---------------------------------------------------

计划课程安排：

FUNDAMENTAL(7天)         该阶段已结束

OOP(11天)           该阶段已结束

JAVASE01(10天) 该阶段已结束

JAVASE02(12天) 该阶段已结束

WEBBASIC(11天) 该阶段已结束

JQUERY(2天) 该阶段已结束

DATABASE(6天) ←当前为该阶段第2天

JDBC(3天)

SERVLETJSP(12天)

SPRINGMYBATIS01(8天)

AJAX(2天)

SPRINGMYBATIS02(12天)

LINUX(2天)

NGINX(2天)

REDIS(2天)

T-STORE(5天)

预计合计：107天

预计剩余：52天

注：上述信息随时可能发生更改，仅供参考。

---------------------------------------------------

**2019年5月17日**

**Java就业班-第65天**

**8:30实训室2-早自习**

今日晨讲主题——简单算法

抄下来的晨讲代码：

import java.util.HashMap;

import java.util.Map;

public class Test01 {

public Map<Integer,Integer> twoSum (int[] arr,int a){

Map<Integer,Integer> map = new HashMap<>();

for (int i = 0; i < arr.length; i++) {

for (int j = i+1; j < arr.length; j++) {

if(a-arr[i]==arr[j]) {

map.put(arr[i], arr[j]);

}

}

}

return map ;

}

public static void main(String[] args) {

int[] arr = new int[] {1,2,3,4,5};

int a = 7 ;//根据数组里的数，求出结果a有几种计算方法。

Test01 t = new Test01();

Map<Integer, Integer> b = t.twoSum(arr, a);

System.out.println(b);

}

}

**9:10实训室2-正课**

开始上课。

复习练习：

1. 创建数据库db2并使用
2. 创建员工表emp 有id和name字段
3. 修改表名t\_emp
4. 添加部门名称字段dept
5. 在name后面添加age字段
6. 插入3个人（Kamui,Subaru,Luna）id分别为1,2,3部门都是缺B乐，再插入4个人(Naruto,Sasuke,Sakura,Kakashi)，id分别为4567，部门为AcFun。
7. 给表再添加工资字段，添加到部门字段的前面
8. 修改AcFun部门的工资为100
9. 修改Kakashi名字为稻草人
10. 修改缺B乐为猴山缺B乐
11. 删除稻草人
12. 删除AcFun的人
13. 修改所有人工资为80
14. 删除所有人
15. 删除表

代码：

//创建数据库

create database db2 ;

use db2 ;

//创建表

create table emp(

id int,

name varchar(10)

);

//修改表名t\_emp

rename table emp to t\_emp;

//添加字段

alter table t\_emp add dept varchar(10);

alter table t\_emp add age int after name ;

//插入数据

insert into t\_emp (id,name,dept) values(1,'Kamui','缺B乐'),(2,'Subaru','缺B乐'),

(3,'Luna','缺B乐'),(4,'Naruto','AcFun'),(5,'Sasuke','AcFun'),(6,'Sakura','AcFun'),

(7,'Kakashi','AcFun');

//插入字段

alter table t\_emp add salary int after age;

//修改数据

update t\_emp set salary=100 where dept='AcFun';

update t\_emp set name='稻草人' where name='Kakashi';

update t\_emp set dept='猴山缺B乐' where dept='缺B乐';

//删除数据

delete from t\_emp where name='稻草人';

delete from t\_emp where dept='AcFun';

//修改数据

update t\_emp set salary=80;

//删除数据

delete from t\_emp ;

//删除表

drop table t\_emp ;

**10:11实训室2-正课**

开始学习新内容——主键约束

//格式1：主键约束

create table t1(id int primary key , name varchar(10));

insert into t1 values(1,'AAA');

insert into t1 values(1,'BBB'); //报错 不能重复

insert into t1 values(null,'CCC'); //报错 不能为null

//格式2：主键约束+自增

create table t2(id int primary key auto\_increment, name varchar(10));

insert into t2 values(null,'aaa');//1

insert into t2 values(2,'bbb'); //2

insert into t2 values(10,'ccc'); //10

insert into t2 values(11,'ddd'); //11

delete from t2 where id>=10 ;

insert into t2 values(12,'eee'); //12

——注释comment

//格式：

create table t3(id int primary key auto\_increment comment '这是id',name varchar(10) comment '名字');

show create table t3;

