1、Statement和PreparedStatement有什么区别？

|  |
| --- |
| \* 1.使用PreparedStatement不再使用+号拼接sql语句，提高语句的简洁性和分离性，不容易出现语法错误！  \* 2.使用PreparedStatement有效的避免了sql注入问题  \* 3.使用PreparedStatement减少了编译的次数，提高了效率 |

2、where和having有什么区别？

|  |
| --- |
| 针对的数据 使用的关键字 位置  分组前筛选 原始表 where group by前面  分组后筛选 分组后的结果集 having group by后面  分组函数做条件放在having后面！ |

3、JDBC连接mysql数据库的步骤(并写出核心类和代码)？

|  |
| --- |
| DriverManager 驱动管理类  Connection连接接口  Statement命令接口  ResultSet 结果集接口  //1.获取连接  Connection connection = JDBCUtils.getConnection();    //2.执行增删改查    //①获取命令对象  Statement statement = connection.createStatement();    //②执行查询语句  ResultSet set = statement.executeQuery("SELECT COUNT(\*) FROM admin WHERE username='"+username+"' AND PASSWORD = '"+password+"'");    //③处理结果  if(set.next()) {  int count = set.getInt(1);  System.out.println(count>0?"Login Success!":"Login Failure!");  }    //3.关闭连接  JDBCUtils.close(set, statement, connection); |

4、SQL题：

为管理业务培训信息，建立3个表：

S （S#，SN，SD，SA） : 4个字段分别代表学号,学员姓名,所属单位,学员年龄

C（C#,CN）：2个字段分别代表课程编号、课程名称

SC（S#，C#，G）：3个字段分别代表学号、所选的课程编号、学习成绩

使用标准SQL语句实现以下要求：

查询选修课程名称为“java编程”的学员号和姓名

查询不选修课程编号为“C5”的学员姓名和所属单位

查询选修课程超过5门的学员学号和所属单位

|  |
| --- |
| select S.S#,S.SN from S,C,SC where S.S#=SC.S# and SC.C#=C.C# and C.CN=’Java编程’;  select S.SN,S.SD from S,C SC where S.S#=SC.S# and SC.C#=C.C# and C.C# !=C5;  select s.s#,s.sd from s where s# in (select s# from sc group by s# having count(\*)>5) |

5、Servlet的生命周期方法有哪几个？分别什么情况被调用？

|  |
| --- |
| 构造器:servlet第一次被访问时服务器创建对象时调用  init(ServletConfig config); servlet对象创建后立即调用完成 初始化[获取初始化参数、建立连接...]  service(ServletRequest req , ServletResponse res); 每次请求都会被调 用处理请求  destroy(); 销毁前调用完成收尾操作[关闭资源、持久化保存数据] |

6、转发和重定向的区别？

|  |
| --- |
| 转发和重定向的区别  转发 重定向    效率 一次请求、在服务器内部执行[高] 两次请求，浏览器发起[低]  发起者 服务器、request[请求没结束] 浏览器、response[给浏览器一个重定向响应  浏览器感知 不知道 知道  浏览器请求次数 1 2  地址栏地址 不变 改变  相对路径 地址栏地址和打开的资源不一致，失效 不失效  使用request域共享数据 可以 不可以 |

7、Servlet和JSP的区别是什么，有何联系？

|  |
| --- |
| 1.jsp经编译后就变成了Servlet.(JSP的本质就是Servlet，JVM只能识别java的类，不能识别JSP的代码,Web容器将JSP的代码编译成JVM能够识别的java类) 2.jsp更擅长表现于页面显示,servlet更擅长于逻辑控制. 3.Servlet中没有内置对象，Jsp中的内置对象都是必须通过HttpServletRequest对象，HttpServletResponse对象以及HttpServlet对象得到.  联系  jsp是对servlet的一种高级封装。本质还是servlet。  在第一次家在jsp页面的时候，会生成一个java文件，在虚拟机编译为.class文件，最后加载并初始化为一个servlet。 |

