### 软件测试 实验报告—JMeter

### 1、实验目的

学习压力测试工具 JMeter 的使用:编写一个简单的注册登录小程序,使用 JMeter 分别对注册和登录进行压力测试。要求:

- (1) 注册和登录成功后要跳转到一个显示用户名的界面,并通过输入的用户名对跳转的页面进行验证;如果失败,则跳转到一个显示"操作失败"的页面,并对该返回页面的内容进行验证。
- (2) 压力测试过程中要求使用参数配置功能,测试多组用户登录和注册操作,需要测试 较多的并发用户数量,能够体现压力,例如执行时间至少 5 分钟。
- (3) 通过增加 Tomcat 对 JVM 内存的分配来进行优化,给出优化前后的性能对比。

### 2、测试范围

对 JMeter 中的注册和登录操作进行测试。

### 3、 测试过程

## 3.1: 新建 test5 项目。进行登录和注册逻辑的编写。

创建实体类,model、dao 和 hibernate-mapping 首先创建数据库 chtTests,然后创建表 user



#### 创建实体类 User

```
■ Project ▼
                  ⊕ 📱 🖈 🗢 🔑 build.gradle (test5) × © User.java ×
                                                                              📐 console 🗴 🌀 HelloServlet.java 🗴 🏭 index.jsp 🗴 🌀 HelloApplication.java 🗴 🌀 HelloResource.java 🗴
                                                package org.tty.test5.model;
    gradle.
    build
                                                     private Integer userId;
private String username;
     gradle
                                                     private String password;
     ∨ 📭 main
          iava

✓ □ org.tty.test5
                                              public Integer getUserId() {
            dao
∨ ■ model
               © User
© HelloApplication
                                                    public void setUserId(Integer userId) {
               HelloResource
               HelloServlet
       > lig resources
> lig webapp
                                                    public String getUsername() {

✓ Im WEB-INF

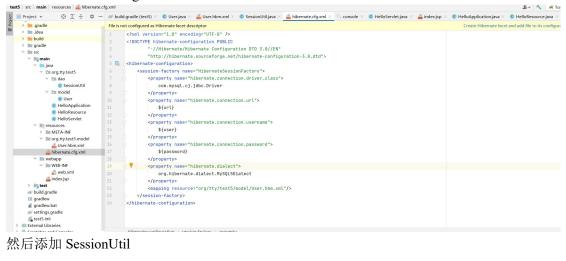
    web.xi
index.jsp

index.jsp
                                                    public void setUsername(String username) {
    gradlew
gradlew.bat
     e settings.gradle
     atest5.iml
                                                     public String getPassword() {
> IIII External Libraries
> % Scratches and Consoles
                                                         return password;
```

