

Project 1

Database

- 一个数据库包含多个表
 - 数据库: db/activity_1.c
 - 每个表包含多个记录
 - 每个记录有多个属性
 - db/activity_1.c中的定义
 - 用一个structure来定义每个记录的属性

```
struct T_student {  
    double student_id;  
    char last_name[9];  
    char first_name[8];  
    double age;  
    char sex[2];  
    double major;  
    double advisor;  
    char city_code[4];  
};
```

Database

- db/activity_1.c中的定义
 - 用一个结构体的数组来给出一个表的数据

```
struct T_student
```

```
student[] = {
```

```
    { 1001, "Smith", "Linda", 18, "F", 600, 1121, "BAL" },
```

```
    { 1002, "Kim", "Tracy", 19, "F", 600, 7712, "HKG" },
```

```
    { 1003, "Jones", "Shiela", 21, "F", 600, 7792, "WAS" },
```

```
    { 1004, "Kumar", "Dinesh", 20, "M", 600, 8423, "CHI" },
```

```
    { 1005, "Gompers", "Paul", 26, "M", 600, 1121, "YYZ" },
```

```
    { 1006, "Schultz", "Andy", 18, "M", 600, 1148, "BAL" },
```

```
    ...
```

- 访问第1个记录的last_name
 - student[0].last_name

Relational database

- 外键 (foreign key)
 - 当一个记录的属性是另一个记录的时候

```
struct T_participates_in {  
    double student_id;    // --> student.student_id  
    double activity_id; // --> activity.activity_id  
};
```

```
struct T_student {  
    double student_id;  
    char last_name[9];  
    char first_name[8];  
    double age;  
    char sex[2];  
    double major;  
    double advisor;  
    char city_code[4];  
};
```

提交

- <http://ecw.sysu.edu.cn:8000>
 - 用户名：你的中文名字
 - 初始密码：1234
 - 请登录后立即修改密码
 - 修改密码后务必牢记

提交

- 点击左面的“完成题目”开始做题



1

数据库

完成题目

数据库: [customers_and_addresses](#)

问题: [What are the state and coun](#)

```
1 #include "customers_and_ad
2 #include <stdio.h>
3
4 int main() {
5
6 }
```

提交

- 点击数据库
– 查看数据库关系图

完成题目

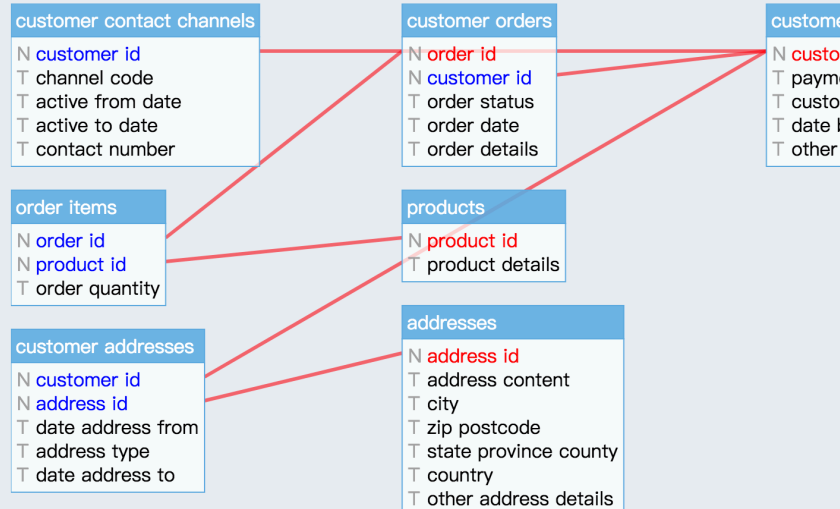
1

数据库: customers_and_addresses

问题: What are the state and country o

查看数据库

数据库名 customers_and_addresses



N 表示 number

T 表示 text

红色表示 主键

蓝色表示 外键

提交

- 点击数据库框查看数据库内容

1

customer addresses

- N customer id
- N address id
- T date address from
- T address type
- T date address to

addresses

- N address id
- T address content
- T city
- T zip postcode
- T state province cou
- T country
- T other address deta

customer addresses

	customer id	address id	date address	address type	
1	2	11	1985-03-29 2	Billing	
2	13	11	2010-08-25 0	Billing	
3	2	14	2010-12-26 0	Residential	
4	15	15	1995-12-16 0	Billing	
5	6	7	1979-10-08 1	Residential	

提交

- 运行程序，做对后进入下一题

完成题目

数据库: [city_record](#)

问题: Which three cities have the largest regional population?

保存并测试

```
1 #include "city_record"
2 #include <stdio.h>
3
4 #define len(t) (sizeof(t) / sizeof(struct T_##t))
5
6 typedef int (* array_function)(void * array, int i, int j);
7
8 void sort(void * array, int length, int reverse, array_function compare, array_function swap) {
9     for (int i = length - 1; i > 0; -- i) {
10         for (int j = 0; j < i; ++ j) {
11             if ((reverse == 0 && compare(array, j, j + 1) > 0) || (reverse == 1 && compare(array, j, j + 1) < 0)) {
12                 swap(array, j, j + 1);
13             }
14         }
15     }
16 }
17
18 int compare(void * array, int i, int j) {
19     struct T_city * city = (struct T_city *)array;
20     if (city[i].regional_population > city[j].regional_population) return 1;
21     if (city[i].regional_population < city[j].regional_population) return -1;
22     return 0;
23 }
24
25 int swap(void * array, int i, int j) {
26     struct T_city * city = (struct T_city *)array;
27     struct T_city temp = city[i];
28     city[i] = city[j];
29     city[j] = temp;
30     return 0;
31 }
32
33 void print(int top) {
34     if (top > len(city)) top = len(city);
35     for (int i = 0; i < top; ++ i) {
36         printf("%s\n", city[i].city);
37     }
38 }
39
40 int main() {
41     sort(city, len(city), 1, compare, swap);
42     print(3);
43 }
```

进入下一题

提交

- 做不出放弃该题，继续做下一题
 - 放弃后将没有机会再做该题

完成题目

数据库: [hospital_1](#)

问题: [Tell me the employee id of the head of the department with the least employees.](#)

 保存并测试

```
1 #include "hospital_1"
2 #include <stdio.h>
3
4 int main() {
5
6 }
```

1

 放弃此题

提交

- 查看排行

排行榜

1

提交

用户	通过题数
TA	2