# 第一部分：Java 语言核心

## Duplicate entry：字段值重复，入库失败

【关键字：Duplicate entry】【难度：初级】【掌握程度：掌握】

解决方案：删除字段重新添加

## Unknown column 字段不存在

【关键字：Unknown column】【难度：初级】【掌握程度：掌握】

解决方案：1，字段有可能真的不存在。2，没有添加自增，在插入的时候没有添加id字段导致

## No Database Selected： 没有数据库选择

【关键字：No Database Selected】【难度：初级】【掌握程度：掌握】

解决方案：没有好的解决办法：只能重新启动数据库

## Can't create table '.\cc\tb\_collect.frm'

【关键字：Can't create table】【难度：初级】【掌握程度：掌握】

解决方案：外键字段大小等不一致影响

## [Err] 1064 - You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near

【关键字：You have an error in your SQL syntax】【难度：初级】【掌握程度：掌握】

解决方案：

学员更改表语法错误：alter table 需加外键的表 add constraint 外键名 foreign key(需加外键表的字段名) referencnes 关联表名(关联字段名);

## substring (firstname,1,1),报错inavilid use of group function

【关键字：inavilid use of group function】【难度：初级】【掌握程度：掌握】

解决方案：函数后面不能有空格

## where salary >avg(salary) from employees 报inavilid use of group function

【关键字：inavilid use of group function】【难度：初级】【掌握程度：死背】

解决方案：where 后面不能跟聚合函数。

## Invalid use of group function 无效使用分组函数

【关键字：Invalid use of group function】【难度：初级】【掌握程度：掌握】

解决方案：分组函数使用不正确

## Column 'job\_id' in group statement is ambiguous

【关键字：statement is ambiguous】【难度：初级】【掌握程度：了解】

解决方案：group by后面的跟的job\_id 不明确

## mysql -uroot -p 库名<路径 这种导入方式会报错

【关键字：SQL导入报错】【难度：初级】【掌握程度：掌握】

解决方案：库名必须与sql文件中的库名一致

## 报错：subquery returns more rows than 1

【关键字：subquery returns more rows than 1】【难度：初级】【掌握程度：掌握】

解决方案：=后面的子查询返回了多行记录，把=改为in

## Unknown column 'column\_name' in 'field list'

【关键字：Unknown colum】【难度：初级】【掌握程度：掌握】

解决方案：列名写错

## 出现Can't connect to MySQL server on 'localhost' (10061)

【关键字：Can't connect to MySQL】【难度：初级】【掌握程度：掌握】

解决方案：问题：不能连接到 localhost 上的mysql

分析：这说明“localhost”计算机是存在的，但在这台机器上却没提供MySQL服务。

需要启动这台机器上的MySQL服务,如果机子负载太高没空相应请求也会产生这个错误。

解决:既然没有启动那就去启动这台机子的mysql。如果启动不成功，多数是因为my.ini配置的有问题。重新配置其即可。

如果觉得mysql负载异常，可以到mysql/bin 的目录下执行mysqladmin -uroot -p123 processlist来查看mysql当前的进程。

## Unknown MySQL Server Host 'localhosadst' (11001)

【关键字：Unknown MySQL Server】【难度：初级】【掌握程度：掌握】

解决方案：

分析：服务器 localhosasdst 不存在。或者根本无法连接

解决：仔细检查 ./config.inc.php 找到$dbhost重新设置为正确的mysql 服务器地址。

## Can't open file: 'xxx\_forums.MYI'. (errno: 145)

【关键字：Can't open file】【难度：中级】【掌握程度：了解】

解决方案：

问题：不能打开xxx\_forums.MYI

问题分析：

这种情况是不能打开 cdb\_forums.MYI 造成的，引起这种情况可能的原因有：

1、服务器非正常关机，数据库所在空间已满，或一些其它未知的原因，对数据库表造成了损坏。

2、linux 操作系统下直接将数据库文件拷贝移动会因为文件的属组问题而产生这个错误。

解决方法：

1、修复数据表

可以使用下面的两种方式修复数据表：（第一种方法仅适合独立主机用户）

1）使用 myisamchk ，MySQL 自带了专门用户数据表检查和修复的工具 —— myisamchk 。更改当前目录到 MySQL/bin 下面，一般情况下只有在这个下面才能运行 myisamchk 命令。常用的修复命令为：myisamchk -r 数据文件目录/数据表名.MYI；

