# 语言规范

使用pascal规则命名类名，即首字母要大写。

1. 使用能够反映类功能的名词或名词短语命名类。
2. 3.不要使用“I”、“C”、“\_”等特定含义前缀。
3. 自定义异常类应以Exception结尾。
4. 文件名要能反映类的内容，最好是和类同名
5. 类字段变量名前可加“\_”前缀。
6. 类中访问修饰符或功能相同的方法应该放在一起， 且公共或实现接口的方法在前。
7. 使用名词定义属性，属性使用Pascal规则，首字符大写
8. 属性和相应字段名称要关联， 可以使用“重构”菜单来生成属性。
9. 参数采用camel规则命名，且首字符小写。
10. 使用描述性参数名称，参数名称应当具有最够的说明性。
11. 接口定义使用Pascal规则，且必须以大写“I”开头。
12. 接口名称要有意义，中间不要有下划线“\_”等字符。
13. 如果类实现了接口，名称尽量和接口相同， 只是省掉“I”字符。
14. 定义事件的委托要使用EventHandler后缀，且包括sender和e两个参数。
15. 命名空间名称尽量反映其内容所提供的整体功能。
16. 文件都包含文件头， 要说明文件名、作者、创建时间、变更记录。
17. 标识符应当直观可望文知意，不提倡使用任何缩写。
18. 缩写可自行定义，一般取单词的前/后字符组成，以含义直观为准则。
19. 每行语句至少占一行，如果语句过长（超过一屏），则该语句断为两行显示。
20. 把相似的内容放在一起，比如字段、属性、方法、事件等，使用“#region--#endregion”命令分组。
21. 每个方法的源程序行数原则上应该少于200行。(如果超过过多，则需要另写一方法)
22. 语句嵌套层次不得超过3层。
23. switch语句，case后面必须接break。
24. 不同类型的操作符混合使用时，使用括号给出优先级。
25. 循环、判断语句的程序块部分用花括号括起来， 即使只有一条语句。
26. 在switch语句中总是要有default字句，建议使用断言。
27. 不允许变量名、类名、属性名、 方法名等与系统标识符重名。
28. 数据库中每一张表对应一个实体类/数据传输对象（DTO）。
29. 实体类名称使用表名，也可带有Dto后缀。
30. 系统输入、资源操作（如内存分配、文件及目录操作）、网络操作（如通信、调用等）、任务间的操作（如通信、调用等）时必须进行错误、超时、或则异常处理。

# 代码示例

#include<stdio.h>

struct dorm

{

int num;

int members\_num;

int Health\_score;

char member1[10];

char member2[10];

char member3[10];

char member4[10];

};

struct student

{

int num;

char name[10];

int dorm\_num;

char gender[10];

char major[10];

char class\_name[10];

char instructor[10];

};

static struct dorm dm[100];

static struct student st[400];

static int d\_i=2;

static int s\_i=2;

void pri\_dorm(struct dorm d)

{

printf("\n 宿 舍 号:%d\n",d.num);

printf("现住人数:%d\n",d.members\_num);

printf("卫生分数:%d\n",d.Health\_score);

printf("宿舍成员:\n");

if(d.member1[0]!=NULL) printf("%s\n",d.member1);

if(d.member2[0]!=NULL) printf("%s\n",d.member2);

if(d.member3[0]!=NULL) printf("%s\n",d.member3);

if(d.member4[0]!=NULL) printf("%s\n",d.member4);

printf("\n");

}

void pri\_student(struct student s)

{

printf("\n 学 号:%d\n",s.num);

printf("姓 名:%s\n",s.name);

printf("宿舍号:%d\n",s.dorm\_num);

printf("性 别:%s\n",s.gender);

printf("班 级:%s\n",s.class\_name);

printf("专 业:%s\n",s.major);

printf("辅导员:%s\n",s.instructor);

printf("\n");

}

int panduansusherenshu(struct dorm d)//通过此函数， 判断宿舍共多少人（宿舍人数最多住四

个人）

{

int i=0;

if(d.member1[0]!=NULL){i++;}

if(d.member2[0]!=NULL){i++;}

if(d.member3[0]!=NULL){i++;}

if(d.member4[0]!=NULL){i++;}

return i;

}

int panduanxueshengxinxi(struct student s)//判断学生信息是否合法

{

int i=1;

int j=0;

int k=0;

if(((s.num/100000000)>0)){

if(!strcmp(s.gender,"男")||!strcmp(s.gender,"女")){

if((s.dorm\_num/100)>0){

i=0;

