

C语言阶段

一、流程结构

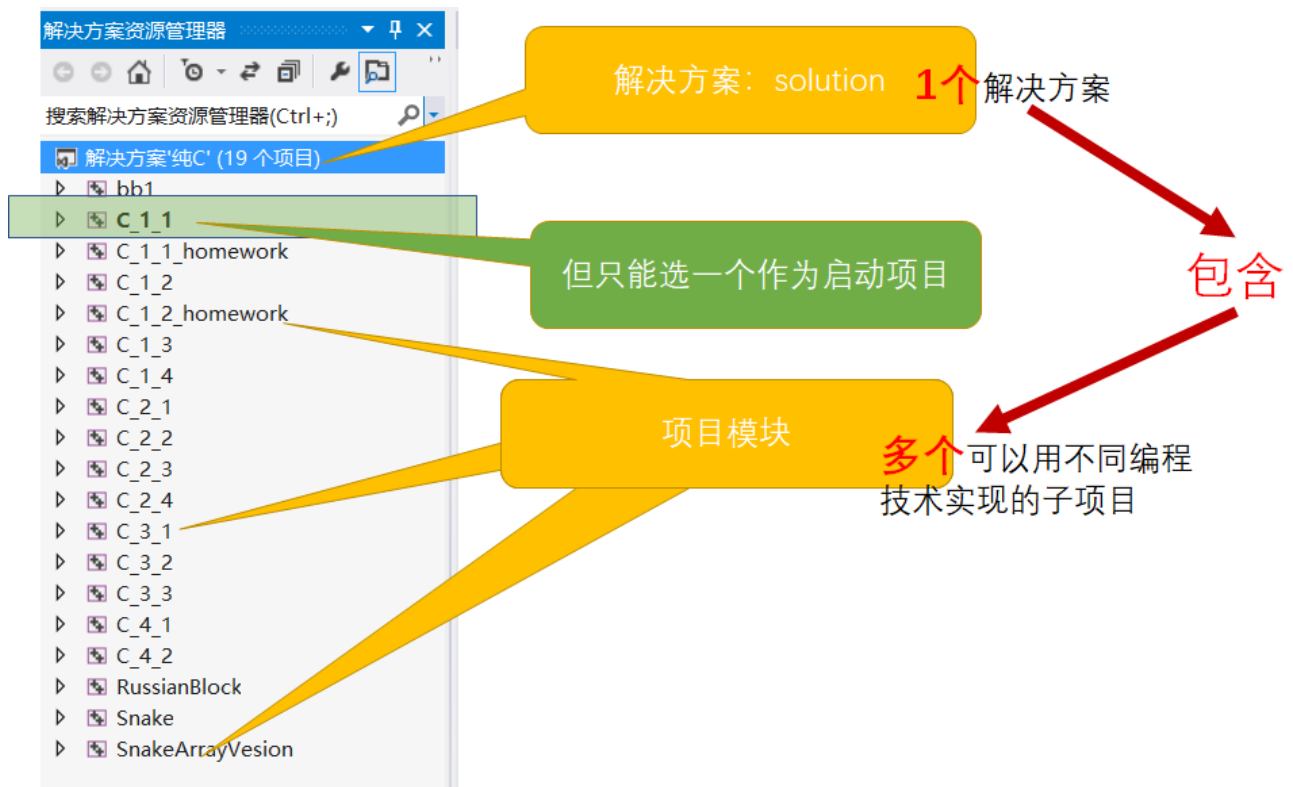
1. 初步掌握编程环境VS2019或更高版本
2. 敏锐识别流程结构：顺序·选择·循环
3. 初步掌握debug调试（先会看·慢慢干）

