

Łączenie tabel



```
select * from emp, dept;
```

UWAGA: brak klauzuli WHERE powoduje zwrócenie wierszy z iloczynu kartezyjańskiego tabel EMP i DEPT.

```
select * from emp, dept  
      where emp.deptno = dept.deptno;
```

```
select ename,deptno,dname from emp, dept  
      where emp.deptno = dept.deptno;
```

```
select ename,emp.deptno,dname from emp, dept  
      where emp.deptno = dept.deptno;
```

Łączenie - aliasy tabel



```
select e.*, d.dname from emp e, dept d  
where e.deptno = d.deptno;
```

```
select e.ename from emp e, dept d  
where e.deptno = d.deptno  
and d.loc = 'NEWYORK'  
and e.sal < 2000;
```

Łączenie zewnętrzne



```
select e.ename, d.dname from emp e, dept d  
      where e.deptno(+) = d.deptno and  
             d.deptno in (30,40);
```

Podwójne użycie tabeli



```
select e.ename pracownik, e.sal zar_prac,  
       m.ename kierownik, m.sal zar_kier  
from emp e, emp m  
       where e.mgr = m.empno  
              and e.sal > m.sal;
```

Operacje mnogościowe na tabelach



UNION, INTERSECT, MINUS

select job from emp where deptno = 10

UNION

select job from emp where deptno = 30;

select job from emp where deptno = 10

INTERSECT

select job from emp where deptno = 30;

select job from emp where deptno = 10

MINUS

select job from emp where deptno = 30;

Operacje mnogościowe na tabelach - powtórzenia



select job from emp where deptno = 10

UNION ALL

select job from emp where deptno = 30;

Operacje mnogościowe na tabelach - porządek



```
select empno, ename, sal  
from emp  
UNION  
select id, name, salary  
from emp_history  
ORDER BY 2;
```

Podzapytania



```
select min(sal) from emp; { =:x }
```

```
select ename, job, sal from emp  
where sal = x;
```

```
select ename, job, sal from emp  
where sal = (select min(sal) from emp);
```


Podzapytania - wiele wierszy



```
select ename, job, sal from emp
where sal = (
    select min(sal) from emp group by deptno
);
```

```
select ename, job, sal from emp
where sal IN (
    select min(sal) from emp group by deptno
);
```

Podzapytania - wiele wierszy



```
select ename, job, sal from emp
where (deptno,sal) IN
(
  select deptno,min(sal) from emp group by deptno
);
```

Podzapytania - wiele wierszy



```
select ename, job, sal from emp
where sal >
(
  select sal from emp where deptno = 20
);
```

Podzapytania - wiele wierszy



```
select ename, job, sal from emp
where sal > ANY
(
  select sal from emp where deptno = 20
);
```

=ANY <=> IN

Podzapytania - wiele wierszy



```
select ename, job, sal from emp
where sal > ALL
    (
    select sal from emp where deptno = 20
    );
```

Podzapytania skorelowane



```
select ename, job, deptno from emp e
where sal >
(
  select avg(sal) from emp where deptno = e.deptno
);
```

DDL - Definowanie struktur



CREATE, ALTER, DROP

drop table tabelka;

create table tabelka
(pole1 varchar2(100),
 pole2 date,
 pole3 number(5,2)
);

create table tabelka **as** select * from emp;

create table tabelka **as**
 select ename,sal
 from emp
 where deptno = 20;

DDL -

Definowanie struktur



alter table tabelka **add** (pole4 char(15));

alter table tabelka **modify** (pole3 number(6,3));

alter table tabelka **drop column** pole4;

Więzy kontrolne



```
create table osoby
( imie CHAR(15),
  nazwisko CHAR(25),
  wiek number(3),
  plec CHAR(1)
  constraint PLEC_CHK CHECK (plec in ('M','K'))
);

insert into osoby values ('Jan','Kowalski',23,'M');
insert into osoby values ('Ewa','Niska',27,'P');
insert into osoby values ('Ewa','Niska',27,null);
```

Więzy kontrolne



```
create table osoby  
( imie CHAR(15),  
  nazwisko CHAR(25),  
  wiek number(3),  
  plec CHAR(1) NOT NULL  
  constraint PLEC_CHK CHECK (plec in ('M','K'))  
);
```

```
insert into osoby values ('Ewa','Niska',27,null);
```

