### JAVA第二阶段—DAY02-JAVA案例

1. 列举数据操作语言DML

* 需求

列出DML有几种类型

列出DML每种类型的语法，包括实例

* 示例代码

-- 插入数据

INSERT INTO student (id, NAME, age, sex) VALUES (1, '张三', 20, '男');

INSERT INTO student (NAME, id, age, sex, address) VALUES ('李四', 2, 23, '女', '广州');

INSERT INTO student VALUES (3, '王五', 18, '男', '北京');

-- 查询数据

select str\_to\_date('08.09.2008 08:09:30', '%m.%d.%Y %h:%i:%s'); -- 2008-08-09 08:09:30

select maketime(12,15,30); -- '12:15:30'

select date\_format('2008-08-08 22:23:01', '%Y-%m-%d %H:%i:%s'); -- 20080808222301

INSERT INTO student2 SELECT \* FROM student;

-- 修改数据

UPDATE student SET sex='男';

UPDATE student SET sex='男' WHERE id=2;

-- 删除数据

DELETE FROM student;

DELETE FROM student WHERE id=3;

1. 列举各个约束创建修改删除方法

* 需求

列出约束有几种类型

列出每种约束达成的效果

列出每种类型约束的创建、删除语法，包括实例

* 示例代码

-- 非空约束

create table tc\_1(

aaa int not null,

bbb varchar(20) not null,

ccc datetime not null

);

alter table tc\_1 modify aaa int;

alter table tc\_1 modify bbb varchar(20) not null;

SELECT \* FROM information\_schema.TABLE\_CONSTRAINTS where table\_schema='test';

-- 唯一性约束，列约束定义模式

create table tc\_2(

aaa int unique,

bbb varchar(20) unique,

ccc datetime unique

);

--表约束定义模式

--表级约束可以给约束起名字(方便以后通过这个名字来删除这个约束)

create table tc\_3(

aaa int,

bbb varchar(20),

ccc datetime,

unique(aaa),

unique(bbb),

constraint u\_ccc unique(ccc)

);

--组合约束模式

create table tc\_4(

aaa int,

bbb varchar(20),

ccc datetime,

unique(aaa, bbb),

unique(ccc)

);

-- 找到对应的唯一约束名称

SELECT \* FROM information\_schema.TABLE\_CONSTRAINTS where table\_schema='test'

and table\_name='tc\_3';

-- 唯一约束同时也是索引，还可以用索引方式找到

show index from tc\_3;

-- 用删除索引的方法来删除唯一约束

drop index u\_ccc on tc\_3;

-- 在删除表时会自动删除表上全部的唯一约束

--以约束模式添加

ALTER TABLE tc\_3 ADD unique(ccc);

--以索引模式添加

create unique index u\_ccc on tc\_3 (ccc);

-- 默认值约束

create table tc\_5(

aaa int default 0,

bbb varchar(20) default 'aaa',

ccc datetime default '2020-1-1'

);

insert into tc\_5 values();

alter table tc\_5 modify aaa int default 5;

alter table tc\_5 modify bbb varchar(20) default 'bbb';

alter table tc\_5 modify aaa int default null;

alter table tc\_5 modify bbb varchar(20) default null;

-- 在删除表时会自动删除表上全部的默认约束

create table tc\_7(

aaa int,

bbb varchar(20),

ccc datetime

);

alter table tc\_7 add primary key(aaa);

alter table tc\_7 add primary key(aaa, bbb);

alter table tc\_7 drop primary key;

-- 在删除表时会自动删除表上的主键约束

-- 检查约束

create table tc\_9(

aaa int primary key,

bbb varchar(20),

ccc datetime,

check(aaa > 100 and aaa<1000)

);

create table tc\_9(

aaa int primary key,

bbb varchar(20),

ccc datetime,

constraint c\_tc\_9 check(aaa > 100 and aaa<1000)

);

insert into tc\_9 values(123, 'aaa', '1-2-3');

-- 不满足约束的数据，无法插入记录表

insert into tc\_9 values(23, 'aaa', '1-2-3');

insert into tc\_9 values(1234, 'aaa', '1-2-3');

create table tc\_9(

aaa int primary key,

bbb varchar(20),

ccc datetime,

constraint c\_tc\_9a check(aaa > 100 and aaa<1000),

constraint c\_tc\_9b check(bbb in ('男', '女')),

constraint c\_tc\_9c check(ccc > '2000-1-1')

);

insert into tc\_9 values(123, '男', '2001-2-3');

-- 不满足约束的数据，无法插入记录表

insert into tc\_9 values(223, 'aaa', '2001-2-3');

insert into tc\_9 values(223, '女', '1995-2-3');

alter table tc\_9 add constraint c\_tc\_9d check(ccc < '2020-1-1');

insert into tc\_9 values(223, '女', '2025-2-3');

alter table tc\_9 drop check c\_tc\_9d;

alter table tc\_9 add constraint c\_tc\_9d check(ccc < '2020-1-1');

-- 自动增长

create table if not exists my\_auto\_increment(

id int primary key auto\_increment,

name varchar(10) not null

) charset utf8;

alter table my\_auto\_increment modify id int;

set auto\_increment\_increment = 5;

1. 列举出交叉查询、内关联、左外关联、右外关联，自然关联、自关联的例子

* 需求

列出关联查询有几种类型

列出每种类型关联查询的语法，包括实例

* 参考答案

-- 交叉关联

select a.teacher\_name, b.student\_name from teacher a

cross join student b;

-- 内关联

select a.teacher\_name, b.student\_name

from teacher a

INNER JOIN student b on a.id = b.teacher\_id;

-- 左外关联

select a.teacher\_name, b.student\_name

from teacher a

left join student b on a.id = b.teacher\_id;

-- 右外关联

select a.teacher\_name, b.student\_name

from teacher a

right join student b on a.id = b.teacher\_id;

-- 自然关联

select a.teacher\_name, b.student\_name

from teacher a

natural join student b;

select a.teacher\_name, b.student\_name

from teacher a

natural left outer join student b;

select a.teacher\_name, b.student\_name

from teacher a

natural right outer join student b;

-- 自关联

select ta.student\_name, tb.student\_city

from studnet ta， student tb

where ta.student\_id = tb.student\_id;