理解流程结构的划分与特性



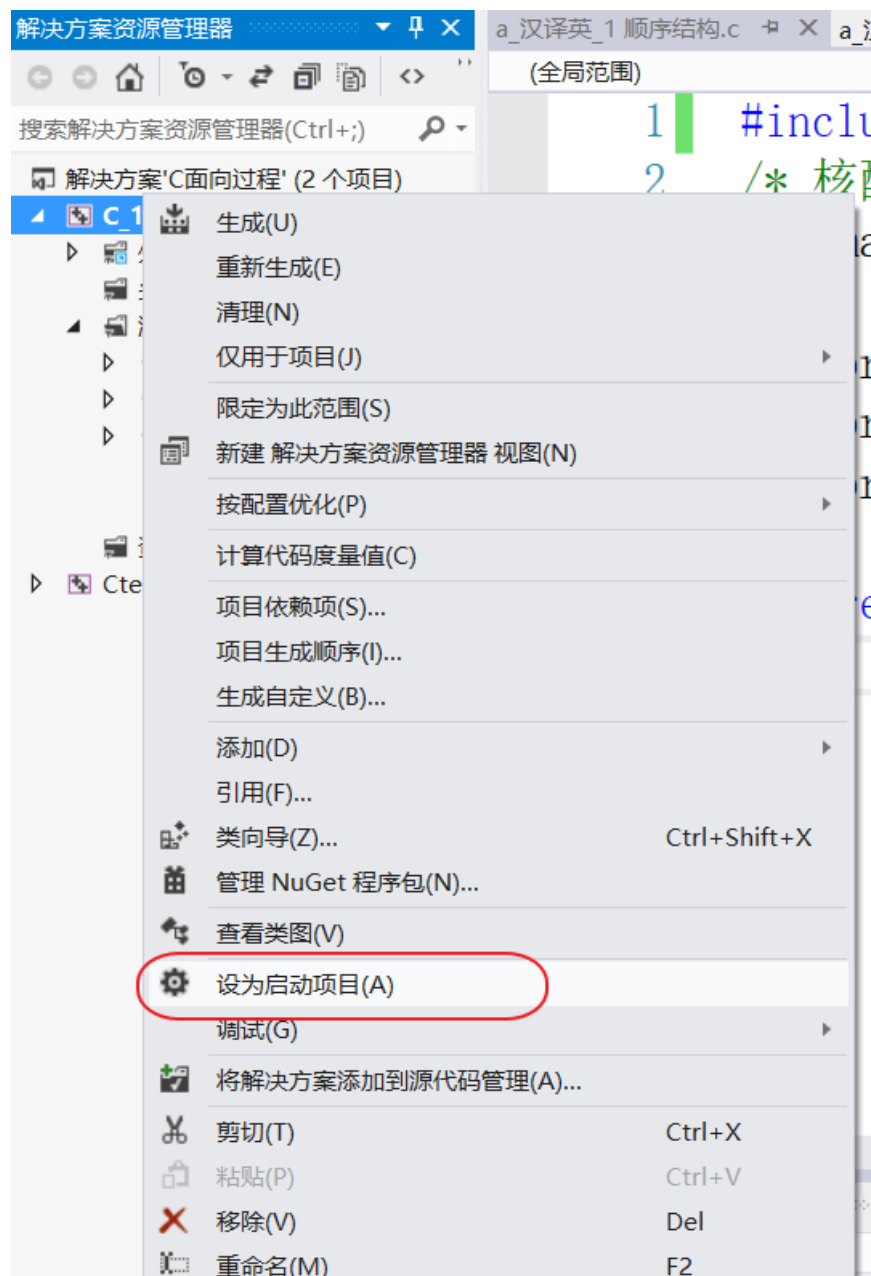
Visual Studio (VS) 重点操作

VS的项目管理方式.

