

1 注释、注解和 Junit

一. 单选题 (共6题, 66.6分)

1. (单选题)用于指定注解存活时间的元注解为 ()

- A. @Target
- B. @Retention
- C. @Documented
- D. @Inherited



11.1 分

2. (单选题)为了使得注解在程序运行时被反射读取, 则应设置@Retention取值为 ()

- A. RetentionPolicy.SOURCE
- B. RetentionPolicy.CLASS
- C. RetentionPolicy.RUNTIME
- D. RetentionPolicy.JVM



11.1 分

3. (单选题)@Override注解只在源码阶段保留, 它的@Retention取值为 ()

- A. RetentionPolicy.SOURCE
- B. RetentionPolicy.CLASS
- C. RetentionPolicy.RUNTIME
- D. RetentionPolicy.JVM



11.1 分

4. (单选题)用于指定注解运用地方的元注解为 ()

- A. @Target
- B. @Retention
- C. @Documented
- D. @Inherited



11.1 分

5. (单选题) 设置一个注解只能修饰方法和类型（包括：类、接口、枚举）的元注解代码应为（）

- A. @Target(ElementType.METHOD,ElementType.TYPE)
- B. @Target({ElementType.METHOD,ElementType.TYPE})
- C. @Retention(ElementType.METHOD,ElementType.TYPE)
- D.
@Retention({ElementType.METHOD,ElementType.TYPE})



11.1 分

6. (单选题) 以下不属于Java注释的为（）

A. //这是单行注释

B.
/*
这是多行注释
*/

C.
/**
* 这是文档注释
**/

D. <|这是一个单行注释 >



11.1 分

二. 多选题 (共2题, 22.2分)

7. (多选题)抑制一个类上的所有警告, 则能在该类上使用的代码为 ()

- A. @SuppressWarnings("all")
- B. @SuppressWarnings(value="all")
- C. @SuppressWarnings(value={"all"})
- D. @SuppressWarnings({"all"})
- E. @SuppressWarnings(value=["all"])



11.1 分

8. (多选题)关于JUnit说法正确的是 ()

- A. JUnit框架, 可以独立于main方法完成单元测试
- B. @Before注解修饰的方法, 只在所有的测试方法前运行一次, 而不会在每一个测试方法前都运行一次
- C. JUnit框架中, 不能使用断言判断测试结果是否正确
- D. 使用@Test注解的方法, 必须是public void类型, 且不能有任何参数。
- E. JUnit支持一个项目中所有方法的自动化测试



11.1 分

三. 填空题 (共1题, 11.2分)

9. (填空题)

已知Student的类定义为:

```
package fafu.jsxy;
public class Student {
    public void study() {
        System.out.println("studying");
    }
    public void speak(){
        System.out.println("speaking");
    }
    private void sleep(){ //私有方法
        System.out.println("sleeping");
    }
}
```

TestAnnotation类的功能为: 通过反射, 执行Student类中所有方法。使用反射的好处为: 不需要知道Student类中定义的方法的名字, 就能够执行Student类中的所有方法。

请在下划线处填写正确的代码。

```
package fafu.jsxy;
import java.lang.reflect.Method;
public class TestAnnotation {
    public static void main(String[] args) throws Exception{
        Class _____ = Student.class; //获得Student的字节码对象
        Student student = (_____)studentClass.newInstance(); //通过字节码对象, 创建Student对象
        _____[] methods = studentClass.getDeclaredMethods(); //通过字节码对象, 获得Student类中的所有方法
        for (Method m:methods){
            m.setAccessible(true); //暴力反射
            m._____(student); //执行方法
        }
    }
}
```

11.2 分



2 集合与泛型

一. 单选题 (共7题, 63.6分)

1. (单选题)以下不属于Java的集合框架的组成部分是 ()

- A. ArrayList
- B. HashMap
- C. TreeSet
- D. Arrays



9 分

2. (单选题)Java中提供了对集合中的元素进行迭代访问的接口为 ()

- A. Iterator
- B. Collection
- C. List
- D. Set



9.1 分

试用水印

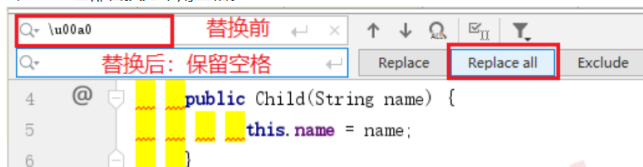
3. (单选题)

以下代码执行时，将引发的异常为（ ）

Plain Text

```
1  import java.util.*;
2  class Test {
3      public static void main(String[] args) {
4          List<String> list = new ArrayList<>();
5          for(int i=0;i<5;i++){
6              list.add(String.valueOf(i));
7          }
8          for (String element:list){
9              if (element.equals("0")){
10                 list.remove(element);
11             }
12         }
13     }
14 }
```

注：拷贝代码到idea，如果报“illegal character U+00A0”错误，可使用快捷键“ctrl+r”，弹出“替换”对话框，将“\u00A0”全部替换为半角空格。



- A. NullPointerException
- B. ConcurrentModificationException
- C. ArrayIndexOutOfBoundsException
- D. ClassCastException



9.1 分

4. (单选题)

以下程序的执行结果为:

Plain Text

```
1  import java.util.*;
2  class Child {
3      public String name;
4      public Child(String name) {
5          this.name = name;
6      }
7      @Override
8      public boolean equals(Object o) {
9          Child child = (Child) o;
10         return Objects.equals(name, child.name);
11     }
12     public static void main(String[] args) {
13         HashSet<Child> set = new HashSet<>();
14         set.add(new Child("小红"));
15         set.add(new Child("小花"));
16         set.add(new Child("小红"));
17         set.add(new Child("小明"));
18         System.out.println(set.size());
19     }
20 }
```