#### 然后创建 User.hbm.xml



### 然后创建 hibernate.cfg.xml 配置文件



### 然后添加 SessionUtil

```
🚇 Eile Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help test5 - Session Util.java [test5.main]
\textbf{test5} \mathrel{\big\rangle} \mathsf{src} \mathrel{\big\rangle} \mathbf{main} \mathrel{\big\rangle} \mathsf{java} \mathrel{\big\rangle} \mathsf{org} \mathrel{\big\rangle} \mathsf{tty} \mathrel{\big\rangle} \mathsf{test5} \mathrel{\big\rangle} \mathsf{dao} \mathrel{\big\rangle} \mathrel{\textcircled{\textbf{c}}} \mathsf{SessionUtil}
                                                                                                                                                                                                                                                                                                Project v 🕀 🚊 🕇 💠 — 🔊 build.gradle (test5) × 🌀 User,java × 🍰 User.hbm.xml × 💽 SessionUtil.java × 👶 UserDao,java × 🍰 hibernate.cl ∨ Database

    Test5 C:\Users\cht\Program\Uava\Test\tr\
    Test5 C:\Univer\Uava\Test\tr\
    Test5 C:\Users\cht\Program\Uava\Test\tr\
    Test5 C:\Users\cht\Program\Uava\Test\tr\
    Test5 C:\Users\cht\Program\Uava\Test\tr\
    Test5 C:\Univer\Uava\Test\tr\
    Test5 C:\Univer\Uava\Test\tr\
    Test5 C:\Univer\Uava\Test\tr\
    Test5 C:\Univa\Test\tr\
    Test5 C:\Univa\Uava\Test\tr\

                                                                                                                                                                                                                                                                                                                                 A2 ^ v + @ G
                                                                                                                                                                                                                                                                                                                                                       ∨ \ chtTest
            > idea
                                                                                                                     private static Session session;
                                                                                                                                                                                                                                                                                                                                                                ∨ 🚮 chtT
∨ 🖿 ti
            > mgradle
                                                                                                                    if (session == null) {
    Session
                                                                                                                      public static Session getSession() {
            ∨ 🖿 src
                 ∨ i main
                                                                                                                                         SessionFactory sessionFactory = new Configuration().configure().buildSessionFa
                       v 📄 java
                                                                                                                                            session = sessionFactory.openSession();

∨ □ org.tty.test5

                                    v 🛅 dao
          © SessionUtil 18
                                                                                                                                  return session;
                                              UserDao
                                 ∨ 🖿 model
                                                                                             20
                                              © User
                                                                                                                        @SuppressWarnings({"unchecked"})
                                      C HelloApplication
                                                                                                                        public <T> List<T> query(String hql) throws RuntimeException {
                                         C HelloResource
                                                                                                                                 Session session = getSession();
                                        © HelloServlet
                                                                                                                                 Query<T> queryObject = (Query<T>) session.createQuery(hql);
                        return queryObject.list();
                                META-INF

✓ Im org.tty.test5.model

                                          User.hbm.xml
                                                                                                                        public <T> void save(T instance) throws RuntimeException {
                                  ahibernate.cfg.xml
                                                                                            29
                                                                                                                          Session session = getSession();
                         v 🛅 webapp
                            ∨ ■ WEB-INE
                                                                                            30
                                                                                                                                     session.save(instance):
                                    a index.isp
                                                                                                               }
      Services
    userId integer auto_increment,
```

然后添加 UserDao

```
🚇 File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help test5 - UserDao.java [test5.main]
\textbf{test5} \ \rangle \ \text{src} \ \rangle \ \textbf{main} \ \rangle \ \textbf{java} \ \rangle \ \text{org} \ \rangle \ \textbf{tty} \ \rangle \ \textbf{test5} \ \rangle \ \textbf{dao} \ \rangle \ \textcircled{\textbf{c}} \ \textbf{UserDao}
                                                                                                                                    ■ Project ▼
                   😌 📱 🌣 — 🔊 build.gradle (test5) × 🍥 User.java × 🏭 User.hbm.xml × 🍥 UserDao.java × 🟭 hibernate.cfg.xml × 🛝 console × 🗸 Dat
         √ 📭 main
                                                  package org.tty.test5.dao;
                                                                                                                                                   A4 ^ v +
          ∨ 📄 java
                                                                                                                                                            ~

∨ org.tty.test5

                                                  jimport org.hibernate.Session;
                V 🛅 dao
                                                  import org.tty.test5.model.User;
                     C Session Itil
                   © SessionUtil
© UserDao
                                                 import java.util.List;
                  <u>□</u> model
                     C User
                                                  public class UserDao extends SessionUtil {

    HelloApplication

    HelloResource

                                           9
                                                      public List<User> findByUsernameAndPassword(String username, String password) {
                                                           return SessionUtil.query(
                  HelloServlet
                                                                      hql: "from User as user where username = '" + username + "' and password =

∨ Image: resources

               META-INF

∨ Im org.tty.test5.model

                                                 a }
                  # User.hbm.xml
                ahibernate.cfg.xml
                                                     public void save(User user) {
          webapp
                                                           SessionUtil.save(user);

✓ ■ WEB-INF

                                          18
                                                  }
        index.jsp
        gradlew bat
```