2）通过管理员修复，管理员带有修复数据表的功能，进入到某一个表中后，点击“操作”，在下方的“表维护”中点击“修复表”即可。

注意：以上两种修复方式在执行前一定要备份数据库。

2、修改文件的属组（仅适合独立主机用户）

1）复制数据库文件的过程中没有将数据库文件设置为 MySQL 运行的帐号可读写（一般适用于 Linux 和 FreeBSD 用户）。

## Error: Can't connect to local MySQL server through socket '/var/lib/mysql/mysql.sock' error.:2002

【关键字：Can't connect to local】【难度：初级】【掌握程度：了解】

解决方案：

问题分析：

出现这个错误一般情况下是因为下面两个原因：

1.MySQL 服务器没有开启。

2.MySQL 服务器开启了，但不能找到 socket 文件。

解决方法：

1.确认数据库是否正常启动。

2.检查一下 MySQL 服务是否已经开启，没有开启，请启动 MySQL 服务；如果已经开启，并且是 Linux 系统，请检查一下 MySQL 的 socket 的路径，然后打开 config.inc.php 找到

$dbhost = 'localhost'; 在 hostname 后面加冒号‘:’和 MySQL 的 socket 的路径。

比如 MySQL 服务器为 localhost

MySQL 的 socket 的路径为 /tmp/mysql.sock

那么就改成如下：

$dbhost = 'localhost:/temp/mysql.sock';

## Error: Client does not support authentication protocol requested by server; consider upgrading MySQL client error.:1251

【关键字：Client does not support】【难度：中级】【掌握程度：了解】

解决方案：

问题分析：

如果升级 MySQL 到 4.1 以上版本后遇到以上问题,请先确定你的 MySQL Client 是 4.1 或者更高版本(Windows 下有问题你就直接跳到下面看解决方法了，因为 MySQL 在 Windows 是 client 和 server 一起装上了的)。

解决方法：

1. Windows 平台

主要是改变连接 MySQL 的帐户的加密方式，MySQL 4.1/5.0 是通过 PASSWORD 这种方式加密的。可以通过以下两种方法得到解决：

1) mysql->SET PASSWORD FOR 'some\_user'@'some\_host'=OLD\_PASSWORD('new\_password');

2) mysql->UPDATE mysql.user SET Password=OLD\_PASSWORD('new\_password') WHERE Host='some\_host' AND User='some\_user';

2. Linux/Unix 平台

Linux 平台下首先确定是否安装过 MySQL 的客户端，这个用 rpm 安装很简单，Linux 代码为：

rpm -ivh MySQL-client-4.1.15-0.i386.rpm

然后在编译的时候要加上：

--with-mysql=/your/path/to/mysql

一般情况下都可以解决。如果还出现这种错误，可以按照下面的方法来做：

mysql->SET PASSWORD FOR 'some\_user'@'some\_host'=OLD\_PASSWORD('new\_password');

mysql->UPDATE mysql.user SET Password=OLD\_PASSWORD('new\_password') WHERE Host='some\_host' AND User='some\_user';

## Can't create a new thread; if you are not out of available memory, you can consult the manual for a possible OS-dependent bug。 error.:11/35

