

SQL

SQL Commands

- ▶ The standard SQL commands to interact with relational databases are CREATE, SELECT, INSERT, UPDATE, DELETE and DROP. These commands can be classified into groups based on their nature:
- 

DDL – Data Definition Language

Command	Description
CREATE	Creates a new table, a view of a table, or other object in database
ALTER	Modifies an existing database object, such as a table.
DROP	Deletes an entire table, a view of a table or other object in the database.

DML – Data Manipulation Language

Command	Description
SELECT	Retrieves certain records from one or more tables
INSERT	Creates a record
UPDATE	Modifies records
DELETE	Deletes records

DCL – Data Control Language

Command	Description
GRANT	Gives a privilege to user
REVOKE	Takes back privileges granted from user

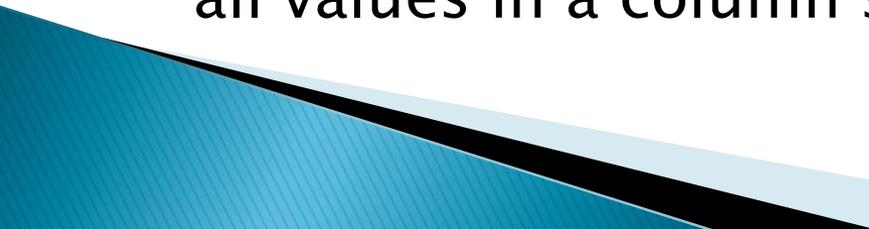
Table or relation

- ▶ The data in RDBMS is stored in database objects called **tables or relations**.
 - ▶ Every table is broken up into smaller entities called fields or attributes.
 - ▶ The fields in the CUSTOMERS table consist of ID, NAME, AGE, ADDRESS and SALARY.
- 

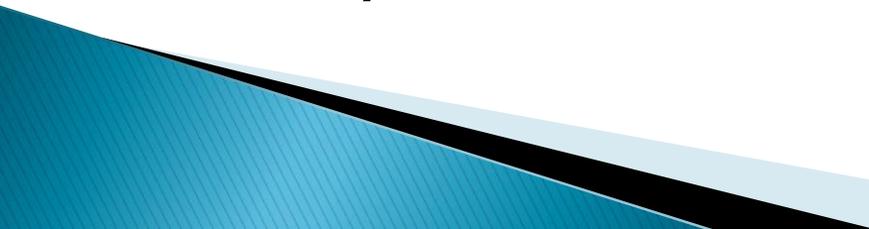
Record or Tuple

- ▶ A record, also called a **row or tuple of data**, is each individual entry that exists in a table.
- ▶ Each customer detail is a **tuple**

Constraints

- ▶ constraints available in SQL:
 - ▶ NOT NULL Constraint: Ensures that a column cannot have NULL value.
 - ▶ DEFAULT Constraint: Provides a default value for a column when none is specified.
 - ▶ UNIQUE Constraint: Ensures that all values in a column are different.
 - ▶ PRIMARY Key: Uniquely identified each rows/records in a database table.
 - ▶ FOREIGN Key: Uniquely identified a rows/records in any another database table.
 - ▶ CHECK Constraint: The CHECK constraint ensures that all values in a column satisfy certain conditions.
- 

Data Integrity

- ▶ **Entity Integrity:** There are no duplicate rows in a table.
 - ▶ **Domain Integrity:** Enforces valid entries for a given column by restricting the type, the format, or the range of values.
 - ▶ **Referential integrity:** Rows cannot be deleted, which are used by other records.
 - ▶ **User-Defined Integrity:** Enforces some specific business rules that do not fall into entity, domain or referential integrity
- 

Database Normalization

- ▶ **Database normalization** is the process of efficiently organizing data in a database. There are two reasons of the normalization process:
 - ▶ Eliminating redundant data, for example, storing the same data in more than one tables.
 - ▶ Ensuring data dependencies make sense.
- 

Character Datatypes

Data Type Syntax	Oracle 9i	Oracle 11g	Explanation
char(size)	Maximum size of 2000 bytes.	Maximum size of 2000 bytes.	Where size is the number of characters to store. Fixed-length strings. Space padded.
nchar(size)	Maximum size of 2000 bytes.	Maximum size of 2000 bytes.	Where size is the number of characters to store. Fixed-length NLS string Space padded.
nvarchar2(size)	Maximum size of 4000 bytes.	Maximum size of 4000 bytes.	Where size is the number of characters to store. Variable-length NLS string.