Więzy kontrolne



drop table osoby;

```
create table osoby  
( imie CHAR(15),  
  nazwisko CHAR(25),  
  wiek number(3),  
  plec CHAR(1) NOT NULL  
  constraint PLEC_WIEK_CHK CHECK  
    (plec in ('M','K') AND wiek < 110)  
);
```

insert into osoby values ('Ewa','Niska',127,'M');

Więzy kontrolne



```
drop table osoby;
```

```
create table osoby  
( imie CHAR(15),  
  nazwisko CHAR(25),  
  wiek number(3),  
  plec CHAR(1)  
  constraint WIEK1_CHK CHECK (wiek > 10),  
  constraint WIEK2_CHK CHECK (wiek < 110)  
);
```

```
insert into osoby values ('Ewa','Niska',9,'M');
```

```
insert into osoby values ('Ewa','Niska',127,'M');
```

Więzy kontrolne



```
create table osoby  
( imie CHAR(15),  
  nazwisko CHAR(25),  
  wiek number(3),  
  plec CHAR(1));
```

```
insert into osoby values ('Jan','Kowalski',9,'X');
```

```
alter table osoby  
  add constraint plec_chk check (plec in ('K','M'));
```

```
alter table osoby disable constraint plec_chk ;
```

```
alter table osoby enable constraint plec_chk ;
```

Więzy unikatowe



```
create table samochody  
( marka char(15),  
  model char(15),  
  nr_rej char(10) unique );
```

```
create table samochody  
( marka char(15),  
  model char(15),  
  nr_rej char(10),  
  constraint nrrej_uk unique (nr_rej) );
```

Więzy unikatowe



```
create table samochody  
( marka char(15),  
  model char(15),  
  nr_rej char(10)) ;
```

```
alter table samochody  
  add constraint nrrej_uk unique (nr_rej) ;
```

```
alter table samochody drop constraint nrrej_uk ;
```

Klucz główny



```
create table samochody  
( marka char(15),  
  model char(15),  
  nr_rej char(10)) ;
```

```
alter table samochody  
  add constraint nrrej_pk PRIMARY KEY (nr_rej) ;
```

```
alter table samochody drop primary key;
```

```
alter table samochody disable primary key;
```


Więzy integralności

```
create table tab_nadrzedna (id number, pole1 char(10));
```

```
create table tab_podrzedna  
(id number, id_nadrz number, pole1 char (10));
```

```
alter table tab_podrzedna  
add constraint integr_fk FOREIGN KEY  
(id_nadrz) REFERENCES tab_nadrzedna (id) ;
```

```
alter table tab_nadrzedna  
add constraint uk1 UNIQUE (id);
```

```
alter table emp  
add constraint FK_DEPTNO foreign key  
(deptno) references dept (deptno);
```

Więzy integralności



insert into tab_podrzedna values (1,10,'xyz') ;

insert into tab_nadrzedna values (10,'abc') ;

delete tab_nadrzedna where id=10;

**alter table tab_podrzedna
add constraint integr_fk FOREIGN KEY
(id_nadrz) REFERENCES tab_nadrzedna (id)
on delete cascade ;**

(...) on delete set null ;

DML - manipulowanie danymi



INSERT, UPDATE, DELETE

**insert into emp (empno, ename, sal)
values (7777,'Kowalski',2500);**

**insert into emp
(7777,'Kowalski','CLERK',7566,trunc(sysdate),2500,
null,30);**

insert into emp_kopia (select * from emp where ...);

DML - manipulowanie danymi



update emp set sal = 1.1 * sal;

update emp set sal = 1.1 * sal where deptno = 20;

**update emp
set hiredate = to_date('17.05.2008','DD.MM.YYYY')
where empno = 7777;**

**update emp
set hiredate = to_date('17.05.2008','DD.MM.YYYY') ,
sal = 2550
where empno = 7777;**

DML - manipulowanie danymi



delete from emp_kopia;

delete emp_kopia where empno = 7777;

**delete emp_kopia where sal =
(select min(sal) from emp);**

truncate table emp_kopia [drop/reuse storage];

DML - transakcje



COMMIT - zatwierdź

ROLLBACK - wycofaj

```
...  
COMMIT;  
    ...ins/upd/del ...  
SAVEPOINT S1;  
    ...ins/upd/del ...  
SAVEPOINT S2;  
    ...ins/upd/del ...  
ROLLBACK TO S2;  
    ...ins/upd/del ...  
COMMIT;
```

Zakleszczenie - DEADLOCK



create table tabelka (pole number primary key);

sesja 1:

insert into tabelka values (1);

sesja 2:

insert into tabelka values (2);

sesja 2:

insert into tabelka values (1);

sesja 1:

insert into tabelka values (2);

ORA-00054: zasób zajęty a wyspecyfikowano pozyskanie z NOWAIT

Perspektywy



CREATE, DROP VIEW

drop view emp_30;

create view emp_30 as
 select ename, sal from emp
 where deptno = 30;

create or replace view emp_30 as
 select empno, ename, sal from emp
 where deptno = 30;

Perspektywy złożone



```
create view emp_d_30 as
  select e.empno numer, e.ename nazwisko,
         d.dname wydzial
  from emp e, dept d
  where e.deptno = d.deptno
         and e.deptno = 30;
```

```
create view emp_d_s as
  select deptno, sum(sal) suma
  from emp
  group by deptno;
```

Operacje DML na perspektywach



```
create or replace view emp_30 as  
  select empno numer, ename nazwisko, deptno nr_wydz  
  from emp  
  where deptno = 30;
```

```
select * from emp_30;
```

```
select numer, nazwisko from emp_30  
where nr_wydz = 20;
```

```
insert into emp_30  
(numer, nazwisko, nr_wydz) values (7777,'Nowak',30);
```

Operacje DML na perspektywach



```
insert into emp_30  
(numer, nazwisko, nr_wydz) values (7777,'Nowak',20);
```

```
create or replace view emp_30 as  
  select empno numer, ename nazwisko, deptno nr_wydz  
  from emp  
  where deptno = 30  
WITH CHECK OPTION ;
```

Operacje DML na perspektywach



```
create view emp_d_30 as
  select e.empno numer, e.ename nazwisko
         d.dname wydzial
  from emp e, dept d
  where e.deptno = d.deptno
         and e.deptno = 30
```

```
insert into emp_d_30
(numer, nazwisko, wydzial) values
(7777, 'Nowak', 'produkcja');
```

Operacje DML na perspektywach



```
create view emp_d_s as  
  select e.deptno, sum(sal) suma  
  from emp  
  group by deptno;
```

```
insert into emp_d_30  
(deptno, suma) values (50, 4500);
```

Operacje DML na perspektywach



```
create view emp_30 as
  select ename, sal, deptno
  from emp
  where deptno = 30;
```

???????

```
insert into emp_d_30
(ename, sal, deptno ) values ('Kowalski',3200,30);
```

Skrypt SQL

```
Bez tytułu - Notatnik
Plik Edycja Format Widok Pomoc

CREATE TABLE Tabela
(pole1 varchar2(100),
pole2 number
);

create view widok as
select * from tabela where pole2 = 30;

insert into tabela ('abc',10);
insert into tabela ('xyz',20);
insert into tabela ('ijk',30);

commit;
```



d:\skrypt.txt



```
Oracle SQL*Plus
Plik Edycja Szukaj Opcje Pomoc

SQL*Plus: Release 10.2.0.3.0 - Production on N Paź 18 20:38:38 2009

Copyright (c) 1982, 2006, Oracle. All Rights Reserved.

Połączono z:
Oracle Database 10g Enterprise Edition Release 10.2.0.3.0 - Production
With the Partitioning, OLAP and Data Mining options

SQL> @d:\skrypt.txt
```

Użytkownicy



create user *nazwa* identified by *haslo*;

grant connect, resource to ... ;

grant DBA to ...

drop user ... ;

alter user ... identified by *nowe_haslo*;

Uprawnienia



grant select on *table* to *uzytkownik*;

**grant update, insert, delete on *table*
to *uzytkownik*;**

grant all on *table* to *uzytkownik*;

revoke select on *table* from *uzytkownik*;

Synonimy



```
create synonym nazwa_synonimu  
for nazwa_tabeli;
```

```
create public synonym nazwa_synonimu  
for nazwa_tabeli;
```