【关键字：Can't create a new thread】【难度：中级】【掌握程度：了解】

解决方案：

问题分析：

数据库服务器问题，数据库操作无法创建新线程。一般是两个原因：

1.服务器系统内存溢出。

2.环境软件损坏或系统损坏。

解决方法：

1.确认数据库服务器的内存和系统是否正常。

2.检查服务器的内存和系统是否正常，如果服务器内存紧张，请检查一下哪些进程消耗了服务器的内存，同时考虑是否增加服务器的内存来提高整个的负载能力。

## dropping database (can't delete '%s', errno: %d) error.:1009

【关键字：dropping database】【难度：中级】【掌握程度：了解】

解决方案：

问题分析：

不能删除数据库文件，导致删除数据库失败。

解决方法：

1.检查是否有权限删除数据。

2.检查数据库是否存在。

## Got error 28 from table handler error.:1030

【关键字：error.:1030】【难度：中级】【掌握程度：了解】

解决方案：

sql文件导入数据缺失，该学员的sql文件有问题，导入数据缺失，换了一份文件没啥问题。

## PROCEDURE hr.rand\_data does not exist 1305

【关键字：PROCEDURE】【难度：中级】【掌握程度：了解】

解决方案：缺少rand\_data 库

## 使用after触发器 ERROR 1362 (HY000): Updating of NEW row is not allowed in after trigger

【关键字：ERROR 1362】【难度：中级】【掌握程度：了解】

解决方案：修改 begin end 之间的代码，写完整的sql语句

## 1366-Incorrect string value:'\xE5\x8D\xA1\xE5......' for column 'filename' at row 1

【关键字：1366-Incorrect 】【难度：中级】【掌握程度：了解】

解决方案：乱码问题，更改数据库的编码，alter database 库名 character set utf8;

## 每个分号语句都会执行

【关键字：每个分好语句都会执行】【难度：中级】【掌握程度：了解】

解决方案：会话结束后，忘记设置 delimiter //

## 设置eclipse中的tab键为4个空格

【关键字：eclipse】【难度：初级】【掌握程度：了解】

解决方案：

1.点击 window->preference-,依次选择 General->Editors->Text Editors,选中右侧的 insert space for tabs;保存，第一步完成；

2.点击 window->preference-,依次选择 java（或C++）->code style ->formatter,点击右侧的editor，选则左侧 tab policy的值为spaces only,确定，应用保存即可;

注意若出现应用Apply按钮为灰色的情况，需要回到上一步，点击new按钮，根据当前的样式重新生成一个新的样式并保存，重复第2步，编辑该样式即可。

## javac' is not recognized as an internal or external command, operable program or batch file

【关键字：javac】【难度：初级】【掌握程度：了解】

解决方案：这个错误提示表明Windows找不到Java编译器（javac），所以要告知Windows改如何去找到javac这个编译工具。

在命令窗口中，可以使用绝对路径来访问javac编译器，如在命令行提示符下输入C:\jdk1.7.0\bin\javac HelloWorldApp.java

使用绝对路径会比较麻烦，因为每次调用javac和java工具时都必须输入它们的完整路径，所以通过更新环境变量PATH来实现。

在windows下为javac和java等JDK工具添加环境变量支持的步骤如下：

右击我的电脑-》属性-》高级系统设置-》环境变量

找到系统变量中的PATH（或者path）变量，点击编辑在变量值的最后面添加根据JDK的安装路径来添加，注意不要漏掉最前面的那个分号，且必须为英文输入法下的！！

环境变量设置成功后可以直接使用javac及java了，检查是否添加成功，我们可以在命令行下直接输入java及javac即可验证

## Exception in thread "main" java.lang.NoSuchMethodError: main

【关键字：NoSuchMethodError】【难度：初级】【掌握程度：了解】

解决方案：Java虚拟机（JVM）要求你所运行的类文件必须提供入口函数main，这个错误表明你所要启动的类文件缺失main方法，造成的原因可能是语法错误或者直接是缺少了main方法

## 递归调用方法进行了sc.close，第二次调用sc.nextInt();时报错：NoSuchElement

【关键字：NoSuchElement】【难度：初级】【掌握程度：了解】

Scanner sc = new Scanner(System.in); sc.nextInt(); sc.close();

解决方案：可以将sc.close()删除掉

## Int j=0; I%j==0; 报错：Exception:/ by zero

【关键字：exception:/ by zero】【难度：初级】【掌握程度：了解】

解决方案：不能对0取余 ,修改初始化值到1

## java.lang.ClassNotFoundException

【关键字：ClassNotFoundException】【难度：初级】【掌握程度：了解】

解决方案：这个异常是很多原本在开发环境中开发的程序，把程序包放在编译经常出现的问题，异常的解释是"指定的类不存在"，这里主要考虑一下类的名称和路径是否正确即可

## java.lang.illegalargumentexception

【关键字：illegalargumentexception】【难度：初级】【掌握程度：了解】

解决方案：这个异常的解释是"方法的参数错误"，很多j类库中的方法在一些情况下都会引发这样的错误，比如比如g.setcolor(intred,intgreen,intblue)这个方法中的三个值，如果有超过255的也会出现这个异常，因此一旦发现这个异常，我们要做的，就是赶紧去检查一下方法调用中的参数传递是不是出现了错误