8 JSP的内置对象有哪些？分别有什么作用？

|  |
| --- |
| Page，pageContext，request，response，session，application，out，config，exception  Page指的是JSP被翻译成Servlet的对象的引用.  pageContext对象可以用来获得其他8个内置对象,还可以作为JSP的域范围对象使用.pageContext中存的值是当前的页面的作用范围》  request代表的是请求对象,可以用于获得客户机的信息,也可以作为域对象来使用，使用request保存的数据在一次请求范围内有效。  Session代表的是一次会话，可以用于保存用户的私有的信息,也可以作为域对象使用，使用session保存的数据在一次会话范围有效  Application：代表整个应用范围,使用这个对象保存的数据在整个web应用中都有效。  Response是响应对象,代表的是从服务器向浏览器响应数据.  Out:JSPWriter是用于向页面输出内容的对象  Config：指的是ServletConfig用于JSP翻译成Servlet后 获得Servlet的配置的对象.  Exception:在页面中设置isErrorPage=”true”，即可使用，是Throwable的引用.用来获得页面的错误信息。 |

9 radio单选组(name属性值为items)的第二个元素为被选中的元素，使用JS的dom和Jquery分别如何获取被选中元素的值？

|  |
| --- |
| var rdsObj = document.getElementsByName(Name);  checkVal = rdsObj[i].value   $('input[name=items]').get(1).checked |

10 请简述JSP页面的工作原理？

|  |
| --- |
| > Jsp本质就是Servlet  - 服务器接受到访问jsp的请求时，会先将jsp页面翻译为Servlet文件，然后再运行  - 翻译后的Servlet类继承了HttpJspBase extends HttpServlet  - 翻译后的Servlet在tomcat的web.xml中统一进行了配置[servllet、 servlet-mapping]  > 服务器处理第一次访问JSP页面请求时的流程  - 根据请求找到对应的jsp页面  - 服务器先将jsp页面翻译为Servlet.java文件  - 将java类编译为class文件  - 服务器再调用其生命周期方法处理请求  构造器  init  service[只要服务器没有重启、jsp页面代码没有被修改，服务器第N此接受到请求时只会调用service处理请求  > 如果项目中大量的使用的jsp页面，如何提高服务器处理用户请求的效率  服务器启动后自己先将所有的jsp页面访问一次 |

11 有两个页面a.jsp和b.jsp，要从a.jsp传值到b.jsp有哪几种实现方式？

|  |
| --- |
|  |

12 四大域对象及其范围？

|  |
| --- |
| pageContext: 在当前jsp页面范围共享数据  每个jsp页面都有自己特有的页面上下文对象，离开则失效  是jsp页面特有的，Servlet中没有  应用： 在页面开始时配置参数，在其他地方获取使用  request: 在一次请求范围内共享数据[直接访问一个资源得到响应、转发 ]  应用： 访问Servlet时操作数据库得到数据，然后共享到request域中，再转发到jsp页面可以获取数据并显示  session： 在浏览器的一次会话中共享数据  只关心浏览器是否关闭，和服务器关闭没有关系，不同浏览器不共享会话  应用：在浏览器的一次会话中保存用户的登录状态、验证码、购物车  application： 在项目的一次运行过程中共享数据  应用：保存全局的参数 |