}

}

}

for(j=0;j<sizeof(dm)/52;j++){

if(dm[j].num==s.dorm\_num)

{

k=1;

break;

}

}

if(k==0) {i=1;}

for(j=0;j<sizeof(st)/60;j++)

{

if(st[j].num==s.num) i=1;

}

for(j=0;j<sizeof(dm)/52;j++)

{

if(dm[j].members\_num==4) {i=1;printf("\n 宿舍人数已满!\n");}

}

return i;

}

void main\_interface()//主界面

{

int i=0;

void interface1();

void interface2();

void interface3();

void interface4();

void interface5();

void interface6();

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\* 宿舍管理系统 \*\*\*\*\*\*\n");

printf("\*\*\* 1.查询宿舍信息 \*\*\*\n");

printf("\*\*\* 2.查询学生信息 \*\*\*\n");

printf("\*\*\* 3.添加入住学生 \*\*\*\n");

printf("\*\*\* 4.删减入住学生 \*\*\*\n");

printf("\*\*\* 5.更改宿舍信息 \*\*\*\n");

printf("\*\*\* 6.更改学生信息 \*\*\*\n");

printf("\*\*\* 7.退出系统 \*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("请输入操作项:");

scanf("%d",&i);

switch(i)

{

case 1:

system("cls");

interface1();

break;

case 2:

system("cls");

interface2();

break;

case 3:

system("cls");

interface3();

break;

case 4:

system("cls");

interface4();

break;

case 5:

system("cls");

interface5();

break;

case 6:

system("cls");

interface6();

break;

case 7:

system("cls");

exit(0);

default:

system("cls");

printf("操作项命令错误,请重新输入!");

main\_interface();

break;

}

}

void interface1()//查找宿舍信息

{

int i=0;//选项选择

int j=0;//记录需要查询的学生学号

int k=0;

int q=0;

printf("\n");

printf("进入宿舍信息查询界面:\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\* 宿舍信息查询 \*\*\*\*\*\*\n");

printf("\*\*\* 1.进入宿舍查询 \*\*\*\n");

printf("\*\*\* 2.返回主界面 \*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("请输入操作项:");

scanf("%d",&i);

switch(i)

{

case 1:

printf("请输入需要查询宿舍的宿舍号:");

scanf("%d",&j);

for(k=0;k<sizeof(dm)/52;k++)

{

if(dm[k].num==j){q=1; break;}

}

if(q==1)

{

system("cls");

pri\_dorm(dm[k]);

printf("\n 返回上一界面\n");

interface1();

}

else

{

system("cls");

printf("\n 不存在您所查询的宿舍号!");

printf("\n 返回上一界面\n");

interface1();

}

break;

case 2:

system("cls");

printf("\n");

printf("进入主界面:\n");

main\_interface();

break;

default :

system("cls");

printf("\n 选项输入错误!\n");

printf("\n 返回上一界面\n");

interface1();

break;

}

}

void interface2()//查找学生信息

{

int i=0;//选项选择

int j=0;//记录学生学号

int q=0;

int k=0;

printf("\n");

printf("进入学生信息查询界面:\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\* 学生信息查询 \*\*\*\*\*\*\n");

printf("\*\*\* 1.进入学生查询 \*\*\*\n");

printf("\*\*\* 2.返回主界面 \*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("请输入操作项:");

scanf("%d",&i);

switch(i)

{

case 1:

printf("请输入需要查询学生的学号:");

scanf("%d",&j);

for(k=0;k<sizeof(dm)/52;k++)

{

if(st[k].num==j){q=1; break;}

}

if(q==1)

{

system("cls");

pri\_student(st[k]);

printf("\n 返回上一界面\n");

interface2();

}

else

{

system("cls");

printf("\n 你查找的学生不存在!");

printf("\n 返回上一界面\n");

interface2();

}

break;

case 2:

system("cls");

printf("\n");

printf("进入主界面:\n");

main\_interface();

break;

default:

system("cls");

printf("\n 选项输入错误!\n");

printf("\n 返回上一界面\n");

interface2();

break;

}

}

void interface3()//添加入住学生

{

struct student ss;

int j=0;

printf("请输入学生的信息:\n");

printf("学 号:");

scanf("%d",&ss.num);

printf("姓 名:");

scanf("%s",&ss.name);

printf("宿舍号:");

scanf("%d",&ss.dorm\_num);

printf("性 别:");

scanf("%s",&ss.gender);

printf("班 级:");

scanf("%s",&ss.class\_name);

printf("专 业:");

scanf("%s",&ss.major);

printf("辅导员:");

scanf("%s",&ss.instructor);

if(panduanxueshengxinxi(ss)==0)