## java.lang.StackOverflowError

【关键字：java.lang.StackOverflowError】【难度：中级】【掌握程度：了解】

解决方案：学生问：

public class digui1{

public static void main(String args[]){

System.out.println("计算1至10000的和为" + method(1));

}

public static int method(int n){

if (n == 10000) return 10000;

else return n + method(n + 1);

}

}

这是源代码，1加到1000没有问题能输出数据，但是1加到10000就会出错

回答：

函数调用，参数保存在当前线程的栈Stack中，而不是保存在堆Heap中。一般来说Java虚拟机默认的Heap大小是128M，用来存放new的对象。但是对于每一个线程，默认的栈大小只有512KB。所以，Java不能支持过多层次的递归调用。

要让这段程序可以运行，可以加上参数

Java -Xss32m digui1

来设置线程的Stack的大小。

至于为什么默认的栈的大小只有512KB, 那是因为在操作系统中，一个进程的最大虚拟内存是2GB（以32位架构为例），一个进程可以启动多个线程。每启动一个线程，操作系统都要分配一段内存做Stack，来保存这个线程函数调用的参数。也就是说，为了达到最大线程数和线程所用Stack的平衡，一个进程会设置它启动的线程的Stack的大小。在Java中，栈的大小是512KB，所以最大可以启动 2GB/512KB ，大约是2000个线程。超过2000个， 将会报错（OutOfMemory)

## java.lang.OutOfMemoryError:Java heap space

【关键字：java.lang.OutOfMemoryError】【难度：中级】【掌握程度：了解】

解决方案：Heap内存溢出，意味着Young和Old generation的内存不够。

调整java启动参数 -Xms -Xmx 来增加Heap内存。

## 34. 飞行物图片出现在窗体的左上角问题。如下图



【关键字：dramImage绘制】【难度：中级】【掌握程度：了解】

解决方案：将绘制的x,y设置成灵活的X, Y,不要写死应改为g,drawImage(getImage(),x,y,null);

## Java 使用内部类时候的容易产生的错误

【关键字：内部类】【难度：中级】【掌握程度：掌握】

解决方案：这是由于内部类定义产生的错误

Java要生成内部类对象时，必须要有外部类对象。内部类对象可以拿到外部类对象的所有成员与方法，是因为java在创建内部类的同时，一定要同时生成一个指向外部对象的引用，如果编译器找不到正引用就会报错。内部类对象的生成是必须与外部类对象绑定的，除非内部类是静态的。

【案例分析】：

public class InnerClass {  
  public static void main(String[] args) {  
    Client client  = new Client();  
  }  
  private class Client{ }  
}  
下面是正确的用法：  
public class InnerClass {  
  public static void main(String[] args) {  
    InnerClass.Client c = new InnerClass().new Client();  
  }  
  private class Client{ }  
}

## No enclosing instance of type Test1 is accessible. Must qualify the allocation with an enclosing instance of type Test1 (e.g. x.new A() where x is an instance of Test1)

【关键字：内部类】【难度：中级】【掌握程度：初级】

解决方案： 1、将static方法换成非static方法。

修改new内部类的对象的方法：cc[0]=p2.new Contents();或：cc[0]=p2.contents();? //contents()不是静态方法

## jdk环境变量都配对了，为什么还显示不成功

【关键字：jdk配置】【难度：初级】【掌握程度：初级】

解决方案：由于用的win10操作系统，压缩包跟文件夹很相识，结果学生没解压

## 运行程序出现NAN

【关键字：Math】【难度：初级】【掌握程度：初级】

解决方案：用Math.sqrt时没有进行边界值判断，出现类型错误

## 为什么会有StringIndexOutOfBoundsException这个异常?

【关键字：string异常】【难度：初级】【掌握程度：掌握】

解决方案：api底层调用只要不符合逻辑的都会报这异常，列如：split（-1,10）就会报错，底层源码就是这样写的 没有太多为什么。

## java开关详解

【关键字：开关的应用】【难度：初级】【掌握程度：掌握】

代码演示：

private static void sort(int[] a) {  
   for(int i=0;i<a.length;i++) {  
    //j循环把小值往前推，  
    //最终把最小值推到 i 位置  
    boolean flag=false;//表示没有交换  
    for(int j=a.length-1;j>i;j--) {  
     if(a[j]<a[j-1]) {  
      int c=a[j];  
      a[j] = a[j-1];  
      a[j-1] = c;  
      flag=true;  
     ｝}  
    //j递减时没有交换数据  
    //数据位置都正确，排序结束  
   /\* if(!flag) {  
     break;//中断i循环，跳出  
    }\*/  
    System.out.println(Arrays.toString(a));  
   }}  
不懂if(!flag) {  
   break;//中断i循环，跳出  
    }为什么代码中要加if(!flag)这个判断才能正常输出这个方法的含义

