**列表分区 散列分区 范围分区**

**范围分区语法**

**CREATE TABLE table(…)**

**PARTITION BY RANGE (column1[,column2,…])**

**( PARTITION partition1 VALUES LESS THAN**

**(literal|MAXVALUE) [TABLESPACE tablespace]**

**[,PARTITION partition2 VALUES LESS THAN**

**(literal|MAXVALUE) [TABLESPACE tablespace],…]**

**)…**

**其中：PARTITION BY RANGE：指明采用范围分区方法。**

**column：分区列，可以是单列分区，也可以是多列分区。**

**PARTITION partition1 ：设置分区名称。**

**VALUES LESS THAN：设置分区列值的上界。**

**TABLESPACE：设置分区对应的表空间。**

**Eg. 创建一个分区表，将学生信息根据其出生日期不同进行分区，将1980年1月1日前出生的学生信息保存在TBS1表空间中，1980年1月1日到1990年1月1日出生的学生信息保存在TBS2表空间中，其他学生信息保存在TBS3表空间中。**

**CREATE TABLE student\_range(**

**sno NUMBER(6) PRIMARY KEY,**

**sname VARCHAR2(10),**

**sage int,**

**birthday DATE**

**)**

**PARTITION BY RANGE(birthday)**

**( PARTITION p1 VALUES LESS THAN**

**(TO\_DATE('1980-1-1', 'YYYY-MM-DD'))**

**TABLESPACE TBS1,**

**PARTITION p2 VALUES LESS THAN**

**(TO\_DATE('1990-1-1', 'YYYY-MM-DD'))**

**TABLESPACE TBS2,**

**PARTITION p3 VALUES LESS THAN(MAXVALUE)**

**TABLESPACE TBS3**

**);**

* **列表分区语法**

**CREATE TABLE table(…)**

**PARTITION BY LIST(column)**

**( PARTITION partition1**

**VALUES([literal|NULL]|[DEFAULT])**

**[TABLESPACE tablespace]**

**[,PARTITION partition2**

**VALUES([literal|NULL]|[DEFAULT])**

**[TABLESPACE tablespace],…]**

**)…**

**Eg.创建一个分区表，将学生信息按性别不同进行分区，男学生信息保存在表空间TBS1中，而女学生信息保存在TBS2中 。**

**CREATE TABLE student\_list(**

**sno NUMBER(6) PRIMARY KEY,**

**sname VARCHAR2(10),**

**sex CHAR(2) CHECK(sex in ('M', 'F'))**

**)**

**PARTITION BY LIST(sex)**

**(**

**PARTITION stu\_male VALUES('M') TABLESPACE TBS1,**

**PARTITION stu\_female VALUES('F') TABLESPACE TBS2**

**);**

* **散列分区**

**CREATE TABLE table(…)**

**PARTITION BY HASH (column1[,column2,…])**

**[(PARTITION partition**

**[TABLESPACE tablespace][,…])]|**

**[PARTITIONS hash\_partition\_quantity**

**STORE IN (tablespace1[,…])]…**

**Eg.创建一个分区表，根据学号将学生信息均匀分布到TBS1**

**和TBS2两个表空间中 。**

**CREATE TABLE student\_hash (**

**sno NUMBER(6) PRIMARY KEY,**

**sname VARCHAR2(10)**

**)**

**PARTITION BY HASH(sno)**

**( PARTITION p1 TABLESPACE TBS1,**

**PARTITION p2 TABLESPACE TBS2**

**);**