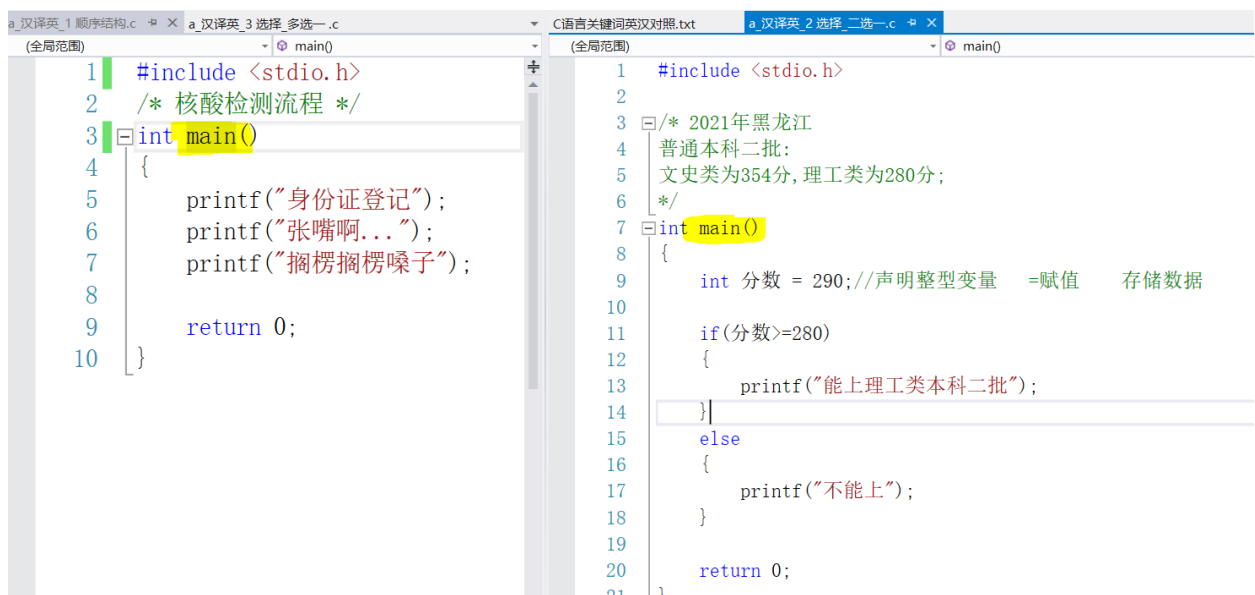


可运行的要求

1. 在解决方案中某个项目要被运行必须设置成: “启动项目”



2. 每个项目有且只能有一个main函数



上图两个main函数在编译时产生错误。解决方法：1. 从项目中把一个含main函数的源文件移除掉，2.将其中一个main函数改名字 如main01，比如：

```

1  #include <stdio.h>
2  /* 核酸检测流程 */
3  int main01()
4  {
5      printf("身份证登记");
6      printf("张嘴啊...");
7      printf("搁楞搁楞嗓子");
8
9      return 0;
10 }

```

理解并记住源代码中每个关键词的基本作用

标准的C代码结构：

```

#include <stdio.h> //必须包含标准输入输出头文件
/* 核酸检测流程 */
int main()
{

    return 0;
}

```

英文代码	汉字解释
#	预处理的标志
include	包含
stdio.h	标准stander输入input输出output头文件head(*.h)
main	主要的函数（主函数） 程序的执行入口，一个项目只能一个
int	整型
float	实型/浮点型/小数型
char	字符型
return	返回
printf	格式化format打印print
if	如果
else	否则

C语言第一单元关键词

#	预处理的标志
include	包含
stdio.h	标准stander输入input输出output头文件head(*.h)
main	主要的函数（主函数） 程序的执行入口，一个项目只能一个主函数
int	整型
float	实型/浮点型/小数型
char	字符型
return	返回
printf	格式化format打印print
if	如果
else	否则
switch	分支
case	当.....情况时
break	打断流程
default	默认情况
for	对于....开始循环
while	每当....开启循环
do	先做....再 开启循环