- A. 1
- B. 2
- C. 3
- D. 4



9.1分

5. (单选题)关于List的说法正确的是 ()

- A. 不能通过下标访问List中的元素
- B. List中的元素不能重复
- C. List的长度可以动态改变
- D. ArrayList和HashMap为List接口的实现类



9.1 分

6. (单选题)

已知List list=new ArrayList(), 以下代码为遍历输出list中的每个元素, 则下划线处的代码为 ()

Plain Text

```
1    for(int i=0;i<_____,i++){  
2        System.out.println(list.get(i));  
3    }
```

- A. list.length()
- B. list.size()
- C. list.length
- D. list.size



9.1 分

7. (单选题)

以下程序的执行结果为:

Plain Text

```
1 String s1 = "hello";
2 String s2 = "hello";
3 String s3 = new String("hello");
4 System.out.println(s1==s2);
5 System.out.println(s1==s3);
6 System.out.println(s1.equals(s3));
```

- A.
false
false
true
- B.
true
true
true
- C.
true
false
true
- D.
true
false
false



9.1分

二. 多选题 (共2题, 18.2分)

8. (多选题)关于HashMap和HashSet的说法正确的是 ()

- A. HashSet底层使用 HashMap作为存储结构。
- B. HashSet 的 add() 方法底层调用了 HashMap 的 put() 方法。
- C. HashSet中的元素可以重复
- D. HashMap以键值对的形式添加元素，键不能重复，值可以重复
- E. HashSet不能保证插入元素的顺序和循环输出元素的顺序一定相同



9.1 分

9. (多选题)

已知: String a=null;

则运行时, 会引发NullPointerException的代码为 ()

- A. System.out.println(a.equals("hello"));
- B. System.out.println(a.length());
- C. System.out.println(a==null);
- D. System.out.println("hello".equals(a));
- E. System.out.println(Objects.equals(a,"hello"));



9.1 分

三. 填空题 (共2题, 18.2分)

10. (填空题)

以下代码的功能为: 使用HashMap存储一个英中词典, 其中, 一个英文单词对应多个中文翻译。当接收控制台输入的英文单词, 将给出对应的多个中文翻译。

```
1  import java.util.*;
2  class Test {
3      public static void main(String[] args) {
4          //1、创建词典
5          Map<_____> map = new HashMap<>();
6          map.put("excellent", new String[]{"杰出的", "非凡的"});
7          map.put("annotation", new String[]{"注解", "评注"});
8          map.put("sale", new String[]{"销售", "销量"});
9          //修改 "excellent"对应的中文翻译
10         map._____("excellent", new String[]{"极好的", "出色的", "杰出的"});
11
12         //2、输出词典的内容
13         Set<Map._____<String, String[]>> mapSet = map.entrySet();
14         Iterator<Map.Entry<String, String[]>> iter = mapSet.iterator();
15         while (iter.hasNext()) {
16             Map.Entry<String, String[]> entry = iter.next();
17             System.out.print(entry.getKey() + "：");    //输出键
18             for (String _____ : entry.getValue()) {    //输出值
19                 System.out.print(value + " ");
20             }
21             System.out.println();
22         }
23
24         //3、接收控制台输入的英文单词，给出中文翻译
25         Scanner scanner = new Scanner(System.in); // 创建Scanner对象
26         System.out.println("请输入英文单词：");
27         String input = scanner.nextLine(); // 接收控制台输入的一行文本
28         if (map.containsKey(input)) {
29             String[] outputs = map._____ (input);
30             System.out.println("该单词的中文翻译为：");
31             for (String output : outputs) {
32                 System.out.println(output);
33             }
34         } else {
35             System.out.println("无该单词对应的中文");
36         }
37     }
38 }
```

9.1 分



11. (填空题)

以下代码的功能为：遍历ArrayList集合，删除以“实验1”结尾的所有字符串。

Plain Text

```
1 List<String> list = new ArrayList<>();
2 list.add("2020001张三实验1");list.add("2020001张三实验2");
3 list.add("2020002李四实验1");list.add("2020002李四实验2");
4 for (Iterator<String> iterator = _____,iterator. _____;) {
5     String str = iterator.next();
6     if(str.endsWith("实验1")){
7         _____; //删除迭代器指向的元素
8     }
9 }
10 System.out.println(list);
```

9.1 分



Java Web 开发入门

一. 单选题 (共3题, 42.8分)

1. (单选题)关于B/S的说法错误的是 ()

- A. 客户端无需安装, 有Web浏览器即可
- B. B/S架构无需升级多个客户端, 升级Web服务器即可
- C. B/S表现要达到C/S程序的程度, 需要较高的设计成本
- D. B/S在速度 and 安全性上的设计成本较低



14.2 分

2. (单选题)要在浏览器的地址栏上隐藏传输给服务器的参数, 需要将<form>标签的method属性设置为 ()

- A. hide
- B. get
- C. post
- D. submit



14.3 分

试用水印

3. (单选题)

已知register.html中的部分代码为：

Plain Text

```
1  <form action="register" method="post">
2    用户名: <input name="username" type="text"/><br/>
3    密码: <input name="userpass" type="password"/><br/>
4    <input type="submit" value="注册"/>
5    <input type="reset" value="重置"/>
6  </form>
```

RegisterServlet.java的部分代码为：

Plain Text

```
1  @WebServlet("/register")
2  public class RegisterServlet extends HttpServlet {
3      @Override
4      protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException,
5          String username = req.getParameter("userName");
6          String userpass = req.getParameter("userpass");
7      }
8  }
```