{

s\_i++;

st[s\_i].num=ss.num;

st[s\_i].dorm\_num=ss.dorm\_num;

strcat(st[s\_i].name,ss.name);

strcat(st[s\_i].gender,ss.gender);

strcat(st[s\_i].class\_name,ss.class\_name);

strcat(st[s\_i].instructor,ss.instructor);

strcat(st[s\_i].major,ss.major);

pri\_student(st[s\_i]);

}

else

{

system("cls");

printf("\n\n 输入的学生信息有错误,请重新输入!\n\n");

interface3();

}

for(j=0;j<=sizeof(dm)/52;j++)

{

if(dm[j].num==ss.dorm\_num) break;

}

if(dm[j].member1[0]==NULL) strcat(dm[j].member1,ss.name);

else if(dm[j].member2[0]==NULL) strcat(dm[j].member2,ss.name);

else if(dm[j].member3[0]==NULL) strcat(dm[j].member3,ss.name);

else strcat(dm[j].member4,ss.name);

dm[j].members\_num++;

system("cls");

printf("添加学生成功!\n");

printf("\n 返回主界面!\n");

main\_interface();

}

void interface4()//删除入住学生

{

int i=0;//存储需要删除学生的学号

int j=0;

int k=0;

int q=0;

printf("请输入需要删除的学生学号:\n");

scanf("%d",&i);

for(j=0;j<sizeof(st)/60;j++)

{

if(st[j].num==i){k=1;break;}

}

if(k==0){system("cls");printf("不存在需要删除学生!\n");printf("返回上一界面,请重新输

入!\n");interface4();}

if(k==1){

for(q=0;q<sizeof(dm)/52;q++)

{

if(dm[q].num=st[j].dorm\_num)

{

dm[q].members\_num--;

if(!strcmp(dm[q].member1,st[j].name)){dm[q].member1[0]=NULL;strcmp(dm[q].member1,"

");}

if(!strcmp(dm[q].member2,st[j].name)){dm[q].member2[0]=NULL;strcmp(dm[q].member2,"

");}

if(!strcmp(dm[q].member3,st[j].name)){dm[q].member3[0]=NULL;strcmp(dm[q].member3,"

");}

if(!strcmp(dm[q].member4,st[j].name)){dm[q].member4[0]=NULL;strcmp(dm[q].member4,"

");}

}

}

st[j].num=0;

st[j].name[0]=NULL;strcmp(st[j].name," ");

st[j].class\_name[0]=NULL;strcmp(st[j].class\_name," ");

st[j].dorm\_num=0;

st[j].gender[0]=NULL;strcmp(st[j].gender," ");

st[j].instructor[0]=NULL;strcmp(st[j].instructor," ");

st[j].major[0]=NULL;strcmp(st[j].major," ");

}

system("cls");

printf("删除学生成功,返回主界面!\n");

main\_interface();

}

void interface5()//更改宿舍信息

{

int i=0;//存储宿舍号

int j=0;//储存更改后的分数

int k=0;

int q=0;

printf("请输入需要更改信息的宿舍号:\n");

scanf("%d",&i);

for(k=0;k<sizeof(dm)/52;k++)

{

if(dm[k].num==i){q=1;break;}

}

if(q==1){

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\* 宿舍信息更改 \*\*\*\*\*\*\n");

printf("\*\*\* 更改卫生分数 \*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

while(1)

{

printf("\n%d 宿舍卫生分数更改为:\n",i);

scanf("%d",&j);

if(j>=0&&j<=10) break;

else {printf("输入的分数有错误,请重新输入!\n");}

}

dm[k].Health\_score=j;

system("cls");

printf("更改宿舍信息成功!\n");

printf("\n 返回主界面:\n");

main\_interface();

}

if(q==0){system("cls");printf(" 输 入 的 宿 舍 号 不 存 在 , 请 确 认 后 重 新 输

入!\n\n");interface5();}

}

void interface6()//更改学生信息

{

int i=0;//存储学生学号

int k=0;//判断该学号是否存在

int j=0;

int q=0;//存储选项

struct student ss;

printf("请输入需要被更改的学生的学号:\n");

scanf("%d",&i);

for(j=0;j<sizeof(st)/60;j++)

{

if(st[j].num==i){k=1;break;}

}

if(k==0){printf("该学号不存在,请确认后重新输入!\n");interface6();}

if(k==1)