调试代码

目的：掌握调试方法，利用调试过程理解流程结构代码的执行原理

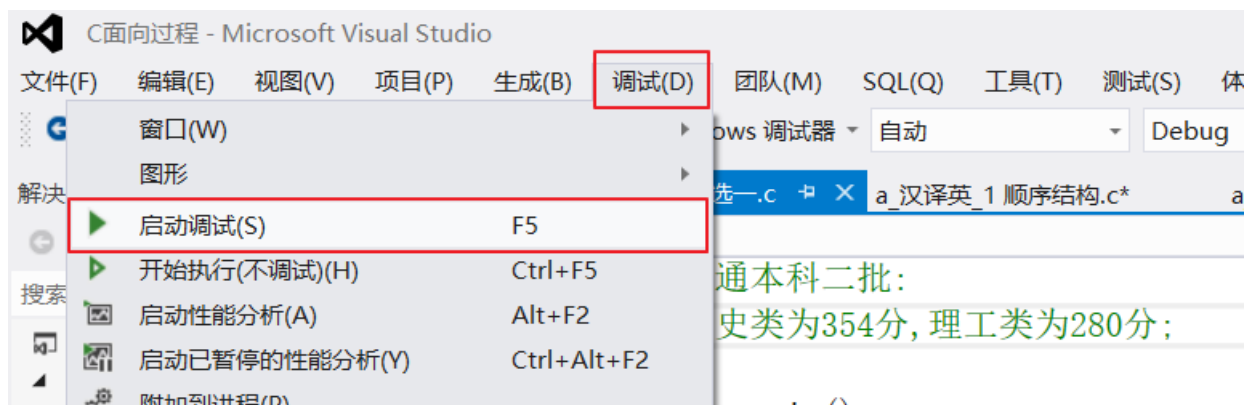
一、设置断点：所谓断点就是程序运行时的暂停点，执行到此位置时就停止，等待程序员的操作。



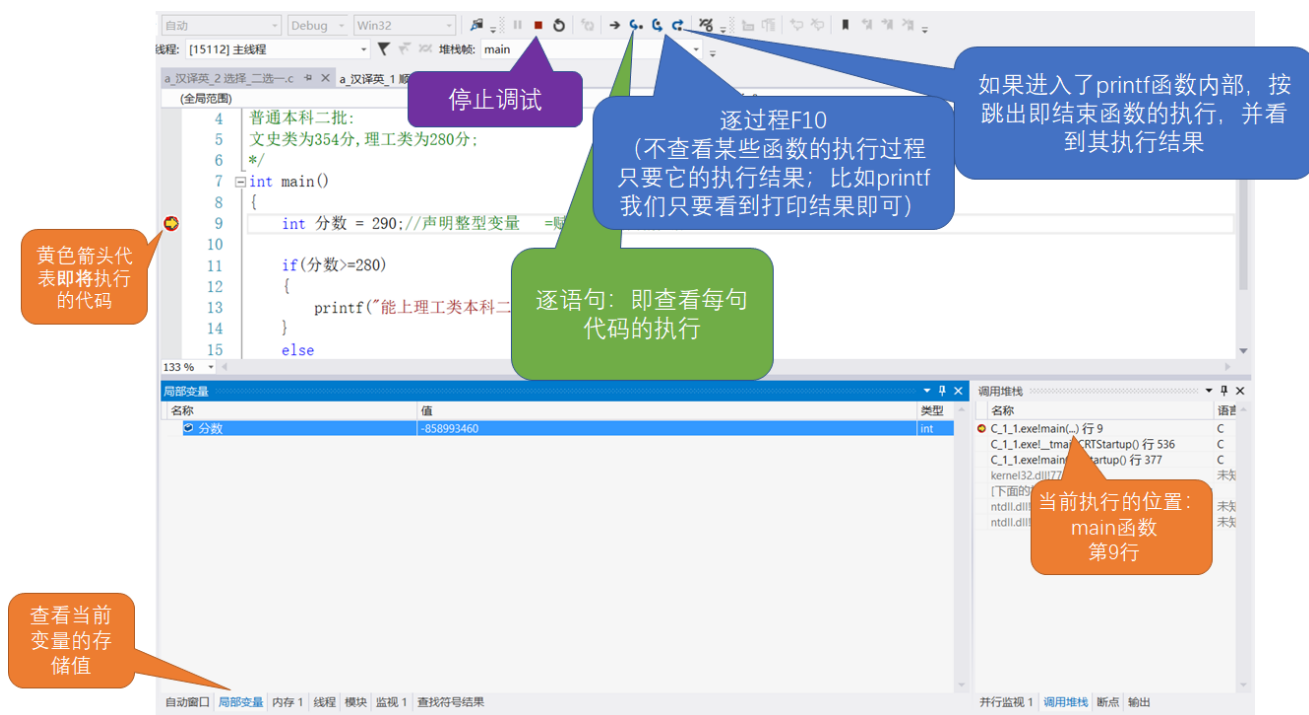
本单元：断点务必设置在main函数的第一条语句上（即从第一句开始理解程序的执行过程）