如果用户在register.html页面中，不输入任何内容，直接点击“注册”按钮，则RegisterServlet的doPost方法中的变量username和userpass的值分别为（ ）

- A. "" ""
- B. "" null
- C. null null
- D. null ""



14.3 分

二. 填空题 (共1题, 14.3分)

4. (填空题)

在如图2-1所示的下载页面download.html中输入正确的密码“123456”，则DownloadServlet生成的页面中显示“idea下载”的链接，如图2-2所示；否则显示如图2-3所示的出错提示。请在下划线处填写正确代码。

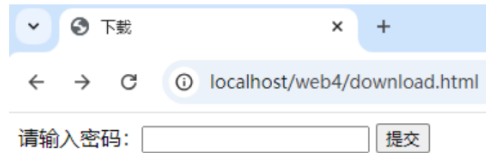


图2-1 下载页面

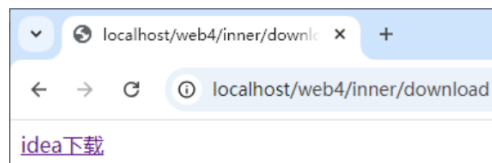


图2-2 密码正确后进入的下载页面

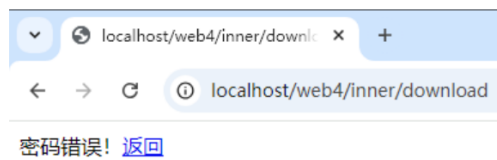


图2-3 密码错误

download.html文件（该文件直接位于项目的web节点下）：

Plain Text

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>下载</title>
6 </head>
7 <body>
8   <form action="inner/download" method="post">
9     请输入密码: <input type="password" name="userpass"/>
10    <input type="submit" value="提交">
11  </form>
12 </body>
13 </html>
```

DownloadServlet.java文件:

Plain Text

```
1 package it.servlet;
2
3 import javax.servlet.ServletException;
4 import javax.servlet.annotation.WebServlet;
5 import javax.servlet.http.HttpServlet;
6 import javax.servlet.http.HttpServletRequest;
7 import javax.servlet.http.HttpServletResponse;
8 import java.io.IOException;
9 import java.util.Objects;
10
11 @WebServlet(value="_____")
12 public class DownloadServlet extends HttpServlet {
13     @Override
14     protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException
15     {
16         _____.setContentType("text/html;charset=utf-8");
17         String userpass = _____.getParameter("userpass");
18         if (_____.equals("123456",userpass)){
19             resp.getWriter().println("<a href='https://www.jetbrains.com.cn/idea/download/?section=windows'");
20         }else{
21             resp.getWriter().println("密码错误! "+"<a href='download.html'>返回</a>");
22         }
23     }
```

14.3 分



三. 多选题 (共3题, 42.9分)

5. (多选题)属于向服务器发送get请求的情况包括 ()

- A. 在浏览器地址栏上直接输入URL, 回车
- B. 在浏览器上点击超链接
- C. 使用form表单提交数据时, form标签中, 没有写method属性
- D. 在form标签中, 指定method属性为: method="get"



14.3 分

6. (多选题)与@WebServlet(urlPatterns="/inner/random") 作用等同的代码包括 ()

- A. @WebServlet(name="/inner/random")
- B. @WebServlet(value="/inner/random")
- C. @WebServlet(WebInitParam="/inner/random")
- D. @WebServlet("/inner/random")



14.3 分

7. (多选题)关于POST和GET请求, 说法错误的是 ()

- A. POST在请求行中提交数据, GET在请求体中提交数据
- B. 向服务器传送大量数据, 应使用POST
- C. 涉及敏感数据的提交, 应使用GET
- D. 向服务器上传一张图片, 应使用POST



14.3 分

Servlet

一. 单选题 (共7题, 49.7分)

1. (单选题)

设有如下代码:

```
<servlet>
  <servlet-name>t</servlet-name>
  <servlet-class>it.servlet.TestServlet</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>t</servlet-name>
  <url-pattern>/test</url-pattern>
</servlet-mapping>
```

与上述代码作用相同的注解语句为 ()

- A. @WebServlet("/")
- B. @WebServlet("/test")
- C. @WebService("/")
- D. @WebService("/test")



7.1分

2. (单选题)在Servlet生命周期方法中,用于执行业务逻辑的方法是 ()

- A. 无参构造函数
- B. init()
- C. service()
- D. destroy()



7.1分

3. (单选题) 被称为web安全目录的文件夹为 ()

- A. META-INF
- B. WEB-INF
- C. webapps
- D. WebRoot



7.1 分

4. (单选题) Java Web项目被编译后，java源文件对应的.class文件存放于 () 文件夹中。

- A. src
- B. WEB-INF/classes
- C. web
- D. lib



7.1 分

5. (单选题) web项目的配置文件为 ()

- A. server.xml
- B. context.xml
- C. tomcat.xml
- D. web.xml



7.1 分

6. (单选题) javax.servlet.GenericServlet类的定义中，使用了abstract修饰的方法为 ()

- A. GenericServlet()
- B. init()
- C. service()
- D. destroy()



7.1 分

7. (单选题)

已知web.xml代码如下:

Plain Text

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"
3     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4     xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app
5     version="4.0">
6
7     <context-param>
8         <param-name>size</param-name>
9         <param-value>100</param-value>
10    </context-param>
11
12 </web-app>
```

TestServlet的init方法中读取size变量的值, 则下划线处应填写的代码为 ()

Plain Text

```
1 @WebServlet("/test")
2 public class TestServlet implements Servlet {
3
4     @Override
5     public void init(ServletConfig servletConfig) throws ServletException {
6         String size = servletConfig.getServletContext()._____("size");
7     }...
8 }
```



7.1 分

二. 多选题 (共5题, 35.9分)

8. (多选题) 动态页面开发技术包括 ()

- A. JSP
- B. Servlet
- C. HTML
- D. CSS
- E. ASPX



7.1 分

9. (多选题) Servlet的生命周期方法包括 ()

- A. 无参构造函数
- B. init()
- C. destroy()
- D. service()
- E. doFilter()



7.2 分

10. (多选题) 在Servlet的生命中, 只执行一次的方法包括 ()

- A. 无参构造函数
- B. init()
- C. destroy()
- D. service()
- E. servlet()



7.2 分

11. (多选题) 静态页面开发技术包括 ()

- A. JSP
- B. Servlet
- C. HTML
- D. CSS
- E. Javascript



7.2 分

12. (多选题)关于ServletConfig和ServletContext的说法, 错误的是 ()

- A. ServletConfig和ServletContext都能从web.xml文件中读取数据, 供Servlet使用
- B. ServletContext对象提供的setAttribute()方法, 只能往ServletContext对象中存储String类型的数据
- C. ServletContext对象中所存储的数据, 可以被当前网站的所有Servlet共享
- D. 当Web应用加载完毕, Tomcat立刻创建ServletConfig对象
- E. ServletConfig是局部对象, 只能获取当前Servlet中定义的数据



7.2 分

三. 填空题 (共2题, 14.4分)

13. (填空题)

已知web.xml文件的代码如下:

Plain Text

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"
3      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4      xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app
5      version="4.0">
6  <!-- 全局参数-->
7      <context-param>
8          <param-name>homePage</param-name>
9          <param-value>home.jsp</param-value>
10     </context-param>
11 <!-- TestServlet的配置-->
12     <servlet>
13         <servlet-name>test</servlet-name>
14         <servlet-class>it.servlet.TestServlet</servlet-class>
15         <init-param>
16             <param-name>databaseURL</param-name>
17             <param-value>jdbc:mysql:192.168.1.4:3306/test</param-value>
18         </init-param>
19     </servlet>
20     <servlet-mapping>
21         <servlet-name>test</servlet-name>
22         <url-pattern>/test</url-pattern>
23     </servlet-mapping>
24 </web-app>
```

请根据注释，完成TestServlet的代码。

Plain Text

```
1  package it.servlet;
2  import javax.servlet.*;
3  import java.io.IOException;
4  import java.util.Objects;
5
6  _____ class TestServlet implements Servlet {
7      String databaseURL;
8
9      @Override
10     public void init(ServletConfig servletConfig) throws ServletException {
11         //获取参数databaseURL的值
12         databaseURL = _____.getInitParameter("databaseURL");
13     }
14
15     @Override
16     public ServletConfig getServletConfig() {
17         return null;
18     }
19
20     @Override
21     public void service(ServletRequest servletRequest, ServletResponse servletResponse) throws Ser
22         //获取ServletContext对象
23         ServletContext servletContext = _____.getServletContext();
24         //获取全局变量homePage的值
25         String homePage = _____.getInitParameter("homePage");
26         //获取请求对象中的参数"inputURL"的值
27         String inputURL = servletRequest._____("inputURL");
28         servletResponse.setContentType("text/html;charset=utf8");
29         if (!Objects.equals(inputURL,databaseURL)) { // 比较字符串的内容
30             servletResponse.getWriter().println("连接字符串错误! 请回到<a href='"+homePage+"'>首页</a>");
31         }else {
32             servletResponse.getWriter().println("连接字符串与预设的一致! ");
33             int index=inputURL.indexOf("/");
34             //从连接字符串中获取数据库的名字
35             String dbName = inputURL._____ (index+1);
36             //将数据库名存储到ServletContext对象中
37             servletContext._____ ("dbName",dbName);
38         }
39     }
40
41     @Override
42     public String getServletInfo() {
43         return null;
44     }
45
46     @Override
47     public void destroy() {
48
49     }
50 }
```

7.2 分



14. (填空题)

每次访问LogServlet，都会在“WEB-INF”下的“logs.txt”文件中增加一条显示当前时间、以及一首诗歌名的记录。
logs.txt为utf8文件，内容如下所示：

1	2024-10-29 23:08:10 园丁集
2	2024-10-29 23:08:39 生如夏花
3	2024-10-29 23:09:37 生如夏花
4	2024-10-29 23:10:38 闲暇

LogServlet.java源文件：

LogServlet.java源文件:

Plain Text

```
1  package it.servlet;
2
3  import javax.servlet.*;
4  import javax.servlet.annotation.WebServlet;
5  import java.io.*;
6  import java.text.SimpleDateFormat;
7  import java.util.Date;
8  import java.util.Random;
9
10 @WebServlet("/log")
11 public class LogServlet implements Servlet {
12     String[] poems;
13
14     @Override
15     public void init(ServletConfig servletConfig) throws ServletException {
16         poems = new String[]{"太阳颂", "生如夏花", "飞鸟集",
17             "园丁集", "吉檀迦利", "闲暇"};
18     }
19
20     @Override
21     public ServletConfig getServletConfig() {
22         return null;
23     }
24 }
```

```

25  @Override
26  public void service(ServletRequest servletRequest, ServletResponse servletResponse) throws Ser
27      //获取ServletContext对象
28      ServletContext servletContext = servletRequest.getServletContext();
29      //获取logs.txt文件的实际路径
30      String filePath = _____.getRealPath("/WEB-INF/logs.txt");
31      //创建输出流对象。代码中，指定了追加模式为true，且生成的文件为utf8格式
32      BufferedWriter bufferedWriter = new BufferedWriter
33          (new _____(new FileOutputStream(filePath, true), "utf8"));
34      //创建Date对象表示当前时间
35      Date date = new Date();
36      //创建SimpleDateFormat对象，指定日期时间格式
37      SimpleDateFormat simpleDateFormat = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");
38      //使用SimpleDateFormat对象，格式化当前时间
39      String currentTime = simpleDateFormat.format(_____);
40      //随机返回一首诗歌
41      String randomPoem = poems[new Random().nextInt(poems._____)];
42      //在输出流中，写入当前时间、诗歌名
43      bufferedWriter.write(currentTime + " " + randomPoem);
44      //在输出流中，写入换行符
45      bufferedWriter.newLine();
46      //关闭输出流
47      bufferedWriter.close();
48  }
49
50  @Override
51  public String getServletInfo() {
52      return null;
53  }
54
55  @Override
56  public void destroy() {
57
58  }
59 }

```

7.2 分



JSP 基本语法

一. 单选题 (共7题, 58.1分)

1. (单选题)

以下代码的输出结果为 ()

```
<%  
    String str = null;  
    out.println("str="+str);  
%>
```

- A. str=
- B. str=null
- C. 运行时抛出异常
- D. 编译出错, 不能运行



8.3 分

2. (单选题)JSP页面在第一次运行的时候被JSP引擎转换为 ()

- A. CSS文件
- B. HTML文件
- C. Java文件
- D. Javascript文件



8.3 分

3. (单选题)JSP的主要作用为 ()

- A. 设计网页特效
- B. 代替Servlet程序回传html页面
- C. 美化网页
- D. 过滤HTML、Servlet等web资源



8.3 分

4. (单选题)定义JSP程序段的代码为 ()

- A. <%= JSP程序段 %>
- B. <% JSP程序段 %>
- C. <%! JSP程序段 %>
- D. <%-- JSP程序段 %>



8.3 分

5. (单选题) JSP的本质是一个 ()

- A. HTML
- B. Servlet
- C. Filter
- D. Javascript



8.3 分

6. (单选题) 在JSP中, 定义JSP表达式的代码为 ()

- A. <%=JSP表达式%>
- B. <% JSP表达式%>
- C. <%! JSP表达式%>
- D. <!--JSP表达式%>



8.3 分

7. (单选题)

以下代码的执行结果为 ()

```
<%  
out.println(97);  
out.write(97);  
%>
```

- A.
97
a
- B. 97 97
- C.
97
97
- D. 97 a



8.3 分

二. 多选题 (共1题, 8.3分)

8. (多选题)关于JSP的说法, 正确的是 ()

- A. JSP程序段翻译之后在_jspService ()方法中
- B. JSP声明翻译之后会放在_jspService()方法中
- C. JSP表达式翻译之后在_jspService()方法中
- D. JSP表达式中的代码, 必须以分号结束
- E. JSP程序段中不能使用request、response、out等JSP内置对象



8.3 分

三. 判断题 (共4题, 33.6分)

9. (判断题)HttpJspBase在jsp-api.jar包中

- A. 对
- B. 错



8.4 分

10. (判断题)JSP表达式<%=msg>翻译后的Java代码为: out.write(msg)

- A. 对
- B. 错



8.4 分

11. (判断题)浏览器的源代码中不能显示Java注释以及JSP特有注释。

- A. 对
- B. 错



8.4 分

12. (判断题)JSP特有注释的格式为: <!-- 注释内容 -->

- A. 对
- B. 错



8.4 分

JSP 内置对象（一）

一. 单选题（共5题，50分）

1. (单选题)

单击“提交”按钮，服务器资源edit能够获取到参数“index=<%=index%>”，则下划线处应填写的代码为（ ）

```
<form action="edit?index=<%=index%>" _____>
```

```
    学号 <input type="text" name="sno" readonly="true" value=<%=sno%>> <br/>
```

```
    姓名 <input type="text" name="name" value=<%=name%>> <br/>
```

```
<input type="submit" value="提交"> </br>
```

- A. method = "post"
- B. method = "get"
- C. method = "hide"
- D. method = "submit"

10 分

2. (单选题)语句response.setContentType("text/html;charset=utf-8")的功能完整表达为（ ）

- A. 设置对请求体中的数据进行重新编码的方案为utf-8
- B. 设置服务器响应给浏览器的编码为utf-8
- C. 设置浏览器对其接收到的数据的解码方案为utf-8
- D. 设置服务器响应给浏览器的编码以及浏览器的解码方案为utf-8

10 分

3. (单选题)

设有如下代码。

page1.jsp:

```
<form action="page2.jsp" method="post">
  姓名: <input type="text" name="username"> <br>
  性别: <input type="radio" name="gender" value="male" checked>男
        <input type="radio" name="gender" value="female">女 <br>
  兴趣: <input type="checkbox" name="interests" value="sport" checked>体育
        <input type="checkbox" name="interests" value="music">音乐 <br>
  <input type="submit" value="提交">
</form>
```

page2.jsp:

```
<% // 以下代码的getParameter()中填写的参数名均正确
  out.println("姓名为: "+request.getParameter("username")+"<br>");
  out.println("性别为: "+request.getParameter("gender")+"<br>");
  out.println("第一项兴趣为: "+request.getParameter("interests"));
%>
```

当用户打开page1.jsp后, 不在页面中进行任何操作, 直接点击“提交”按钮, 则page2.jsp的输出结果为 ()

- A. 姓名为:
性别为: male
第一项兴趣为: sport
- B. 姓名为:
性别为: 男
第一项兴趣为: 体育
- C. 姓名为: null
性别为: male
第一项兴趣为: sport
- D. 姓名为: null
性别为: 男
第一项兴趣为: 体育

10 分

4. (单选题)语句response.setCharacterEncoding("utf-8")的作用为 ()

- A. 设置请求体的编码方案为utf-8
- B. 设置服务器响应给浏览器的编码为utf-8
- C. 设置浏览器对其接收到的数据的解码方案为utf-8
- D. 设置服务器响应给浏览器的编码以及浏览器的解码方案为utf-8

10 分

5. (单选题)在jsp中, 跳转到百度页面, 应使用代码 ()

- A. request.getRequestDispatcher("www.baidu.com").forward(request,response);
- B. request.getRequestDispatcher("https://www.baidu.com").forward(request,response);
- C. response.sendRedirect("www.baidu.com");
- D. response.sendRedirect("https://www.baidu.com");

10 分

试用水印

二. 多选题 (共3题, 30分)

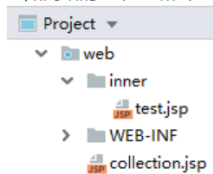
6. (多选题)关于请求转发和重定向说法正确的是 ()。

- A. 请求转发使用的是客户端路径
- B. 重定向使用的是服务器端路径
- C. 想实现跨app的资源跳转, 必须使用重定向
- D. 如果两个页面之间需要通过 request来传值, 则必须使用转发
- E. 重定向可以防止“浏览器刷新, 导致用户重复提交表单”的问题

10 分

7. (多选题)

已知网站的上下文路径为"/demo", 项目中的部分目录结构如下所示:



在test.jsp中, 重定向到collection.jsp可以使用代码包括 ()

- A. `response.sendRedirect("collection.jsp");`
- B. `response.sendRedirect("/collection.jsp");`
- C. `response.sendRedirect(request.getContextPath()+"/collection.jsp");`
- D. `response.sendRedirect(application.getContextPath()+"/collection.jsp");`
- E. `response.sendRedirect(request.getServletContext().getContextPath()+"/collection.jsp");`

10 分

8. (多选题)关于`request.setCharacterEncoding("utf-8")`的说法正确的是 ()

- A. 用于设置浏览器的页面编码为utf-8
- B. 用于解决get请求和post请求的中文乱码
- C. 用于设置服务器对请求体中数据进行重新编码的字符方案为utf-8
- D. 用于设置服务器响应给浏览器的字符编码方案为utf-8
- E. 在调用`request.getParameter()`方法之前使用, 才能起作用

10 分

三. 填空题 (共2题, 20分)

9. (填空题)

编程实现：如果用户不是通过以“http://localhost:8080/web7”开头的页面来到news.jsp，则跳转到当前网站下的index.jsp。

news.jsp文件：

Plain Text

```
1  <%@ page contentType="text/html;charset=UTF-8" language="java" %>
2  <html>
3  <head> <title>Title</title></head>
4  <body>
5  <%
6      String referer = _____.getHeader("referer");//获取上一级页面的URL
7      //如果request请求头的referer字段不是以“http://localhost:8080/web7”开头，则重定向当前网站的index.
8      if (_____ || !referer.startsWith("http://localhost:8080/web7")){
9          response.sendRedirect(request.getContextPath()+ "_____");
10         _____;
11     }
12 %>
13 新闻内容
14 </body>
15 </html>
```

10 分

10. (填空题)

index.jsp页面提供了一个供用户选择一种或者多种车型的下拉列表框。当用户选择车型后，点击“提交”按钮，将在show.jsp页面中显示用户的选择的所有车型值。

index.jsp:

Plain Text

```
1  <%@ page contentType="text/html;charset=UTF-8" language="java" %>
2  <html>
3  <head>
4    <title>多选下拉列表框</title>
5  </head>
6  <body>
7    <form action="show.jsp">
8      <label for="cars">选择多个车型: </label>
9      <select id="cars" _____ multiple>
10     <option value="volvo">Volvo</option>
11     <option value="saab">Saab</option>
12     <option value="opel">Opel</option>
13     <option value="audi">Audi</option>
14   </select><br><br>
15   <input type="submit" value="提交">
16 </form>
17 </body>
18 </html>
```

show.jsp代码:

Plain Text

