

约束

与表一起使用，约束表中的数据

1.主键约束

表中能够体现一行记录的唯一标识，这样列就是主键特性

- 一张表中只能有一个主键约束
- 主键约束可能不止作用在一列上（联合主键）
- 主键约束列中的值不能重复
- 主键约束中不能出现null值（非空）

创建表时，直接创建主键

1.直接在列上指定主键

```
create table haha(  
    hahaid number primary key,--系统会自动创建主键名字  
  
    hahaname VARCHAR2(16),  
    hahadate date  
  
);  
...  
2.建表语句之后追加主键
```

sql

```
create table haha(  
hahaid number ,  
hahaname VARCHAR2(16),  
hahadate date,  
constraint pk_hahaid primary key(hahaid)--pk_hahaid 为主键名  
);
```

联合主键，只能在

- 建表语句之后追加主键
- 创建表之后追加主键

sql

```
create table haha(  
hahaid number ,  
hahaname VARCHAR2(16),  
hahadate date,  
constraint pk_hahaid primary key(hahaid,hahaname)--联合主键  
);  
...
```

创建表之后追加主键---推荐使用

```

alter table haha
add constraint pk_hahaid primary key(hahaid);

alter table haha
add constraint pk_hahaid primary key(hahaid,hahaname);--联合主键

create table stu_cour(
stuid number,
courid number,
score number
constraint pk_stucour primary key (stuid,courid)
);

create table stu_cour(
stuid number,
courid number,
score number
);

alter table stu_cour
add constraint pk_stucour primary key (stuid,`courid`);

```

主键约束创建规范---重要

主键列，应该没有实际任何意义
一旦列作为主键，也不应该有任何其它的含义

2.外键约束

--foreign key
--一个表的某个列，关联到另一个表的主键，或者唯一键，这样的键就是外键

创建外键,建表语句之后，表级定义

```

create table stu(
stuid number,
courid number,
deptno number,
constraint fk_stu_deptno foreign key(deptno) references dept(deptno)
);

```

创建外键,列级定义

```

create table stu(
stuid number,
courid number,
deptno number references dept(deptno)
);

```

创建外键,建表之后，DDL语句

外键 列 可以填null值

```
create table stu(
  stuid number,
  courid number,
  deptno number);

alter table stu add constraint fk_stu_deptno foreign key(deptno) references dept(deptno);
```

3.唯一约束

--表中的某个列，数据不能重复，需要加入唯一约束
--unique

可以录入多个null值

主键约束与唯一约束的区别

- 一张表可以有多个主键约束，但是可以有多个唯一约束
- 主键不为空，唯一可为空
- 主键创建的索引是聚集索引，唯一约束创建的非聚集约束

```
create table stu(
  stuid number,
  courid number,
  deptno number,
  constraint un_hahaname unique(deptno);
  constraint un_hahaname1 unique(courid);
)

;

alter table haha add constraint un_hahaname unique (hahaname);
```

4.检查性约束

--check(条件)

```
alter table haha
add (salary number);

alter table haha
add constraint ck_haha_salary check (salary between 0 and 1000);

create table stu(
  stuid number,
  courid number,
  deptno number,
  salary number,
  constraint pk_stu_id primary key(stuid);
  constraint un_stu_cour unique(courid);
  constraint ck_stu_sal check(salary between 0 and 1000)
);
```

5.非空约束

只能在列上定义，不能使用constraint关键字声明

```

create table stu(
    stuid number,
    courid number not null,
    deptno number not null,
    salary number,
    constraint pk_stu_id primary key(stuid);
    constraint un_stu_cour unique(courid);
    constraint ck_stu_sal check(salary between 0 and 1000)
);

```

6.启动与禁用

一个列上可以多个约束
系统默认的约束都是启用状态

禁用约束：

```
alter table stu disable constraint ck_stu_sal;
```

###启动约束：

```
alter table stu enable constraint ck_stu_sal;
```

NOTE: 表中的数据一定不能违反约束条件，否则不能启动

练习：创建一个项目表projects

```
projid number pk prono VARCHAR2 un pname VARCHAR2 not null manager number fk emp empno endtime date ck < 2020-1-1
```

```

create table projects1(
    projid number primary key,
    prono varchar2(16) unique,
    pname varchar2(10) not null,
    manager number references emp(empno),
    endtime date check(endtime<to_date('1-1-2020','dd-mm-yyyy'))
);

alter table projects add constraint ck_projects_endtime
check(endtime<to_date('1-1-2020','dd-mm-yyyy'));

alter table projects add constraint pk_projects_proid primary key (projid);

create table projects (
    projid number ,
    prono varchar2(10),
    pname varchar2(18) not null,--not null 只能在列级定义!!!
    manager number,
    endtime DATE,
    constraint un_pro_prono unique (prono)
) ;

alter table projects add constraint fk_projects_manager foreign key
(manager) references emp(empno);

```