未来：断点设置在程序可能的bug位置。一个源代码文件可以设置多个断点。

二、调试运行：快捷键F5 启动调试



三、了解调试相关的按钮功能及观察的窗口



附件(一)：流程结构关键代码的模板

基本语句：

```
#include<stdio.h>
int main()
{
    ; //单独分号 代表一个空语句， ;也是单条语句的结束标志

    {

    } //一对{}称为复合语句。代表把几句单挑语句集合在一起，形成一组语句
    //从语法上可以把它看成 一条语句

    return 0;
}
```

条件语句（选择结构）

if语句（二选一结构）

```
#include<stdio.h>
int main()
{
    //方式一、
    if ( 条件表达式 )
        ; //if管理单条语句

    //方式二、
    if ( 条件表达式 )
    {

    } //if管理复合语句（多条语句）

    //方式三、
    if ( 条件表达式 )
    {
        //True 条件真对应的功能
    }
    else
    {
        //False 条件假对应的功能
    }

    return 0;
}
```

switch 多分支语句（多选一结构）

```
#include<stdio.h>
int main()
{
    //方式一、
    switch (整数结果型表达式)
    {
        case 1: //1 代表可能的计算结果
            break; //break 代表结束switch多选一结构
        case 2:
            break;
        case 3:
            break;
        default://default 默认入口 即前面没有满足时 执行次入口
            break;//此处的break可以省略 因为后面没有语句了
    }

    //方式二、
    switch (整数结果型表达式)
    {
        case 1: //1 代表可能的计算结果

        case 2:
            //没有break 则从满足入口开始依次向下执行指导遇到一个break为止
        case 3:
            break;
        default://default 默认入口 即前面没有满足时 执行次入口
            break;//此处的break可以省略 因为后面没有语句了
    }
```

```
    }  
  
    return 0;  
}
```

循环语句

for语句（知：起止）明确执行次数的循环

```
#include<stdio.h>  
int main()  
{  
    int i ;//定义循环变量  
    for(循环变量初始化 ； 循环条件 ； 循环变量变化规律 )  
    {  
        循环体  
    }  
    return 0;  
}
```

do while (≥ 1 次) 至少执行一次，再根据判断条件继续执行的循环

```
#include<stdio.h>  
int main()  
{  
  
    do  
    {  
        循环体  
    } while (循环条件);  
  
    return 0;  
}
```

while (≥ 0) 每当型循环，先判断条件，再决定是否进入循环体

```
#include<stdio.h>  
int main()  
{  
    while (循环条件)  
    {  
        循环体  
    }  
  
    return 0;  
}
```


附件（二）VS2019常用快捷键：

较常用快捷键

Ctrl + E + D = 代码格式化1

Ctrl + K + F = 代码快速格式化2

Ctrl + E + C = 注释代码1

Ctrl + K + C = 注释代码2

Ctrl + E + U = 取消注释代码1

Ctrl + K + U = 取消注释代码2

Tab + Tab = 代码联想

Alt + Enter = 快速引用

Ctrl + Shift + C 新建类

Ctrl + E + E = 在交互窗口中运行选中代码(很实用)

F6 = 生成(整个解决方案中类库)

Shift + F6 = 生成(当前类库)

Ctrl + R + G = 删除无效Using

Ctrl + H = 批量替换

Ctrl + G = 跳转到指定行

Shift + F9 = 监控(代码运行时)

Shift + F12 = 查找所有引用

Ctor + Tab + Tab = 生成无参构造器

项目相关的快捷键

Ctrl + Shift + B = 生成项目

Ctrl + Alt + L = 显示 Solution Explorer (解决方案资源管理器)