——`的作用

**11:28实训室2-正课**

——冗余

//格式：

create table emp(id int primary key auto\_increment , name varchar(10) , age int , gender varchar(5) , salary int ,

dept\_id int);

create table dept( id int primary key auto\_increment , name varchar(10) , parent\_id int);

insert into dept values(null , 'AcFun' , null),(null,'文章区',1),(null,'工作板块',2),(null,'学习日记',1),(null,'第65天',4);

insert into emp values(null,'Kamui',25,'男',100,3),(null,'Subaru',19,'女',200,5);

练习：

商品表digtal 分类表office

1. 保存家电分类下 电视机分类下的 小米电视 价格（price）1888 库存（num）200
2. 办公用品分类下的 打印机分类下的 惠普打印机 价格1500 库存100

代码：

//创建表

create table digtal (id int primary key auto\_increment , name varchar(10) , price int , num int ,office\_id int);

create table office (id int primary key auto\_increment , name varchar(10) , parent\_id int);

//插入数据

insert into office( name , parent\_id ) values('家电分类' , null),( '电视分类' , 1) , ( '办公用品' , null) , ( '打印机分类' , 3);

insert into digtal values(null , 'AC电视' , 1888 , 200 , 1),(null , 'AC打印机' , 1500 , 100 , 3);

**11:50实训室2-午休时间**

这该死的机构终于肯开空调了。

虽然仍然不算凉快，但至少我的内裤不会再反复进行被汗湿和被吹干的循环了。

**14:03实训室2-正课**

继续上课。

——事务

//练习:验证转账

create table user(id int primary key auto\_increment, name varchar(10),money int,status varchar(5));

insert into user values(null,'Kamui',50,'冻结'),(null,'Subaru',20000,'正常');

//转账SQL:

update user set money= money-2000 where id=2 and status='正常';

update user set money= money+2000 where id=2 and status='正常';

//回滚

rollback;

//有事务保护下提交流程

insert into user values(null,'Kamui',20,'正常');

//开启事务

begin;

//Subaru-5000

update user set money= money-5000 where id=2 and status='正常';

//Kamui+5000

update user set money= money-5000 where id=3 and status='正常';

//提交事务命令

commit;

//保存回滚点

begin;

update user set money=200 where id=3;

savepoint s1;

update user set money=300 where id=3;

savepoint s2;

update user set money=400 where id=3;

rollback to s2;

**15:31实训室2-正课**

——SQL分类

——数据类型

**17:00实训室2-正课**

今天讲的内容比写的多，所以下午基本上就没怎么写代码了，等会把今天的笔记补上应该就结束了。

…………

今日课程笔记：

###主键约束

- 主键：表示数据唯一性的字段称为主键

- 约束： 创建表时给表字段添加的限制条件

- 主键约束： 限制值唯一且非空

- 格式： create table t1(id int primary key,name varchar(10));

insert into t1 values(1,'AAA');

insert into t1 values(1,'BBB');//报错 不能重复

insert into t1 values(null,'CCC');//报错 不能为null

###主键约束+自增

- 从历史最大值基础上+1 ， 自增数值只增不减

- 格式： create table t2(id int primary key auto\_increment,name varchar(10));

insert into t2 values(null,'aaa'); //1

insert into t2 values(2,'bbb'); //2

insert into t2 values(10,'ccc'); //10

insert into t2 values(null,'ddd'); //11

delete from t2 where id>=10;

insert into t2 values(null,'eee'); //12

###注释 comment

- 格式： create table t3(id int primary key auto\_increment comment '这是id啊',name varchar(10) comment '名字');

查看： show create table t3;

###`的作用

- 用于修饰表名和字段名 可以省略

create table `t4`(`id` int,`name` varchar(10));

