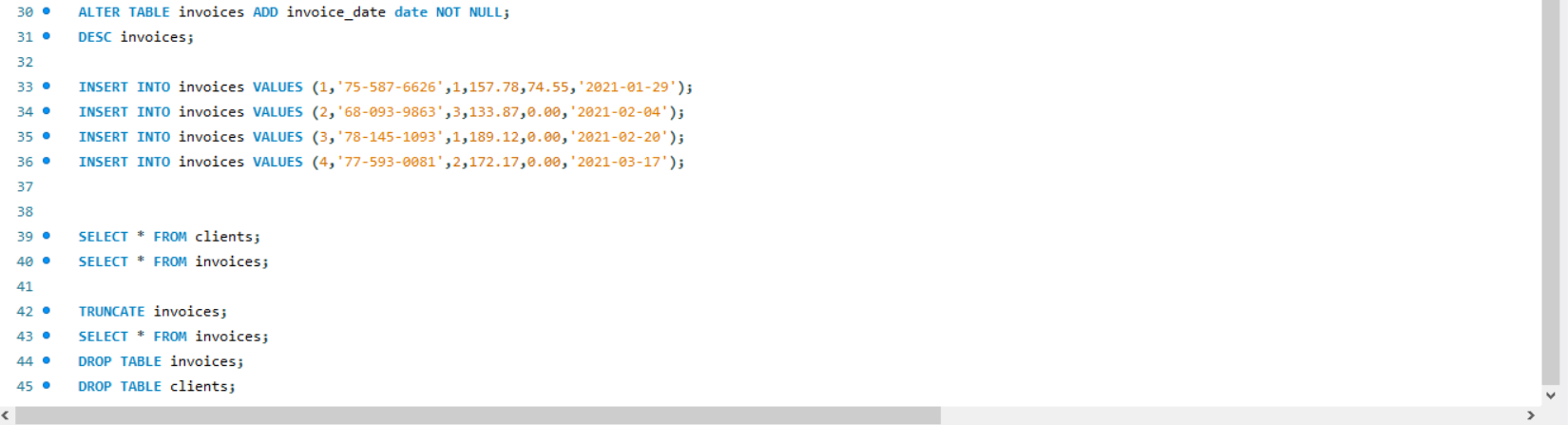
## Department of Computer Engineering

**College of Engineering**

SɅRɅSWɅTI

**Code**:





**Output**:

