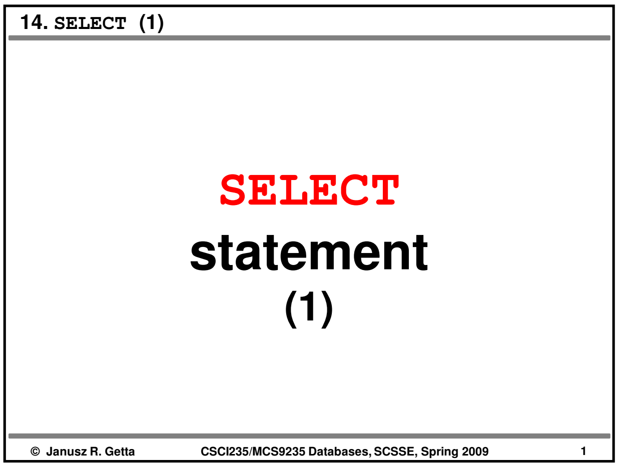
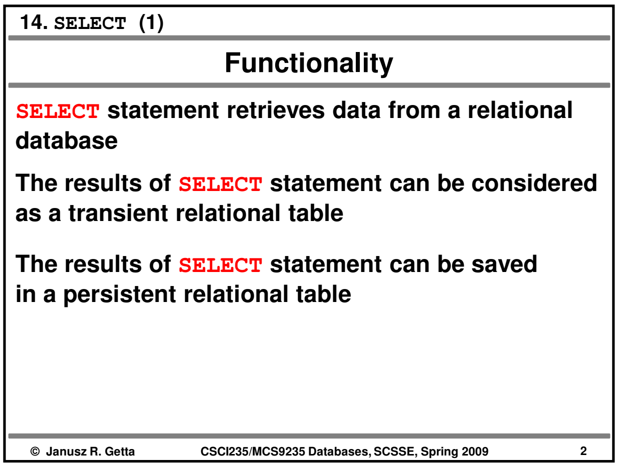


SELECT statement (1)

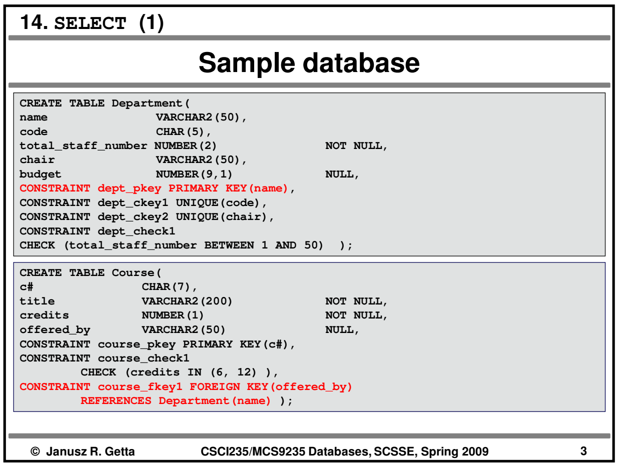


Functionality

SELECT statement retrieves data from a relational database

The results of SELECT statement can be considered as a transient relational table

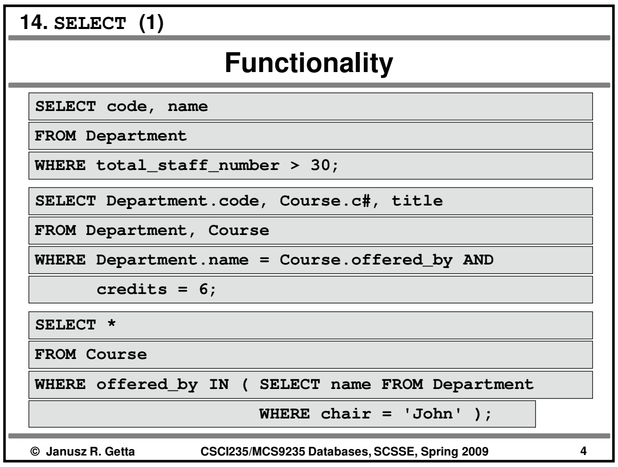
The results of SELECT statement can be saved in a persistent relational table