Shift + Alt+ C = 添加新类

Shift + Alt + A = 添加新项目到项目

编辑相关的键盘快捷键

Ctrl + Enter = 在当前行插入空行

Ctrl + Shift + Enter = 在当前行下方插入空行

Ctrl + 空格键 = 使用 IntelliSense (智能感知) 自动完成

Alt + Shift + 箭头键(←,↑,↓,→) = 选择代码的自定义部分

Ctrl + } = 匹配大括号、括号

Ctrl + Shift +} = 在匹配的括号、括号内选择文本

Ctrl + Shift + S = 保存所有文件和项目

Ctrl + K, Ctrl + C = 注释选定行

Ctrl + K, Ctrl + U = 取消选定行的注释

Ctrl + K, Ctrl + D = 正确对齐所有代码

Shift + End = 从头到尾选择整行

Shift + Home = 从尾到头选择整行

Ctrl + Delete = 删除光标右侧的所有字

导航相关的键盘快捷键

Ctrl + Up/Down = 滚动窗口但不移动光标

Ctrl + - = 让光标移动到它先前的位置

Ctrl ++ = 让光标移动到下一个位置

F12 = 转到定义

调试相关的键盘快捷键

Ctrl + Alt + P = 附加到进程

F10 = 调试单步执行

F5 = 开始调试

Shift + F5 = 停止调试

Ctrl + Alt + Q = 添加快捷匹配

F9 = 设置或删除断点

搜索相关的键盘快捷键

Ctrl + K Ctrl + K = 将当前行添加书签

Ctrl + K Ctrl + N = 导航至下一个书签

Ctrl + . = 如果你键入一个类名如 **Collection**，且命名空间导入不正确的话，那么这个快捷方式组合将自动插入导入

Ctrl + Shift + F = 在文件中查找

Shift + F12 = 查找所有引用

Ctrl + F = 显示查找对话框

Ctrl + H = 显示替换对话框

Ctrl + G = 跳转到行号或行

Ctrl + Shift + F = 查找所选条目在整个解决方案中的引用

VS常用快捷键搜集

以下内容来自网络，没有进行一一验证，不同版本VS肯定会有所差别，最常用的也就是打开窗口，代码编辑及调试类的快捷键。

窗口操作快捷键

Ctrl+W,W: 浏览器窗口

Ctrl+W,S: 解决方案管理器

Ctrl+W,C: 类视图

Ctrl+W,E: 错误列表

Ctrl+W,O: 输出视图

Ctrl+W,P: 属性窗口

Ctrl+W,T: 任务列表

Ctrl+W,X: 工具箱

Ctrl+W,B: 书签窗口

Ctrl+W,U: 文档大纲

Ctrl+D,B: 断点窗口

Ctrl+D,I: 即时窗口

Ctrl+Tab: 活动窗体切换

Ctrl+Shift+N: 新建项目

Ctrl+Shift+O: 打开项目

Ctrl+Shift+S: 全部保存

Shift+Alt+C: 新建类

Ctrl+Shift+A: 新建项

项目快捷键

CTRL + SHIFT + B生成解决方案

CTRL + F7 生成编译

CTRL + O 打开文件

CTRL + SHIFT + O打开项目

CTRL + SHIFT + C显示类视图窗口

F4 显示属性窗口

SHIFT + F4显示项目属性窗口

CTRL + SHIFT + E显示资源视图

F12 转到定义

CTRL + F12转到声明

CTRL + ALT + J对象浏览

CTRL + ALT + F1帮助目录

CTRL + F1 动态帮助

F1 帮助

SHIFT + F1当前窗口帮助

CTRL + ALT + F3帮助-搜索

SHIFT + ALT + ENTER全屏显示

CTRL + -向后定位

CTRL + SHIFT + -向前定位

CTRL + F4关闭文档窗口

CTRL + PAGE DOWN光标定位到窗口上方

CTRL + PAGE UP光标定位到窗口下方

CTRL + F6

CTRL + TAB下一个文档窗口

CTRL + SHIFT + F6

CTRL + SHIFT + TAB上一个文档窗口

ALT + F6下一个面板窗口

CTRL + K, CTRL + L取消remark

CTRL + K, CTRL + C注释选择的代码