###冗余

- 由于表设计不够合理导致出现的大量重复数据成为冗余。通过合理拆分表的形式解决冗余问题

- 创建表保存一下数据：

### 练习：

商品表item 分类表category

1. 保存家电分类下电视机分类下的小米电视 价格(price)1888 库存(num)200

2. 办公用品分类下的打印机分类下的惠普打印机 价格1500 库存100

- 创建表：

create table item(id int primary key auto\_increment,name varchar(10),price int,num int,category\_id int);

create table cateogry(id int primary key auto\_increment,name varchar(10),parent\_id int);

- 插入数据：

insert into cateogry(name,parent\_id) values('家电',null),('电视机',1),('办公用品',null),('打印机',3);

insert into item values(null,'小米电视',1888,200,2),(null,'惠普打印机',1500,100,4);

###事务

- 什么是事务： 事务是数据库中执行同一业务多条SQL语句的工作单元，事务可以保证多条SQL语句全部执行成功或全部执行失败。

- 和事务相关的SQL：

1. 开启事务： begin;

2. 提交： commit; 将内存中的改动一次性提交到磁盘中

3. 回滚： rollback; 将内存中的改动清除，回滚到正确的点

- 验证转账：

create table user(id int primary key auto\_increment,name varchar(10),money int,status varchar(5));

insert into user values(null,'Kamui',50,'冻结'),(null,'Subaru',20000,'正常');

- 转账SQL：

update user set money=money-2000 where id=2 and status='正常';

update user set money=money+2000 where id=1 and status='正常';

- 有事务保护的情况下 回滚流程：

1. 开启事务

begin;

2. Subaru-2000

update user set money=money-2000 where id=2 and status='正常';

3. 此时当前终端查询数据已经发生改变（因为查询到的是内存中的数据），开启另外一个终端查询会发现数据并未改变

4. Kamui+2000

update user set money=money+2000 where id=1 and status='正常';

5. 当判断出有一条执行失败，则需要回滚

rollback;

- 有事务保护下提交流程：

insert into user values(null,'Luna',20,'正常');

1. 开启事务

begin;

2. Subaru-5000

update user set money=money-5000 where id=2 and status='正常';

3. Luna+5000

update user set money=money+5000 where id=3 and status='正常';

4. 当判断出两条全部执行成功则执行提交事务命令

commit;

- 保存回滚点

begin;

update user set money=200 where id=3;

savepoint s1;

update user set money=300 where id=3;

savepoint s2;

update user set money=400 where id=3;

rollback to s2;

###SQL分类

####DDL Data Definition Language 数据定义语言

-truncate：删除表并创建新表 自增数值清零

truncate table 表名;

- 包括： create 、 alter 、 drop 、 truncate

- 不支持事务

####DML Data Manipulation Language 数据操作语言

- 包括： insert 、 update 、 delete 、 select（DQL）

- 支持事务

####DQL Data Query Language 数据查询语言

- 只包括select

####TCL Transaction Control Language 事务控制语言

- 包括： begin 、 commit 、rollback、 savepoint xxx 、 rollback to xxx;

####DCL Data Control Language 数据控制语言

- 分配用户权限相关的SQL

###数据类型

1. 整数： int(m) 和 bigint(m) ,m代表显示长度，不足m长度时补零，需要结合zerofill关键字使用

create table t\_int(id int,age int(10) zerofill);

insert into t\_int values(1,20);

select \* from t\_int;

2. 浮点数: double(m,d) m代表总长度 d代表小数长度, 超高精度浮点数decimal(m,d) 当涉及超高精度运算时使用。

25.234 m=5 d=3

3. 字符串：

- char(m) 固定长度,最大长度255 好处：执行效率高

- varchar(m) 可变长度, 好处：节省资源， 最大长度65535 但是超高255建议使用text

- text(m) 可变长度, 最大长度65535

4. 日期

- date：只能保存年月日

- time：只能保存时分秒

- datetime：年月日时分秒 最大值：9999-12-31 ，默认值为null

- timestamp：时间戳 年月日时分秒 最大值：2038-1-19 ，默认值为当前系统时间

create table t\_date(t1 date,t2 time,t3 datetime,t4 timestamp);

insert into t\_date values('2019-05-17',null,null,null);

insert into t\_date values

(null,'16:29:00','2019-05-17 16:30:00',null);