解决方案：if里面若为true,则说明flag为false,没有交换数据，说明不需要再继续进行比较，进入判断语句，breank中断循环。

## 41. java.lang.OutOfMemoryError: GC overhead limit exceeded

【关键字：GC垃圾回收】【难度：初级】【掌握程度：掌握】

解决方案：

在GC花费了大量时间，却仅回收了少量内存时，也会报出OutOfMemoryError，学生遇到这问题，

主要是为了实验内存中是如何布局的。  
 当使用-XX:+UseParallelGC或-XX:+UseConcMarkSweepGC收集器时，在上述情况下会报错，

【分析官方文档】：

这 是并行（并发）收集器将将抛出错误，如果太多时间浪费在垃圾收集：超过98%的总时间是花在垃圾收集和小于2%的堆进行回收，将抛出一个OutOfMemoryError。  
对这个问题，一是需要进行GC turning，二是需要优化程序逻辑。

## 42. java.lang.OutOfMemoryError:Requested array size exceeds VM limit

【关键字：最大长度数组】【难度：初级】【掌握程度：了解】

解决方案：

这个错误比较少见（试着new一个长度1亿的数组可能会发生这样的问题），是由于Heap空间不足。如果需要new一个如此之大的数组，程序逻辑多半是不合理的。  
解决：修改程序逻辑，或者也可以通过-Xmx来增大堆内存。

## 43. 子类继承父类，super（）报错？

【关键字：继承】【难度：中级】【掌握程度：掌握】

解决方案：父类有有参构造器，所以不会自动生成无参构造器，加了个无参构造器

## 44. 使用RandomAccessFile写入字符到文件中，能够读取到，可是文件中却没有东西。

【关键字：RandomAccessFile】【难度：中级】【掌握程度：掌握】

解决方案：因为raf是写入字节到文件中的，文件默认解码ascii解码的，文件中可能是空白，然后误以为没有写入到文件中

## 45. java.io.EOFException异常？

【关键字：RandomAccessFile文件】【难度：中级】【掌握程度：掌握】

解决方案：EOFException表示输入过程中意外地到达文件尾或流尾的信号,导致从session中获取数据失败。异常是tomcat本身的问题，由于tomcat上次非正常关闭时有一些活动session被持久化（表现为一些临时文件），在重启时，tomcat尝试去恢复这些session的持久化数据但又读取失败造成的。此异常不影响系统的使用。

## 46. User [id=2, name=fancq, pwd=3 读写案例中，后面的内容读不出来？

【关键字：课堂案例】【难度：中级】【掌握程度：掌握】

解决方案：

public static User readUser(RandomAccessFile raf,int row) throws Exception{  
 int start = row\*64;  
 raf.seek(start);  
 int id=raf.readInt();  
 byte[] bytes = new byte[22];  
 raf.read(bytes);//\*\*\*  
 String name=new String(bytes,0,20,"utf-8").trim();  
 raf.read(bytes);//\*\*\*  
 String pwd = new String(bytes, 0,18, "utf-8").trim();  
 raf.read(bytes);  
 String mobile = new String(bytes, "utf-8").trim();  
 return new User(id, name, pwd, mobile);  
 }

//\*\*\*行都没有写具体读的长度，应分别改为

raf.read(bytes, 0, 20);

raf.read(bytes, 0, 18);

## 47. FileInputStream in = new FileInputStream("...") 当使用in读取完文件内容后，换一种方式再使用in读取，发现读不出来了。

【关键字：FileInputStream流】【难度：中级】【掌握程度：掌握】

解决方案：因为你使用的in这个FileInputStream对象读取过了，并且读完了。可以再new一个出来重新读取。

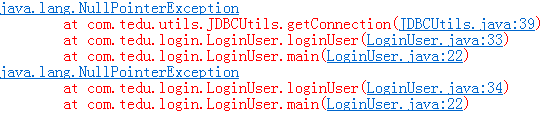
## 48. ring file = "E:/demo.txt";   FileInputStream in = new FileInputStream(file);   byte[] bytes = new byte[file.length()];   in.read(bytes);   System.out.println(new String(bytes,"gbk"));