{

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\*\*\*\*\*\* 学生信息更改 \*\*\*\*\*\*\n");

printf("\*\*\* 1.更改姓名 \*\*\*\n");

printf("\*\*\* 2.更改班级 \*\*\*\n");

printf("\*\*\* 3.更改专业 \*\*\*\n");

printf("\*\*\* 4.更改辅导员 \*\*\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\n 请输入选项:\n");

scanf("%d",&q);

switch(q)

{

case 1:

printf("请输入改变后的姓名:\n");

scanf("%s",ss.name);

strcpy(st[j].name,ss.name);

system("cls");

printf("\n 更改信息成功,返回主界面!\n");

main\_interface();

break;

case 2:

printf("请输入改变后的班级:\n");

scanf("%s",ss.class\_name);

strcpy(st[j].class\_name,ss.class\_name);

system("cls");

printf("\n 更改信息成功,返回主界面!\n");

main\_interface();

break;

case 3:

printf("请输入改变后的专业:\n");

scanf("%s",ss.major);

strcpy(st[j].major,ss.major);

system("cls");

printf("\n 更改信息成功,返回主界面!\n");

main\_interface();

break;

case 4:

printf("请输入改变后的辅导员:\n");

scanf("%s",ss.instructor);

strcpy(st[j].instructor,ss.instructor);

system("cls");

printf("\n 更改信息成功,返回主界面!\n");

main\_interface();

break;

default :

system("cls");

printf("选项输入错误! 返回上一界面!\n");

interface6();

break;

}

}

}

void start()

{

int j=103;

int i=0;

dm[0].num=101;

dm[0].Health\_score=10;

dm[0].member1[0]=NULL;

strcat(dm[0].member1,"赵钱");

dm[0].member2[0]=NULL;

dm[0].member3[0]=NULL;

dm[0].member4[0]=NULL;

dm[0].members\_num=panduansusherenshu(dm[0]);

dm[1].num=102;

dm[1].Health\_score=9;

dm[1].member1[0]=NULL;

strcat(dm[1].member1,"周吴");

dm[1].member2[0]=NULL;

dm[1].member3[0]=NULL;

dm[1].member4[0]=NULL;

dm[1].members\_num=panduansusherenshu(dm[1]);

for(i=2;i<100;i++)

{

dm[i].num=j++;

dm[i].members\_num=0;

dm[i].Health\_score=0;

dm[i].member1[0]=NULL;

dm[i].member2[0]=NULL;

dm[i].member3[0]=NULL;

dm[i].member4[0]=NULL;

}

st[0].num=201011011;

st[0].name[0]=NULL;

strcat(st[0].name,"赵钱");

st[0].dorm\_num=101;

st[0].class\_name[0]=NULL;

strcat(st[0].class\_name,"工商 1001");

st[0].gender[0]=NULL;

strcat(st[0].gender,"女");

st[0].major[0]=NULL;

strcat(st[0].major,"工商管理");

st[0].instructor[0]=NULL;

strcat(st[0].instructor,"巴菲特");

st[1].num=201011012;

st[1].name[0]=NULL;

strcat(st[1].name,"周吴");

st[1].dorm\_num=102;

st[1].class\_name[0]=NULL;

strcat(st[1].class\_name,"软件 1001");

st[1].gender[0]=NULL;

strcat(st[1].gender,"女");

st[1].major[0]=NULL;

strcat(st[1].major,"软件工程");

st[1].instructor[0]=NULL;

strcat(st[1].instructor,"李开复");

for(i=2;i<400;i++)

{

st[i].num=0;

st[i].name[0]=NULL;

st[i].class\_name[0]=NULL;

st[i].dorm\_num=0;

st[i].gender[0]=NULL;

st[i].instructor[0]=NULL;

st[i].major[0]=NULL;

}

}

int main()

{

start();

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\

n");

printf(" 宿舍管理系统说明: \n\n");

printf(" 该程序现有两人（201011011 赵钱、 201011012 周吴） 分别入住在寝室 101、

102\n\n");

printf(" 宿舍号必须为三位数字,如 101;学生学号必须为 9 位数字,如 201093046;\n\n");

printf(" 添加入住学生时， 学生性别只能输入'男'或'女',其他输入视为无效输入!\n\n");

printf(" 寝室最多入住人数为 4 人,添加学生到一宿舍时,该宿舍人数为 4 人,则提醒错

误!\n\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\n\n");

main\_interface();

return 0;

}