

## 006.Java程序对子进程进行读写

使用Runtime对象的exec()方法可以运行平台上的其他程序，该方法产生一个Process对象，

Process对象代表由该Java程序启动的子进程。

Process类提供了3个方法，用于让程序和其子进程通信

InputStream getErrorStream(): 获取子进程的错误流。

InputStream getInputStream(): 获取子进程的输入流。

OutputStream getOutputStream(): 获取子进程的输出流。

### 范例：读取其他进程的输出信息

```
public class ReadFromProcess {
    public static void main(String[] args) throws IOException {
        Process p = Runtime.getRuntime().exec("javac");
        //try() {}jdk7新语法, 关闭()中的流, 效果类型在finally中br.close()
        try (BufferedReader br = new BufferedReader(new
InputStreamReader(p.getErrorStream()))){
            String buff = null;
            while((buff = br.readLine()) != null){
                System.out.println(buff);
            }
        }
    }
}
```

### 范例：写信息到其他进程，这个程序要在命令里执行

```
public class FileTest1
{
    public static void main(String[] args)
    {
        PrintStream ps = null;
        try
        {
            //运行java ReadStandard命令, 返回运行该命令的子进程
            Process p = Runtime.getRuntime().exec("java ReadStandard");
            //以p进程的输出流创建PrintStream对象
            //这个输出流对本程序是输出流, 对p进程则是输入流)
            ps = new PrintStream(p.getOutputStream());
            //向ReadStandard程序写入内容, 这些内容将被ReadStandard读取
            ps.println("普通字符串");
            ps.println(new FileTest1());
        }
        catch (IOException ex)
        {
            ex.printStackTrace();
        }
    }
}
```

```

        finally
        {
            if (ps != null)
                ps.close();
        }
    }
}

//定义一个ReadStandard类，该类可以接受标准输入，
//并将标准输入写入out.txt文件。
class ReadStandard
{
    public static void main(String[] args) throws Exception
    {
        //使用System.in创建Scanner对象，用于获取标准输入
        Scanner sc = new Scanner(System.in);
        PrintStream ps = new PrintStream(
            new FileOutputStream("D:\\out.txt"));
        //增加下面一行将只把回车作为分隔符
        sc.useDelimiter("\n");
        //判断是否还有下一个输入项
        while (sc.hasNext())
        {
            //输出输入项
            ps.println("键盘输入的内容是：" + sc.next());
        }
        ps.close();
    }
}

```