```
1  <%@ page contentType="text/html;charset=UTF-8" language="java" %>
2  <html>
3  <head>
4    <title>Title</title>
5  </head>
6  <body>
7  <%
8    String[] cars = request._____("cars");
9    if (_____) {
10      out.println("您选择的车型为: ");
11      for (String in : cars) {
12        out.println(in);
13      }
14    } else {
15      out.println("您没有选中任何车型");
16    }
17  %>
18  </body>
19  </html>
```

10 分

JSP 内置对象（二）

一. 单选题（共7题，77.7分）

1. (单选题) 将图片image输出到浏览器的页面中，应使用的代码为（ ）

- A. ImageIO.write(image, "jpg", request.getOutputStream());
- B. ImageIO.write(image, "jpg", request.getWriter());
- C. ImageIO.write(image, "jpg", response.getWriter());
- D. ImageIO.write(image, "jpg", response.getOutputStream());

11.1 分

2. (单选题) 存储用户登录时的验证码文本，应使用的JSP内置对象为（ ）

- A. pageContext
- B. request
- C. session
- D. application

11.1 分

3. (单选题) JSP内置对象application的数据类型为（ ）

- A. Appliction
- B. ServletConfig
- C. ServletContext
- D. ServletApplication

11.1 分

4. (单选题) 为了让session的有效时间对Tomcat服务器的所有Web应用生效，应修改的配置文件为（ ）

- A. <Tomcat安装目录>\conf\server.xml
- B. <Tomcat安装目录>\conf\tomcat-users.xml
- C. <Tomcat安装目录>\conf\context.xml
- D. <Tomcat安装目录>\conf\web.xml

11.1 分

5. (单选题) Web服务器用于判断客户端的会话是否结束的机制为 ()

- A. 自主存储机制
- B. 超时机制
- C. 加密机制
- D. 死锁机制

11.1 分

6. (单选题) 定义session的失效时间为30分钟的正确代码为 ()

- A. session.setMaxInactiveInterval(30*60)
- B. session.setMaxInactiveInterval(30)
- C. session.setMaxAge(30*60)
- D. session.setMaxAge(30)

11.1 分

7. (单选题)

设置图片的来源为当前网站下的/checkCodeServlet, 则下划线处应填写的代码为 ()。

- A. <%=request.getContextPath(); %>
- B. <%=request.getContextPath()%>
- C. <%=response.getContextPath(); %>
- D. <%=response.getContextPath()%>

11.1 分

二. 多选题 (共2题, 22.3分)

8. (多选题) 被称为域对象的JSP内置对象包括 ()

- A. application
- B. session
- C. response
- D. request
- E. pageContext

11.1 分

9. (多选题)关于session说法错误的是 ()

- A. session对象存储于服务器中
- B. 当浏览器关闭, 存储于浏览器的sessionID消失, 服务器中的session对象也同时消失
- C. 如果Servlet是浏览器访问的第一个资源, 则服务器默认会为此次会话创建session对象
- D. session.setMaxInactiveInterval(60)的作用为: 设置session的失效时间为60分钟
- E. session对象中的数据只能在一次请求内使用, 不能被不同的请求共享

11.2 分

试用水印

Cookie

一. 单选题 (共2题, 33.2分)

1. (单选题)删除cookie使用的正确代码为 ()

- A. cookie.setMaxAge(0); request.addCookie(cookie);
- B. cookie.setMaxAge(0); response.addCookie(cookie);
- C. cookie.setMaxAge(-1); request.addCookie(cookie);
- D. cookie.setMaxAge(-1); response.addCookie(cookie);

16.6 分

2. (单选题)实现不同tomcat服务器的项目之间的cookie共享, 应使用的方法为 ()

- A. setPath()
- B. setDomain()
- C. setMaxAge()
- D. setLoad()

16.6 分

试用水印

二. 多选题 (共3题, 50.1分)

3. (多选题)关于session和cookie说法正确的是 ()

- A. session和cookie都是会话技术
- B. session的数据保存在服务器端, cookie的数据保存在客户端
- C. session保存的数据是String
- D. session随着会话的结束而销毁, cookie可以长期保存在客户端电脑中
- E. cookie保存的数据是Object

16.7 分

4. (多选题)

已知Servlet中与cookie相关的代码为:

```
Cookie c = new Cookie("name","lucy");  
cookie.setPath("/");  
request.addCookie(c);
```

其中, Servlet对应的网站的上下文路径为: /web8, 浏览器访问Servlet的路径为:

http://localhost:8084/web8/a/AServlet, 那么, 能够获取Servlet中保存的数据name=lucy的资源路径为 ()

- A. http://localhost:8084/web8/a/AServlet
- B. http://localhost:8084/web8/a/BServlet
- C. http://localhost:8084/web8/a/bbb/CServlet
- D. http://localhost:8084/web8/b/DServlet
- E. http://localhost:8084/web9/EServlet

16.7 分

5. (多选题)关于会话技术, 说法错误的是 ()

- A. 会话技术包括: Cookie、session、request等
- B. Cookie为JSP的内置对象
- C. Cookie对象的setMaxAge()方法用于设置cookie在服务器端的存活时间
- D. 会话技术指的是, 实现不同会话之间的数据共享
- E. session的setMaxInactiveInterval()方法用于设置session对象在服务器端的有效时间

16.7 分

三. 填空题 (共1题, 16.7分)

6. (填空题)

第一次访问product.jsp的显示结果如图1所示, 每个超链接代表一个商品类别。单击任何一个超链接时, 将把对应的商品类别存入cookie中。下一次, 在7天内访问product.jsp时, 则会在页面中显示上一次访问的商品类别, 如图2所示。

<p>商品类别列表</p> <p>服装 电器 礼品 娱乐</p> <p>上一次浏览的商品类别</p>	<p>商品类别列表</p> <p>服装 电器 礼品 娱乐</p> <p>上一次浏览的商品类别</p> <p>服装</p>
图1 第一次访问product.jsp	图2 第二次访问product.jsp

```
<!--product.jsp-->
<body>
  <h3>商品类别列表</h3>
  <a href="product.jsp?item=服装">服装</a>
  <a href="product.jsp?item=电器">电器</a>
  <a href="product.jsp?item=礼品">礼品</a>
  <a href="product.jsp?item=娱乐">娱乐</a>
  <h3>上一次浏览的商品类别</h3>
  <%
    String item = "";
    //读取cookie
    Cookie[] cookies = _____;
    if (cookies != null) {
      for (Cookie cookie : cookies) {
        if ("item".equals(_____)) {
          item = _____;
          item = URLDecoder.decode(item, "utf-8"); //使用utf-8对item (urlencoded字符串) 解码
          break;
        }
      }
    }
    //在页面中输出item
    out.print(item);
    item = _____.getParameter("item"); //获得用户点击的商品类别名, 例如, 服装
    if (_____ ) { //如果用户点击了某个商品类别, 则将该类别存入cookie中
      item = _____; //使用utf-8, 将item转换为urlencoded 格式的字符串
      Cookie c = new Cookie("item", item);
      c.setMaxAge(60 * 60 * 24 * 7);
      _____; //将c响应到客户端
    }
  %>
</body>
```

16.7 分

JDBC 连接 mysql

一. 填空题 (共2题, 33.2分)

1. (填空题)

Dbutils为访问数据库的通用类。