13 -cookie过期和session超时区别

|  |
| --- |
|  |

14 JavaWeb阶段哪里使用过$，以及它的作用？

|  |
| --- |
| 根据传入参数的不同有四个作用  1、传入参数为函数：$(function(){}) 相当于文档加载结束 window.onload = function(){}  2、传入参数为html字符串：$("html字符串") 相当于根据字符串创建节点对象并返回包装的jquery对象  3、传入参数为选择器字符串：$("选择器字符串") 根据选择器字符串查找匹配的元素集合封装为jquery对象返回  4、传入参数为dom对象：$(domObj) 将dom对象转为jquery对象  -${key } 全域查询，从最小域到最大域查询，找到则返回值  -${xxxScope.key} 指定域查询值  ageScope、requestScope、sessionScope、applicationScope  -${bean.属性名} ：javabean必须是一个标准的javabean，私有属性共有getset方法  获取域中对象的属性值 相当于调用对象的get属性名方法  ${user.username} ==> user.getUsername 以属性名的形式去调用get方法  BeanHandler<>(type)： 可以将数据库查询的一条记录封装为一个对象  通过反射根据type创建一个对象：type.newInstance();  在根据查询到的数据库记录的列名去匹配type中的setXXX方法，将值设置给对象  -${map.key}  获取map指定键对应的值键没有特殊符号：${map.key}  键有特殊符号：${map['t2-1']}  -${list[index]}  取出list指定索引位置的元素 |

15 Jquery常见选择器有哪些、常见的事件有哪些?

|  |
| --- |
| ①基本选择器  标签选择器：$("标签名")  id选择器： $("#id值")  class选择器： $(".class值")  选择所有： $("\*")  复合选择器： $("seletor1 , selector2....")  ②层次选择器  祖先后代选择器： $("祖先 后代")  子选择器： $("父亲>儿子")  下个兄弟选择器：$("上个兄弟+下一个兄弟")  后面所有兄弟：$("上一个兄弟~后面兄弟")  ③过滤选择器  参考API  事件略 |

16 简述Session和Cookie的联系(Session对象能够在一次会话中共享数据的原理)？

|  |
| --- |
| 服务器从请求报文中查找JSESSIONID的cookie  - 如果没有： 证明浏览器第一次访问服务器需要使用session对象  服务器创建session对象并返回  服务器再创建Cookie("JSESSIONID" , UUID.randomUUID().toString().replace("-",""));并将cookie交给浏览器  服务器再将session作为值，JSESSIONID的值作为键保存到服务器内部维护的SessionMap<String,Session>中 [将浏览器的键和服务器的值关联起来]  - 如果有： 证明浏览器之前访问过服务器，并且服务器已经为其分配了session对象了  服务器获取JSESSIONIDcookie的值  再根据值作为键获取SessionMap中对应的Session值[session对象也有可能被服务器销毁了]  > 如果获取到session对象则返回使用 就可以使用本次会话中的唯一的一个session对象  > 如果获取不到，证明session已经销毁  登录超时  服务器会将请求当做第一次访问服务器时进行处理 |

17 HTTP协议是无状态的，服务器如何维持session的？

|  |
| --- |
|  |

19 HTTP get请求 和 post请求的区别,html from表单不设置method属性默认使用哪种请求方式？

|  |
| --- |
| method属性： 表单数据提交的方法  默认值是GET(get方式提交数据时表单会将参数和值拼接在url地址后 url?username=xxx&password=112e21)地址栏参数可见，不安全 ，效率高  POST方式 ， 参数不在地址栏中显示，相对安全,可以上传文件 |

20 Jquery对象和dom对象有什么区别，如何互相转换？

|  |
| --- |
| dom对象： 通过原生的js HTML DOM标准方法查到的对象就是dom对象  dom对象可以通过原生js api中的方法获取标签的属性和子标签  query对象： 通过$包装或查找到的就是jquery对象，本质是dom对象的集合  jquery对象只能调用jquery api中的方法或属性  dom对象转为jquery对象：$(domObj)  jquery对象转dom对象： 遍历可以获取jquery对象每个索引位置上的dom对象  $jqueryObj[0]  命名规范： jquery对象推荐使用$开始命名 |

21 Ajax验证用户名是否存在

<input type=”text” name=”username”/>

要求：使用jquery获取标签绑定内容改变监听，当内容改变时通过$.ajax()方法提交用户名给服务器验证用户名是否存在

|  |
| --- |
|  |