CTRL + K, CTRL + U取消对选择代码的注释

CTRL + M, CTRL + O折叠代码定义

CTRL + M, CTRL + L展开代码定义

CTRL + DELETE删除至词尾

CTRL + BACKSPACE删除至词头

SHIFT + TAB取消制表符

CTRL + U转小写

CTRL + SHIFT + U转大写

CTRL + SHIFT + END选择至文档末尾

CTRL + SHIFT + HOME选择至文档末尾开始

SHIFT + END选择至行尾

SHIFT + HOME选择至行开始处

SHIFT + ALT + END垂直选择到最后尾

SHIFT + ALT + HOME垂直选择到最前面

CTRL + A全选

CTRL + W选择当前单词

CTRL + SHIFT + PAGE UP选择至本页前面

CTRL + SHIFT + PAGE DOWN选择至本页后面

CTRL + END文档定位到最后

CTRL + HOME文档定位到最前

CTRL + G转到...

CTRL + K, CTRL + P上一个标签

CTRL + K, CTRL + N下一个标签

ALT + F10调试-ApplyCodeChanges

CTRL + ALT+ Break停止调试

CTRL + SHIFT + F9 取消所有断点

CTRL + F9允许中断

CTRL + SHIFT + F5调试-重新开始

F5运行调试

CTRL + F5运行不调试

F10跨过程序执行

F11单步逐句执行

CTRL + J列出成员

CTRL + PAGE DOWN下一个视图

CTRL + B格式-粗体

CTRL + SHIFT + T格式-文字缩进

调试快捷键

F6: 生成解决方案

Ctrl+F6: 生成当前项目

F7: 查看代码

Shift+F7: 查看窗体设计器

F5: 启动调试

Ctrl+F5: 开始执行(不调试)

Shift+F5: 停止调试

Ctrl+Shift+F5: 重启调试

F9: 切换断点

Ctrl+F9: 启用/停止断点

Ctrl+Shift+F9: 删除全部断点

F10: 逐过程

Ctrl+F10: 运行到光标处

F11: 逐语句

编辑快捷键

Ctrl+E,D ----格式化全部代码

Ctrl+E,F ----格式化选中的代码

Shift+Alt+Enter: 切换全屏编辑

Ctrl+B,T / Ctrl+K,K: 切换书签开关

Ctrl+B,N / Ctrl+K,N: 移动到下一书签

Ctrl+B,P: 移动到上一书签

Ctrl+B,C: 清除全部标签

Ctrl+I: 渐进式搜索

Ctrl+Shift+I: 反向渐进式搜索

Ctrl+F: 查找

Ctrl+Shift+F: 在文件中查找

F3: 查找下一个

Shift+F3: 查找上一个

Ctrl+H: 替换

Ctrl+Shift+H: 在文件中替换

Alt+F12: 查找符号(列出所有查找结果)

Ctrl+Shift+V: 剪贴板循环

Ctrl+左右箭头键: 一次可以移动一个单

词

Ctrl+上下箭头键: 滚动代码屏幕, 但不移动光标位置。

Ctrl+Shift+L: 删除当前行

Ctrl+M,M: 隐藏或展开当前嵌套的折叠状态

Ctrl+M,L: 将所有过程设置为相同的隐藏或展开状态

Ctrl+M,P: 停止大纲显示

Ctrl+E,S: 查看空白

Ctrl+E,W: 自动换行

Ctrl+G: 转到指定行

Shift+Alt+箭头键: 选择矩形文本

Alt+鼠标左按钮: 选择矩形文本

Ctrl+Shift+U: 全部变为大写

Ctrl+U: 全部变为小写

代码快捷键

Ctrl+J / Ctrl+K,L: 列出成员

Ctrl+Shift+空格键 / Ctrl+K,P: 参数信息

Ctrl+K,I: 快速信息

Ctrl+E,C / Ctrl+K,C: 注释选定内容

Ctrl+E,U / Ctrl+K,U: 取消选定注释内容

Ctrl+K,M: 生成方法存根

Ctrl+K,X: 插入代码段

Ctrl+K,S: 插入外侧代码

F12: 转到所调用过程或变量的定义

单元测试

Ctrl + R , A : 执行所有测试

Ctrl + R , T : 执行当前所在测试方法

Ctrl + R , D: 执行最后一次测试的所有测试

Ctrl + R , F: 执行最后一次测试的所有测试的失败测试