```
public class Dbutils {  
    protected Connection conn = null;  
    protected PreparedStatement pstmt = null;  
    protected ResultSet rs = null;  
    public Connection getConnection() { ... } //获取连接对象  
    public void closeAll(Connection conn, PreparedStatement pstmt, ResultSet rs) { ... } //关闭资源  
    public ResultSet executeQuery(String sql, Object[] params) { ... } //执行查询  
    public int executeUpdate(String sql, Object[] params) { //执行增删改  
        _____.getConnection();  
        int result = 0;  
        try {  
            pstmt = _____.prepareStatement(_____);  
            if (null != params) {  
                for (int i = 0; i < params.length; i++) {  
                    pstmt.setObject(_____, _____);  
                }  
            }  
            result = pstmt._____;  
        } catch (SQLException e) {  
            e.printStackTrace();  
        } finally {  
            this.closeAll(conn, pstmt, _____);  
        }  
        return result;  
    }  
}
```

16.6 分

2. (填空题)

使用上一题定义好的Dbutils，实现用户的登录。注：注册时，密码经过了md5加密后存入数据库中。md5算法的特点是，环境相同，同一字符串加密的结果也相同。

@WebServlet("/login")

```
public class LoginServlet extends HttpServlet {  
    public static String stringToMD5(String plainText) { //md5加密方法  
        byte[] secretBytes = null;  
        try {  
            secretBytes = MessageDigest.getInstance("md5").digest(plainText.getBytes("utf-8"));  
            /*getBytes()用于将字符串转换为字节数组；这里指定了编码字符集为utf-8；  
            如果没有在getBytes()中指定转换的编码字符集，  
            同一个字符串在windows和linux下的加密结果可能不同；  
            同一个字符串在web项目和java项目下的加密结果也可能不同*/  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
        String md5code = new BigInteger(1, secretBytes).toString(16);  
        for (int i = 0; i < 32 - md5code.length(); i++) {  
            md5code = "0" + md5code;  
        }  
        return md5code;  
    }  
}
```

```
@Override  
protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {  
    req.setCharacterEncoding("UTF-8");  
    resp.setContentType("text/html;charset=UTF-8");  
    String username = req.getParameter("username");  
    String userpass = req.getParameter("userpass");  
    Dbutils dbutils = _____;  
    //根据输入的用户名和密码查询数据库  
    String sql = "select * from users where username=? and userpass=?";  
    Object[] params = {username, _____}; //将username和md5加密后的密码作为查询参数  
    ResultSet resultSet = dbutils. _____;  
    try {  
        if( _____ ){ //登录成功，跳转到welcome.jsp  
            _____sendRedirect(req.getContextPath()+"/welcome.jsp");  
        }else{ //登录失败，将出错信息转发给login.jsp  
            req.setAttribute("error","用户名或者密码错误");  
            _____getRequestDispatcher("/login.jsp").forward(req,resp);  
        }  
    } catch (SQLException e) {  
        e.printStackTrace();  
    }finally{  
        dbutils.closeAll(dbutils.conn, dbutils.pstmt, dbutils.rs);  
    }  
}
```

16.6 分

二. 单选题 (共2题, 33.4分)

3. (单选题)设有如下代码:

```
Class.forName("com.mysql.jdbc.Driver");
Connection conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/my_db","root","123456");
String snameByUser = req.getParameter("sname"); //接收sname参数的值
String sql = "delete from student where sname='"+snameByUser+"'";
Statement statement = connection.createStatement();
int i = statement.executeUpdate(sql);
则当sname参数值为 ( ) 时, 上述代码将删除student表中的全部记录。
```

- A. ' or '1'='1
- B. ' or '1'='1'
- C. ' and '1'='1
- D. ' and '1'='1'

16.7 分

4. (单选题)JDBC中, 用于执行SQL语句的对象是 ()

- A. Connection
- B. Statement
- C. ResultSet
- D. DriverManager

16.7 分

三. 多选题 (共2题, 33.4分)

5. (多选题)

已知sage字段在student表中定义为int型。以下代码用于：根据用户输入的年龄，查询学生的学号sno和姓名sname。

```
Class.forName("com.mysql.jdbc.Driver");
Connection conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/my_db", "root", "123456");
String sage = req.getParameter("sage"); //接收用户输入的年龄sage
String sql = "select sno,sname from student where sage=" + sage;
//sage字段在student表中是int型，因此，不用单引号 ' 拼接SQL语句
Statement statement = conn.createStatement();
ResultSet resultSet = statement.executeQuery(sql);
while (resultSet.next()){
    String sno = resultSet.getString("sno");
    String sname = resultSet.getString("sname");
    resp.getWriter().println(sno+" "+sname);
}
```

当输入的年龄为（ ）时，将输出student表中的全部记录的sno和sname。

- A. 1 or 1=1
- B. ' or '1'='1
- C. 1 or true
- D. 12 or 2=2
- E. 2 or true

16.7 分

6. (多选题) JDBC API的类和接口包括（ ）

- A. Connection
- B. PreparedStatement
- C. ResultSet
- D. Servlet
- E. HttpSession

16.7 分