Sample database

```
CREATE TABLE Department(
name              VARCHAR2(50),
code              CHAR(5),
total_staff_number NUMBER(2)               NOT NULL,
chair             VARCHAR2(50),
budget            NUMBER(9,1)             NULL,
CONSTRAINT dept_pkey PRIMARY KEY(name),
CONSTRAINT dept_ckey1 UNIQUE(code),
CONSTRAINT dept_ckey2 UNIQUE(chair),
CONSTRAINT dept_check1
CHECK (total_staff_number BETWEEN 1 AND 50)  );
```

```
CREATE TABLE Course(
c#               CHAR(7),
title            VARCHAR2(200)            NOT NULL,
credits          NUMBER(1)                NOT NULL,
offered_by       VARCHAR2(50)             NULL,
CONSTRAINT course_pkey PRIMARY KEY(c#),
CONSTRAINT course_check1
        CHECK (credits IN (6, 12) ),
CONSTRAINT course_fkey1 FOREIGN KEY(offered_by)
        REFERENCES Department(name) );
```

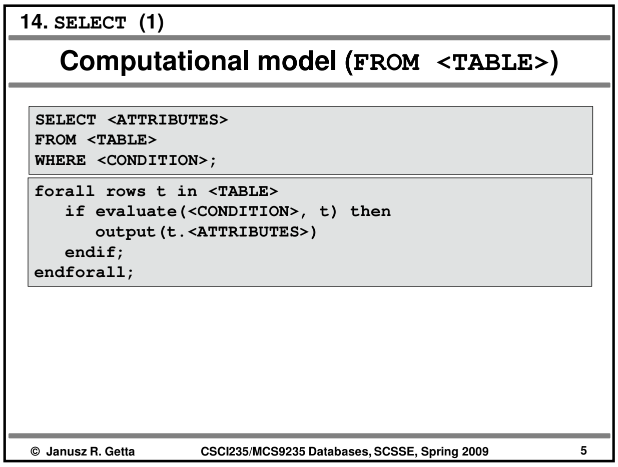


Functionality

```
SELECT code, name
FROM Department
WHERE total_staff_number > 30;
```

```
SELECT Department.code, Course.c#, title
FROM Department, Course
WHERE Department.name = Course.offered_by AND
        credits = 6;
```

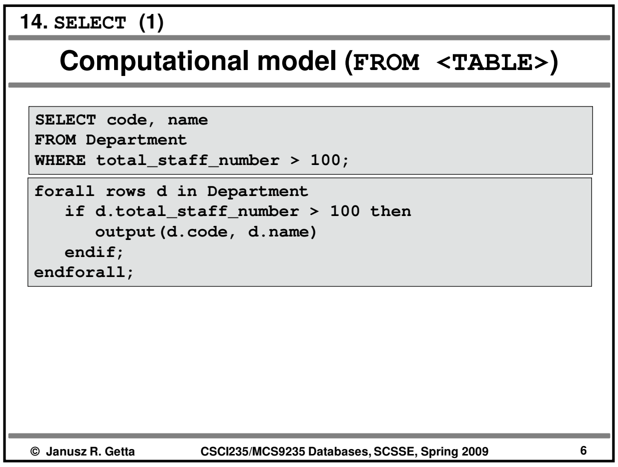
```
SELECT *
FROM Course
WHERE offered_by IN ( SELECT name FROM Department
                      WHERE chair = 'John' );
```



Computational model (FROM <TABLE>)

```
SELECT <ATTRIBUTES>
FROM <TABLE>
WHERE <CONDITION>;
```

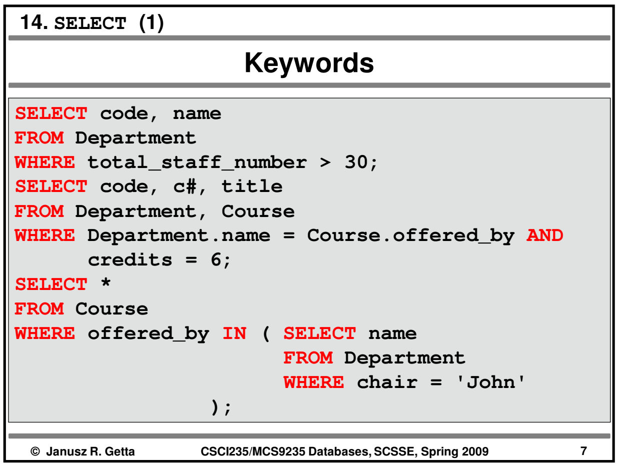
```
forall rows t in <TABLE>
   if evaluate(<CONDITION>, t) then
      output(t.<ATTRIBUTES>)
   endif;
endforall;
```



Computational model (FROM <TABLE>)

```
SELECT code, name
FROM Department
WHERE total_staff_number > 100;
```