5. 其它

###导入sql文件

1. windows系统

把文件放到d盘根目录，然后执行

source d:/tables.sql;

2. linux系统

把文件放到桌面, 然后执行

source /home/soft01/桌面/tables.sql;

- 验证：

show tables;

#### is null 和 is not null

1. 查询没有领导的员工姓名和工资

select ename,sal from emp where mgr is null;

2. 查询有奖金的员工姓名，工资，奖金

select ename,sal,comm from emp where comm is not null;

####别名

select ename from emp;

select ename as '姓名' from emp;

select ename '姓名' from emp;

select ename 姓名 from emp;

###课程回顾

####day01

1. 数据库相关

- show databases;

- create database db1 character set utf8/gbk;

- show create database db1;

- drop database db1;

- use db1;

2. 表相关

- create table t1(name varchar(10),age int) engine=myisam/innodb charset=utf8/gbk;

- show tables;

- show create table t1;

- desc t1;

- drop table t1;

- truncate table t1;

- rename table t1 to t2;

- alter table t1 engine=myisam/innodb charset=utf8/gbk;

- alter table t1 add 字段名 类型 first/after xxx;

- alter table t1 drop 字段名;

- alter table t1 change 原名 新名 新类型;

- alter table t1 modify 字段名 新类型 first/after xxx;

3. 数据相关

- insert into t1 (name,age) values(值1,值2),....,...;

- select \* from t1 where 条件;

- update t1 set xxx=xxx where 条件;

- delete from t1 where 条件;

####day02

1. 主键约束+自增 primary key auto\_increment 唯一且非空

从历史最大值+1 只增不减

2. 注释 comment

3. `用于修饰表名和字段名 可以省略

4. 冗余： 由于表设计不够合理导致的大量重复数据，通过合理拆分表解决

5. 事务：数据库中执行同一业务多条SQL语句的工作单元，可以保证多条SQL语句全部执行成功或全部失败

- 相关SQL： begin; commit; rollback; savepoint xxx; rollback to xxx;

6. SQL分类

- DDL 数据定义语言 包括： create drop alter truncate 不支持事务

- DML 数据操作语言 包括： insert update delete select 支持事务

- DQL 数据查询语言 包括： select

- TCL 事务控制语言 包括：begin commit rollback savepoint xxx rollback to xxx;

- DCL 数据控制语言 用于分配用户权限相关的SQL

7. 数据类型

- 整数： int 和 bigint(m) m代表显示长度 需要结合zerofill使用

- 浮点数：double(m,d) m代表总长度 d代表小数长度 超高精度浮点数decimal

- 字符串： char不可变长度 最大255 执行效率高 varchar 可变长度 最大65535 超过255建议使用text text可变长度 最大 65535

- 日期： date年月日 time时分秒 datetime 最大9999-12-31 默认null, timestamp 最大 2038-1-19 默认值系统时间

- 其它

8. 导入SQL文件 source 路径

9. is null 和 is not null

10. 别名

**17:50实训室2-下课**

下课回宿舍。

今天的课程说实话讲的很一般，有很多内容我也没听进去，班上很多人也睡过去了。这个讲师的技术水平可能很高，但讲课水平是真的太次了。

**18:12宿舍-休息**

今天也是这周最后一天课了，明天周六自习计划还是只自习上午半天，明天下午因为需要回去当工具人，所以明天应该不会有太多记录。

有个问题想问一下老哥们，各位还在学习阶段或工作初期时独自写过的最牛逼哄哄的项目是什么呢？实现的什么功能呢？能否分享一下经验呢，我想学习学习。

今日感想总结：我其实很想知道数据库存储数据的原理，但是百度了一时半会又看不太懂，是不是要买相关书籍看一下比较好呢。

今日评分：2/10分

今天就到这里，明天继续投稿。

感谢支持！

================================================================

**Acer程序员群：790482850**

**up主的java学习日记word文件及全部代码（随日记同步更新）：**

Adress：github.com/wt62635