

```
1  /**
2   * file: musician.c
3   *
4   * Created by hengxin on 12/12/21.
5   */
6
7  #include <stdio.h>
8  #include <stdlib.h>
9  #include <string.h>
10
11 typedef enum gender {
12     MALE, FEMALE
13 } Gender;
14
15 typedef struct score {
16     int c_score;
17     int java_score;
18     int python_score;
19 } Score;
20
21 typedef struct musician {
22     char *name;
23     // enum {MALE, FEMALE} gender;
24     Gender gender;
25
26     char *album;
27
28     Score score;
29 } Musician;
30
31 void PrintMusician(const Musician *m);
32 Musician MakeMusician(char *name, Gender gender,
33                       char *album,
34                       int c_score, int java_score, int python_score);
35 int TotalScore(const Musician *m);
36 int CompareMusicians(const void *m1, const void *m2);
37
38 int main() {
39     Musician luo = {
40         "Luo Dayou", MALE,
41         "0000",
42         20, 30, 40
43     };
44
45     Musician cui = {
46         .name = "Cui Jian",
47         .gender = MALE,
48         .album = "00000000",
49         .score = {
50             .c_score = 10,
51             .java_score = 20,
52             .python_score = 30
53         }
54     }
```

```

54     };
55
56     Musician zhang = {
57         .name = "Zhang Chu",
58         .gender = MALE,
59         // .album = "周杰伦周杰伦",
60         .score = {
61             .c_score = 0,
62             .java_score = 10,
63             .python_score = 20
64         }
65     };
66     zhang.album = malloc(50 * sizeof *zhang.album);
67     strcpy(zhang.album, "周杰伦周杰伦");
68
69     Musician musicians[] = {luo, cui, zhang};
70     int len = sizeof musicians / sizeof *musicians;
71     for (int i = 0; i < len; i++) {
72         PrintMusician(musicians + i);
73     }
74
75     qsort(musicians, len, sizeof(*musicians), CompareMusicians);
76     for (int i = 0; i < len; i++) {
77         PrintMusician(musicians + i);
78     }
79
80     Musician li = zhang;
81     li.name = "Li Chaochao";
82     // It is even possible for an enum variable
83     // to hold an integer that does not represent any of the
84     // enumeration values.
85     li.gender = 'M';
86     // Danger! Now li.album and zhang.album point to the same album.
87     strcpy(li.album, "周杰伦周杰伦");
88
89     PrintMusician(&li);
90     PrintMusician(&zhang);
91
92     free(zhang.album);
93 }
94
95 void PrintMusician(const Musician *m) {
96     printf("\n");
97     printf("%s\n%c\n%s\n%d\t%d\t%d\n",
98         m->name,
99         m->gender == MALE ? 'M' : 'F',
100         m->album,
101         m->score.c_score, m->score.java_score, m->score.python_score
102     );
103     printf("\n");
104 }
105
106 int CompareMusicians(const void *m1, const void *m2) {

```

```
105     int score_1 = TotalScore(m1);
106     int score_2 = TotalScore(m2);
107
108     if (score_1 > score_2) {
109         return 1;
110     }
111
112     if (score_1 < score_2) {
113         return -1;
114     }
115
116     return 0;
117 }
118
119 int TotalScore(const Musician *m) {
120     return m->score.c_score + m->score.java_score + m->score.
        python_score;
121 }
122
123 Musician MakeMusician(char *name, Gender gender,
124                       char *album,
125                       int c_score, int java_score, int python_score
126 ) {
127     Musician m;
128     m.name = malloc(strlen(name) + 1);
129     strcpy(m.name, name);
130
131     m.gender = gender;
132
133     m.album = malloc(strlen(album) + 1);
134     strcpy(m.album, album);
135
136     m.score.c_score = c_score;
137     m.score.java_score = java_score;
138     m.score.python_score = python_score;
139
140     return m;
141 }
```