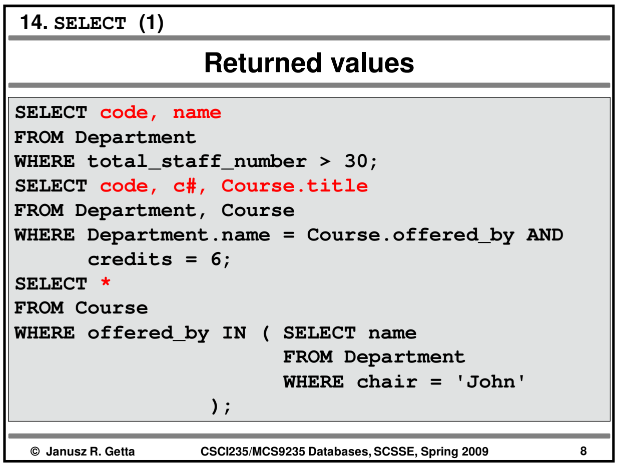
```
forall rows d in Department
   if d.total_staff_number > 100 then
      output(d.code, d.name)
   endif;
endforall;
```



14. SELECT (1)

Keywords

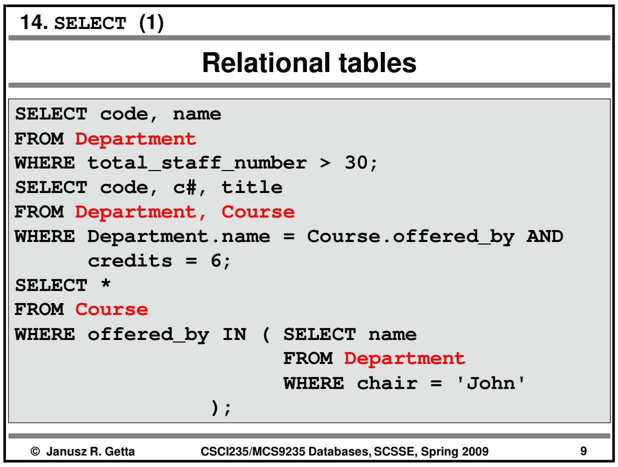
```
SELECT code, name
FROM Department
WHERE total_staff_number > 30;
SELECT code, c#, title
FROM Department, Course
WHERE Department.name = Course.offered_by AND
      credits = 6;
SELECT *
FROM Course
WHERE offered_by IN ( SELECT name
                      FROM Department
                      WHERE chair = 'John'
                    );
```



14. SELECT (1)

Returned values

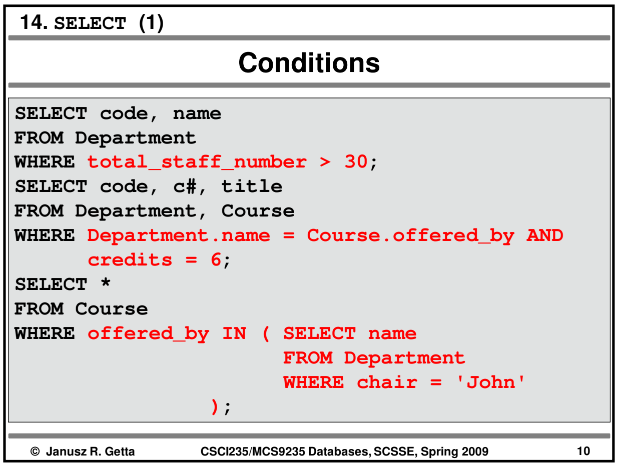
```
SELECT code, name
FROM Department
WHERE total_staff_number > 30;
SELECT code, c#, Course.title
FROM Department, Course
WHERE Department.name = Course.offered_by AND
      credits = 6;
SELECT *
FROM Course
WHERE offered_by IN ( SELECT name
                      FROM Department
                      WHERE chair = 'John'
                    );
```



14. SELECT (1)

Relational tables

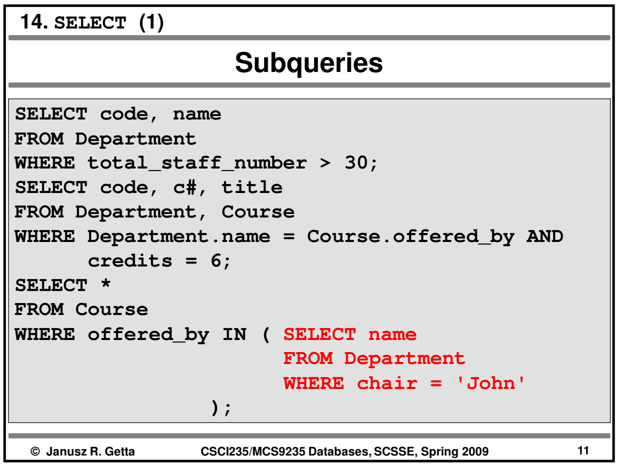
```
SELECT code, name
FROM Department
WHERE total_staff_number > 30;
SELECT code, c#, title
FROM Department, Course
WHERE Department.name = Course.offered_by AND
      credits = 6;
SELECT *
FROM Course
WHERE offered_by IN ( SELECT name
                      FROM Department
                      WHERE chair = 'John'
                    );
```