## 49.该程序运行之后控制台只能输出一部分，剩下的用了个“？”代替

【关键字：IO流操作】【难度：初级】【掌握程度：掌握】

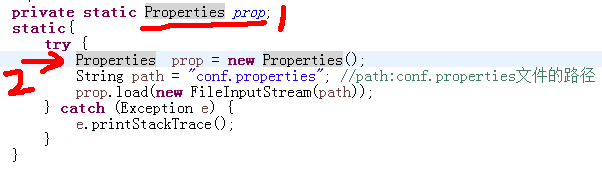
解决方案：这里创建的byte数组的长度出了问题，不能用字符串file的长度当作数组的长度。如果要用file.length()来创建byte数组，建议使用File file = new File("E:/demo.txt");

## 50.代码bug



【关键字：空指针异常】【难度：初级】【掌握程度：掌握】

解决方案：



去掉2处的Properties

## 51. FileNotFoundException

【关键字：找不到文件】【难度：初级】【掌握程度：掌握】

解决方案：Conf.properties 在加载的时候找不到文件,配置文件位置错误,直接加载配置文件需要在当前项目目录下,应放在工程目录下

## 52.NoClassNotFoundException:com.masql.jdbc.Driver  类找不到异常

【关键字：找不到驱动类】【难度：初级】【掌握程度：掌握】

解决方案：驱动名称写错,需要修改配置文件,将写错的驱动名称,改为正确的.

## 53. jdbc实现添加数据的时候，数据库里面会连续添加两条相同的数据。

【关键字：jdbc】【难度：初级】【掌握程度：掌握】

解决方案：检查jdbc代码，executeUpdate调用了两次。

## 54. java.net.ConnectException  Connection refused: connect

【关键字：数据库连接】【难度：初级】【掌握程度：掌握】

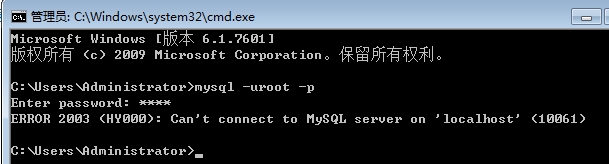
解决方案：安装mysql的时候修改了默认端口号，连接的时候端口号不是3306了。

## 55. jdbc插入中文数据之后数据库中查询出来是乱码

【关键字：jdbc】【难度：初级】【掌握程度：掌握】

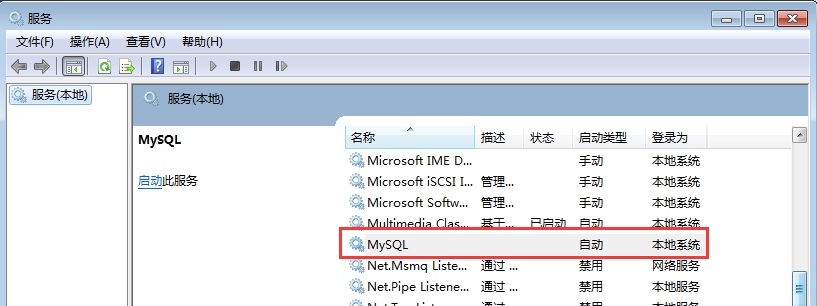
解决方案：set names gbk；之后再查询。如果是???，则是安装数据库的时候字符集设置错误，需要重装mysql，或者修改表的字符集。

