





Debug概述

Debug: 是供程序员使用的程序调试工具,它可以用于查看程序的执行流程,也可以用于追踪程序执行过程来调试程序。





Debug操作流程

Debug调试,又被称为断点调试,断点其实是一个标记,告诉Debug从标记的地方开始查看

- ① 如何加断点
- ② 如何运行加了断点的程序
- ③ 看哪里
- ④ 点哪里
- ⑤ 如何删除断点





如何加断点

选择要设置断点的代码行,在行号的区域后面单击鼠标左键即可。

```
public class DebugDemo {
            public static void main(String[] args) {
13
                //定义两个变量
14
                int i = 10;
                int j = 20;
16
                //求和
18
                int sum = i + j;
19
20
                //输出结果
                System. out. println("sum:" + sum);
23
```





如何运行加了断点的程序

在代码区域右键Debug执行

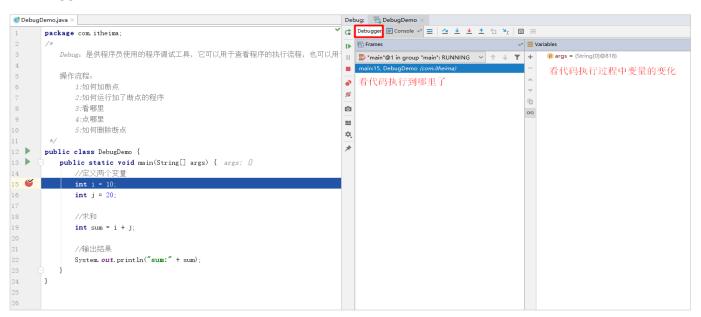
```
public class DebugDemo {
              public static void main(String[] args)
13
                                                                                           Copy Reference
                                                                                                                     Ctrl+Alt+Shift+C
                  //定义两个变量
14
                                                                                        Paste
                                                                                                                             Ctrl+V
15
                  int i = 10;
                                                                                                                        Ctrl+Shift+V
                                                                                           Paste from History...
                                                                                                                     Ctrl+Alt+Shift+V
                   int j = 20:
                                                                                           Paste Simple
16
                                                                                           Column Selection Mode
                                                                                                                     Alt+Shift+Insert
                                                                                           Find Usages
                                                                                                                             Alt+F7
                  //求和
18
                                                                                           Refactor
                   int sum = i + j;
19
                                                                                           Folding
                                                                                           Analyze
                  //输出结果
                                                                                           Go To
                  System. out. println("sum:" + sum);
                                                                                           Generate...
                                                                                                                          Alt+Insert
                                                                                           Recompile 'DebugDemo.java'
                                                                                                                       Ctrl+Shift+F9
24
                                                                                        Run 'DebugDemo.main()'
                                                                                                                      Ctrl+Shift+F10
                                                                                           Debug 'DebugDemo.main()'
                                                                                        Run 'DebugDemo.main()' with Coverage
26
                                                                                        ave 'DebugDemo.main()'
```





看哪里

看Debugger窗口







看哪里

看Console窗口

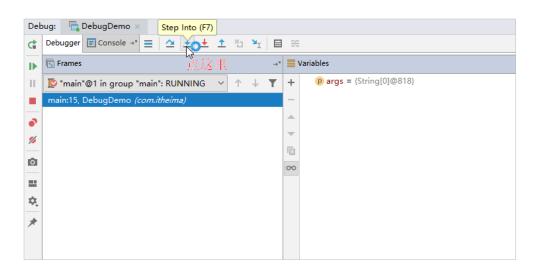
```
DebugDemo
                                                                     ✓ 😝 Debugger 🗉 Console → 😑 🔼 ± ± ± ± ½ ⅓ ⅓ 🗎 🕾
       package com. itheima;
                                                                              D:\develop\Java\jdk-11\bin\java.exe -agentlib:jdwp=transport=dt_socket,address=127.0.
          Debug: 是供程序员使用的程序调试工具,它可以用于查看程序的执行流程,也可以用:
                                                                              Connected to the target VM, address: '127.0.0.1:52039', transport: 'socket'
                                                                       操作流程:
                                                                              看程序执行过程中的结果展示
             1:如何加断点
             2:如何运行加了断点的程序
             3:看哪里
             4:点哪里
9
             5:如何删除断点
12
       public class DebugDemo {
13
          public static void main(String[] args) { args: 0
             //定义两个变量
14
             int i = 10;
             int j = 20;
             //求和
             int sum = i + j;
             //输出结果
             System. out. println("sum:" + sum):
24
```





点哪里

点Step Into (F7)这个箭头,也可以直接按F7

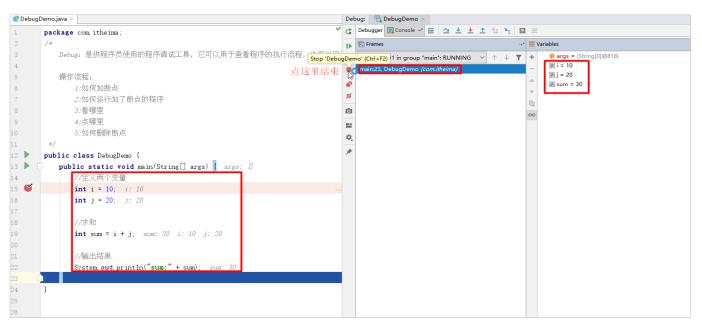






点哪里

点Stop结束







如何删除断点

选择要删除的断点,单击鼠标左键即可。

```
public class DebugDemo {

public static void main(String[] args) { args: 0

//定义两个变量
    int i = 10; i: 10

int j = 20; j: 20

//求和
    int sum = i + j; sum: 30 i: 10 j: 20

//输出结果
System. out. println("sum:" + sum); sum: 30

30

31

32

32

33

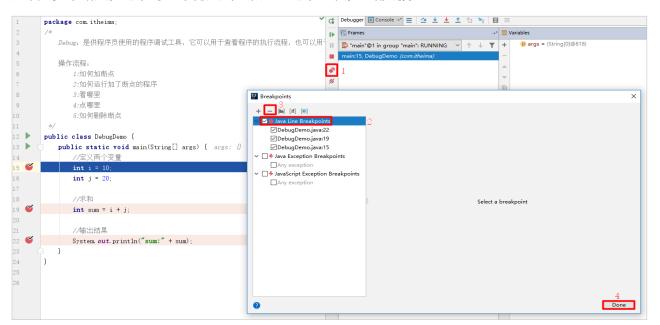
34
```





如何删除断点

如果是多个断点,可以每一个再点击一次。也可以一次性全部删除。







Debug使用

查看循环求偶数和的执行流程





Debug使用

● 查看方法调用的执行流程

注意事项:

如果数据来自于键盘输入,一定要记住输入数据,不然就不能继续往下查看了。



传智播客旗下高端IT教育品牌