14. SELECT (1)

Conditions

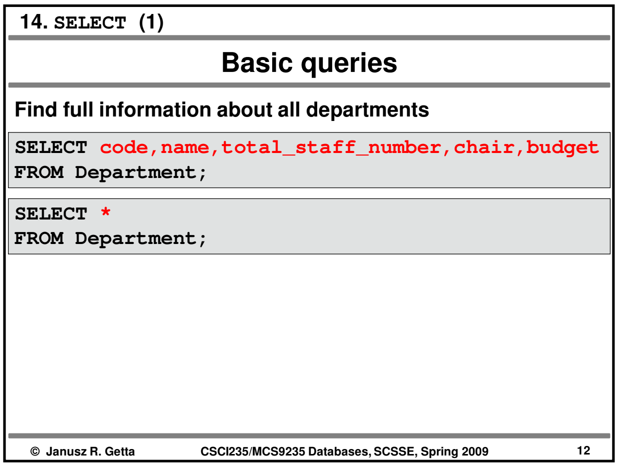
```
SELECT code, name
FROM Department
WHERE total_staff_number > 30;
SELECT code, c#, title
FROM Department, Course
WHERE Department.name = Course.offered_by AND
      credits = 6;
SELECT *
FROM Course
WHERE offered_by IN ( SELECT name
                      FROM Department
                      WHERE chair = 'John'
                    );
```



14. SELECT (1)

Subqueries

```
SELECT code, name
FROM Department
WHERE total_staff_number > 30;
SELECT code, c#, title
FROM Department, Course
WHERE Department.name = Course.offered_by AND
      credits = 6;
SELECT *
FROM Course
WHERE offered_by IN ( SELECT name
                      FROM Department
                      WHERE chair = 'John'
                    );
```



14. SELECT (1)

Basic queries

Find full information about all departments

```
SELECT code,name,total_staff_number,chair,budget
FROM Department;
```

```
SELECT *
FROM Department;
```

14. SELECT (1)

Projection queries

Find the names and chairpersons of all departments

```
SELECT name,chair
FROM Department;
```

Find the titles of all courses

```
SELECT title
FROM Course;
```

14. SELECT (1)

Projection queries with duplicates

Find the credit points of all courses

```
SELECT credits
FROM Course;
```

Find the total number of staff members in each department

```
SELECT total_staff_number
FROM Department;
```

14. SELECT (1)

Projection queries with no duplicates

Find the distinct credit points of all courses

```
SELECT DISTINCT credits
FROM Course;
```

Find the distinct total number of staff members in each department

```
SELECT DISTINCT total_staff_number
FROM Department;
```

14. SELECT (1)

Queries with row functions

List the names of departments in uppercase format

```
SELECT UPPER(name)
FROM Department;
```

Find the first three characters from all course codes and full titles of all courses

```
SELECT SUBSTR(c#, 1, 3), title
FROM Course;
```

14. SELECT (1)

Queries with group functions

Find the total number of courses

```
SELECT COUNT(*)
FROM Course;
```

Find the total number of all staff members in all departments

```
SELECT SUM(total_staff_number)
FROM Department;
```

Find an average number of credit points per course

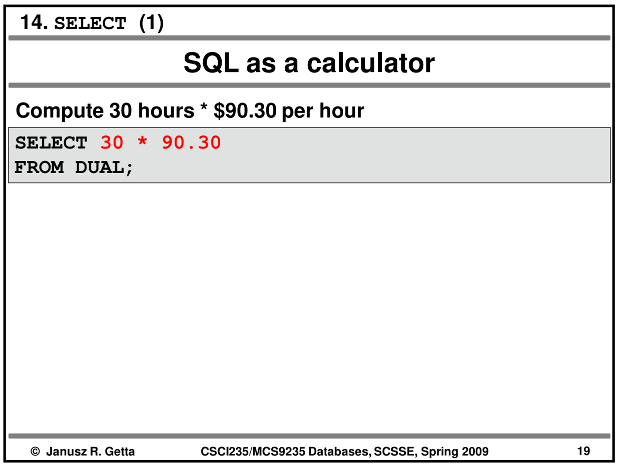
```
SELECT AVG(credits)
FROM Course;
```