## 56. 数据库打不开



【关键字：数据库】【难度：初级】【掌握程度：掌握】

解决方案：mysql服务未启动



## 57. 第六题，mysql命令行全班都报错：s0 data too long

【关键字：mysql】【难度：初级】【掌握程度：掌握】

解决方案：使用navicat工具执行命令就正常了

## 58. pow(a, b) 求a的b次方，这个函数占用的系统的函数名的，while i<b-1为负数的时候也会执行系统的pow函数

【关键字：函数】【难度：初级】【掌握程度：掌握】

解决方案：改pow函数名，不要和系统函数名一致

## 59. 登录服务器跟访问数据库的区别

【关键字：数据库】【难度：初级】【掌握程度：掌握】

解决方案：举例mysql，先登录数据库服务器才能通过use访问数据库

## 60. varchar和char的区别

【关键字：数据库字符】【难度：初级】【掌握程度：掌握】

解决方案：主要他们暂用内存的区别，varchar按照实际内容暂用内存，char按照定义字节长度暂用内存

## 61. mysql闪退，不可以开2个客户端

【关键字：mysql】【难度：初级】【掌握程度：掌握】

解决方案：最后装个mdb就可以了

## 62. pom.xml一直报错

【关键字：pom配置】【难度：初级】【掌握程度：掌握】

解决方案：查看maven配置是不是有问题，重新配置maven，检查maven路径

## 63. 创建maven项目，报cannot read lifecycle mapping metadata for artifact org.apache.maven.plugins:mav

【关键字：maven】【难度：初级】【掌握程度：掌握】

解决方案：配置都没问题，将本地.me文件夹下的repository文件夹删除，重新部署。

## 64. address already in use

【关键字：服务器端口】【难度：初级】【掌握程度：掌握】

解决方案：端口被占用，需要调整端口号，更换其他的

## 65. 通过浏览器访问页面时，显示的页面是整个html的文本格式，存在所有标签。

【关键字：HTML文本格式】【难度：初级】【掌握程度：掌握】

解决方案：检查HttpResponse部分代码，response.setProtocol(request.getProtocol())中的request写成了response

## 66. maven项目手动配置第三方jar包，项目里的代码不能运行？

【关键字：Maven】【难度：初级】【掌握程度：掌握】

解决方案：maven项目是自动管理jar包的，不能自己手动配置。必须要用pom.xml文件来进行操作jar包

## 67. 关于DecimalFormat类底层代码针对parse()方法会有一个自动判断返回值类型的情况，学生出现了类型转换异常

【关键字：DecimalFormat】【难度：初级】【掌握程度：掌握】

解决方案：问题未解决

## 68. 按钮  submit没反应？

【关键字：提交按钮】【难度：初级】【掌握程度：掌握】

解决方案：form标签没有post请求

## 69. 部分css属性没有生效？

【关键字：CSS属性】【难度：初级】【掌握程度：掌握】

解决方案：

在编写css时,未对上一个css属性进行大括号关闭,导致此css以下编写的css都无法生

## 70. js代码，写修改方法之前，添加删除功能都能正常运行，写了修改之后，前面的都不行了？

【关键字：js】【难度：初级】【掌握程度：掌握】

解决方案：

修改的方法里面有什么符号写错了，导致整个js代码块出错。浏览器F12控制台打开，看报错位置，根据位置找错。

## 71. ArrayIndexOutOfBoundsException？

【关键字：数组】【难度：初级】【掌握程度：掌握】

解决方案：

检查数组的长度，在引用数组时是否超出其范围。

## 72.input == null! atImageIO.read(Unknown Source）？

【关键字：数据源】【难度：初级】【掌握程度：掌握】

解决方案：

检查要加载的图片的名称是否正确

## 73. 射击游戏，敌机无法显示出来

【关键字：射击游戏】【难度：初级】【掌握程度：掌握】

解决方案：

使用输出语句，查看生成的敌机对象具体位置，发现初始化位置为地图外，解决完毕！

## 74. 飞机大战中，背景图片的移动出现断层

【关键字：射击游戏】【难度：初级】【掌握程度：掌握】

解决方案：public void paintObject(Graphics g) {  
 //画sky对象  
 g.drawImage(getImage(), x, y, null);  
 g.drawImage(getImage(), x, y1, null);  
 }  
sky类中的paint方法重写父类中的方法名写错

## 74. tomcat启动闪退

【关键字：tomcat】【难度：初级】【掌握程度：掌握】

解决方案：1、环境变量配置错误

2、tomcat没有正确解压

3、端口号与其它程序冲突

## 75. 飞机大战中，暂停的时候，暂停的图片没有画出来

【关键字：】【难度：初级】【掌握程度：掌握】

解决方案：



## 76. 记事本手写servlet后 javac 不能编译，报字符集问题，改成utf-8之后也不行。

【关键字：】【难度：初级】【掌握程度：掌握】

解决方案：新建一个.java文件 把之前的不能编译的java文件里的代码复制过去，以utf-8另存为，可以编译。