Character Datatypes

Data Type Syntax	Oracle 9i	Oracle 11g	Explanation
<code>varchar2(size)</code>	Maximum size of 4000 bytes. Maximum size of 32KB in PLSQL.	Maximum size of 4000 bytes. Maximum size of 32KB in PLSQL.	Where size is the number of characters to store. Variable-length string.
<code>long</code>	Maximum size of 2GB.	Maximum size of 2GB.	Variable-length strings. (backward compatible)
<code>raw</code>	Maximum size of 2000 bytes.	Maximum size of 2000 bytes.	Variable-length binary strings
<code>long raw</code>	Maximum size of 2GB.	Maximum size of 2GB.	Variable-length binary strings. (backward compatible)

Data Type Syntax	Oracle 11g	Explanation
<h1>Numeric Datatypes</h1>		
number(p,s)	Precision can range from 1 to 38. Scale can range from -84 to 127.	Where p is the precision and s is the scale. For example, number(7,2) is a number that has 5 digits before the decimal and 2 digits after the decimal.
numeric(p,s)	Precision can range from 1 to 38.	Where p is the precision and s is the scale. For example, numeric(7,2) is a number that has 5 digits before the decimal and 2 digits after the decimal.
float		
dec(p,s)	Precision can range from 1 to 38.	Where p is the precision and s is the scale. For example, dec(3,1) is a number that has 2 digits before the decimal and 1 digit after the decimal.

Numeric Datatypes

Data Type Syntax	Oracle 11g	Explanation
decimal(p,s)	Precision can range from 1 to 38.	Where p is the precision and s is the scale. For example, decimal(3,1) is a number that has 2 digits before the decimal and 1 digit after the decimal.
integer		
int		
smallint		
real		
double precision		

Date/Time Datatypes

Data Type Syntax	Oracle 11g	Explanation
date	A date between Jan 1, 4712 BC and Dec 31, 9999 AD.	
timestamp (fractional seconds precision)	fractional seconds precision must be a number between 0 and 9. (default is 6)	Includes year, month, day, hour, minute, and seconds. For example: timestamp(6)
timestamp (fractional seconds precision) with time zone	fractional seconds precision must be a number between 0 and 9. (default is 6)	Includes year, month, day, hour, minute, and seconds; with a time zone displacement value. For example: timestamp(5) with time zone

CREATE TABLE Statement

CREATE TABLE Customers

(custid number(10) **PRIMARY KEY NOT NULL**,
cname varchar2(50) NOT NULL,
city varchar2(50)
);

Create an table called *Suppliers* that stores supplier ID, name, and address information.

CREATE TABLE Statement

CREATE TABLE Customers

```
( custid number(10) PRIMARY KEY NOT NULL,  
  cname varchar2(50) NOT NULL,  
  city varchar2(50)  
);
```

Or

CREATE TABLE Customers

```
( custid number(10),  
  cname varchar2(50) NOT NULL,  
  city varchar2(50) ,  
  CONSTRAINT cust_pk PRIMARY KEY (customer_id) );  
);
```

```
CREATE TABLE Dept
( deptId number(10),
  dname varchar2(50) NOT NULL,
  city varchar2(50) ),
CONSTRAINT dept_pk PRIMARY
KEY (deptId );
);
```

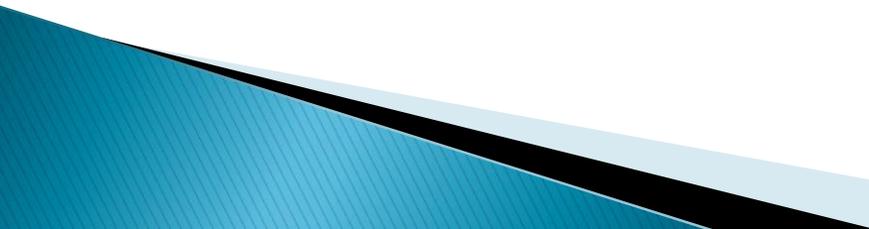


Table with Foreign key

```
CREATE TABLE employees (  
  eno number(10) NOT NULL,  
  ename varchar2(50) NOT NULL,  
  deptno number(10),  
  salary number(6),  
  CONSTRAINT emp_pk PRIMARY KEY (eno),  
  CONSTRAINT fk_dept FOREIGN KEY (deptno)  
  REFERENCES Dept(deptId) );
```

Insert Command

```
INSERT INTO suppliers (supplier_id,  
supplier_name) VALUES (5000, 'Apple');
```



select

- ▶ **Select all fields from one table**

```
SELECT * FROM homes WHERE rooms >= 2;
```

- ▶ **NOT NULL**

```
SELECT * FROM homes WHERE rooms is not  
null;
```

- ▶ **LIKE**

```
SELECT home_id, home_type, rooms FROM  
homes WHERE home_id < 500 AND home_type  
like 'two-storey'
```