14. SELECT (1)

Queries with group functions

Find the total number of staff members in the largest department

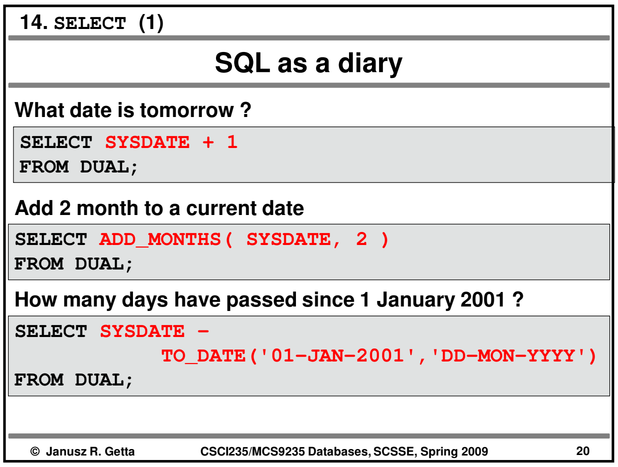
```
SELECT MAX(total_staff_number)
FROM Department;
```



SQL as a calculator

Compute 30 hours * $90.30 per hour

```
SELECT 30 * 90.30
FROM DUAL;
```



SQL as a diary

What date is tomorrow ?

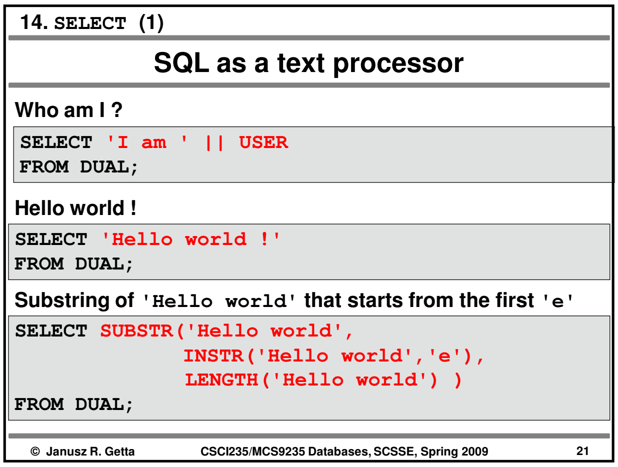
```
SELECT SYSDATE + 1
FROM DUAL;
```

Add 2 month to a current date

```
SELECT ADD_MONTHS( SYSDATE, 2 )
FROM DUAL;
```

How many days have passed since 1 January 2001 ?

```
SELECT SYSDATE -
            TO_DATE('01-JAN-2001','DD-MON-YYYY')
FROM DUAL;
```



SQL as a text processor

Who am I ?

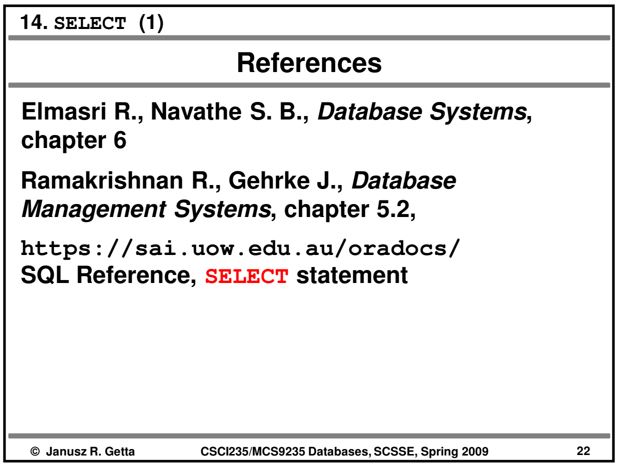
```
SELECT 'I am ' || USER
FROM DUAL;
```

Hello world !

```
SELECT 'Hello world !'
FROM DUAL;
```

Substring of `'Hello world'` that starts from the first `'e'`

```
SELECT SUBSTR('Hello world',
              INSTR('Hello world','e'),
              LENGTH('Hello world') )
FROM DUAL;
```



References

Elmasri R., Navathe S. B., *Database Systems*, chapter 6

Ramakrishnan R., Gehrke J., *Database Management Systems*, chapter 5.2,

`https://sai.uow.edu.au/oradocs/`
SQL Reference, SELECT statement