|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 实验报告七 | | | | | | |
| 实验项目名称 | 数据库完整性语言实验 | | 姓名 | KAFLE SAMRAT | 日期 | 2020.10.13 |
| 教师评语 |  | | | | | |
| 实验成绩： | | 指导教师（签字）： 年 月 日 | | | | |
| 实验目的  掌握实体完整性、参照完整性和用户自定义完整性的定义和维护方法。  实验要求  对TPC-H数据库：  定义实体完整性，删除实体完整性。能够写出两种方式定义实体完整性的SQL语句：创建表时定义实体完整性、创建表后定义实体完整性。设计SQL语句验证完整性约束是否起作用。  1. Create a representation to define column level entity integrity   1. Define the entity integrity of the supplier table   use TPC\_H;  go  CREATE TABLE supp(  suppkey INTEGER CONSTRAINT supp\_1 PRIMARY KEY,  sup\_name CHAR(25),  sup\_address VARCHAR(80),  sup\_nation DECIMAL(11) REFERENCES SALES.NATION(N\_NATIONKEY),  sup\_phone CHAR(30),  sup\_acctbal REAl,  sup\_comment VARCHAR(101)  );     1. Defines the entity integrity of the table REGION   use TPC\_H;  go  create table region1(  regionkey1 int constraint region\_1 primary key,  region\_name char(25),  comment varchar(100)  );    2. Table level entity integrity is defined when a table is created  A. Define the entity integrity of the supplier table  use TPC\_H;  go  create table suppliar2(  supp2key int,  supp2\_name char(30),  supp2\_address char(30),  supp2\_nation decimal(11) references SALES.NATION(N\_NATIONKEY),  supp2\_phone char(30),  supp2\_accbal real,  comment varchar(100),  constraint supp\_2 primary key (supp2key)  );    B . Defines the entity integrity of the table REGION  use TPC\_H;  go  create table region2(  region2key int,  region2name char(30),  comment varchar(100),  constraint regi\_2 primary key(region2key)  );     1. Define entity integrity after defining tables   use TPC\_H;  go  create table part2 (  part2\_partkey int not null,  part2\_name char(30),  mgmt char (50),  part2\_brand varchar(40),  part2\_type varchar(40),  size int,  container char(10),  retailprice real,  comment varchar(100),  );  alter table part2  add constraint partkey\_2 primary key(part2\_partkey)     1. Delete integrity   use TPC\_H;  go  alter table part2  drop constraint partkey\_2;     1. Verify integrity        1. 定义参照完整性，定义参照完整性的违约处理，删除参照完整性。写出两种方式定义参照完整性的 SQL语句：创建表时定义参照完整性、创建表后定义参照完整性。 2. Define entity integrity for Nation3, and redefine supplier3's column-level referential integrity on Nation3   use TPC\_H;  go  create table nation3 (  nationkey int primary key,  name char(40),  regionkey int,  comment varchar(100),  );  create table supp3(  supkey int primary key,  namr char (40),  address char(30),  nation int references nation3 (nationkey),  phone char(20),  accbal real,  comment varchar(100),  );     1. After defining entity integrity for Nation3, we define Supplier4's table-level referential integrity on Nation3   use TPC\_H;  go  create table supp4(  supkey int primary key,  name char(40),  address char(40),  nation int,  phone char(30),  accbal real,  comment varchar(100),  constraint nation\_3 foreign key(nation) references nation3 (nationkey)  );    2. Define referential integrity after creating the table  use TPC\_H;  go  create table nation4(  nationkey int primary key,  name char(40),  regionkey decimal(11),  comment varchar(100),  );  alter table nation4  add constraint regionkey\_4 foreign key (regionkey) references SALES.REGION(R\_REGIONKEY)     1. Defines referential integrity for default handling   use TPC\_H;  go  create table cust3(  custkey int primary key,  name char(40),  address char(40),  nationkey decimal(11),  phone char(20),  accbal real,  mgmt char(10),  comment varchar(100),  constraint nationkey\_3 foreign key (nationkey) references SALES.NATION(N\_NATIONKEY)  on delete cascade  on update set null  );    Remove referential integrity  use TPC\_H;  go  alter table nation4  drop constraint regionkey\_4;  ii. 设计具体应用语义，选择 NULL/NOT NULL、DEFAULT、UNIQUE、CHECK 等，定义属性上的约束条件。   * + - 1. Define NULL/NOT NULL constraints   use TPC\_H;  go  create table nation2(  nationkey int primary key,  name char(40),  regionkey decimal(11) references SALES.REGION(R\_REGIONKEY),  comment varchar(100)  );     * + - 1. Define the attribute UNIQUE constraint   A constraint that the region name defining the REGION table must be unique  use TPC\_H;  go  create table region2(  regionkey int primary key,  name char(40),  comment varchar(100)  );    Use the CHECK  Use CHECK to define the constraints that some of the properties in the order table should satisfy  use TPC\_H;  go  create table order2(  orderkey int primary key,  custkey int,  orderstatus char (1),  orderdate datetime,  orderpriority char (10),  clerk char(10),  shippriority int,  comment varchar(100)  );    实验总结及体会：  通过本次实验，我学到了任何关系数据库系统都应该支持实体完整性和参照完整性，此外，不同关系数据库根据应用环境的不同，往往还需要一些特殊的条件，用户定义的完整性就是针对某一具体关系数据库的约束条件，它反映某一具体数据必须满足的语义要求，个人认为正是有了用户完整性，数据才会更加严谨，使用起来也会更加方便。 | | | | | | |