## 3.2: 编写注册界面

register.jsp

```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help test5 - register.jsp [test5.main]
  \textbf{test5} \; \rangle \; \mathsf{src} \; \rangle \; \mathbf{main} \; \rangle \; \mathsf{webapp} \; \rangle \; \underset{\mathbf{ss}}{\mathbf{ds}} \; \mathsf{register.jsp}
                                                                                                                                                                                                                                                                                                                                                               🎩 ▼ 🐧 Tomcat 9.0.38 🔻
Project ▼

> Image gradle

> Image gradle

> Image gradle
                                                          🚷 🚡 🛣 | 💠 — | d.gradle (test5) × © User.java × 🏭 User.hbm.xml × © UserDao.java × © RegisterServlet.java × 📠 register.jsp × 🔻 Dat
                > 🖿 gradle
                                                                                                                                        <%@ page contentType="text/html;charset=UTF-8" language="java" %>
                                                                                                                                         <html>
                     ∨ 📭 main
                                                                                                                                                                                                                                                                                                                                      0 0 0 0 0 0 0 0
                                                                                                                                      <head>
                                                                                                                                                    <title>注册界面</title>

∨ □ orq.tty.test5

                                                                                                                                      </head>
                                           V 🛅 dao
                                                                                                                                      =<body>
                                                          © SessionUtil
                                                                                                                                                  <form action="${pageContext.request.contextPath}/register" method="post">
                                                         UserDao
                                                                                                                                                           <label>用户名: <input type="text" name="username"/></label>
                                             ∨ 🛅 model
                                                                                                                                                                <label>密码: <input type="password" name="password"/></label>
                                                         © User
                                                                                                                   10
                                                                                                                                                                  <input type="submit" value="注册"/>
                                                   HelloApplication

    HelloResource

    HelloServlet

                                                                                                                                        </body>
                                                  © RegisterServlet
                                                                                                                                         </html>

√ Image resources

                                     > META-INF
                                     ∨ ■ org.tty.test5.model
                                                   User.hbm.xml
                                            ahibernate.cfg.xml

√ Image: ✓ webapp

✓ 

✓ ■ WEB-INF

                                                   🚜 web.xml
                                             index.jsp
                                        == register.jsp
                      build.gradle
        Services
       金豆类 共飞用 七
                                                                                                                                                                            Server Tomcat Localhost Log × Tomcat Catalina Log
```

RegisterServlet 的逻辑

```
|Project v ② 〒 ☆ - | | build-gradie (test) x © User-java v & User-java v & User-java v solvenskip v solvenski
                                                                                                                                                                                                                                                                                                                                                           grade(texts) ** Userjana*** Userhammal ** User Import javas.servlet.ServletException; Import javas.servlet.annotation.##@Servlet; Import javas.servlet.http.HttpServletRequest; Import javas.servlet.http.HttpServletRequest; Import javas.servlet.http.HttpServletResponse; Import javas.io.10Exception;
Import java.io.10Exception;
Import java.io.PrintWriter;
            .gradle
.idea
build
gradle
src
                                                                                                                                                                                                                                                                                 import java.sevret.http.nttpservletresponse;
import java.io.PrintWriter;

| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.PrintWriter;
| import java.io.Print
                                                                      ijava

ij
                                                                                                                                                                                                                                                                                                                                                                                  80veride
protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
    req.getRequestDispatcher(pubm:*/register.jsp*).formard(req, resp);
                                                                                                                                                                                                                                                                                                                                                                                               @Override
protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
    req.setCharacterEncoding("utf-8");
    resp.setCharacterEncoding("utf-8");
                                                                                                © RegisterServlet 24
sources 25 of @
                                                         resources

META-INF

org.tty.test5.model

User.hbm.xml
                                                                                                                                                                                                                                                                                                                                                                                                            String username = req.getParameter( name: "username");
String password =req.getParameter( name: "password");
                                                                                                                                                                                                                                                                                                                                                                                                            User user = new User();
user.setUsername(username);
user.setPassword(password);
   result.jsp

light test
light build.gradle
light gradlew.bat
light settings.gradle
light testS.iml
light External Libraries
Scratches and Consoles
                                                                                                                                                                                                                                                                                                                                                                                                        Session session = SessionUtil.getSession();
                                                                                                                                                                                                                                                                                                                                                                                                                                            Transaction transaction = session.beginTransaction();
                                                                                                                                                                                                                                                                                                                                                                                                                                            String result =
```

## 3.3: 编写登录界面

#### login.jsp

```
<head>
                                      <title>登录界面</title>
                                  <body>
                                     boy>
cform action="${pageContext.request.contextPath}/login" method="post">
<label>用户名、<input type="text" name="username"/></label>
<abel>密码、<input type="password" name="password"/></label>
<input type="submit" value="音景"/>
</form>
                                  </body>
                                   </html>
```

### LoginServlet

```
jęct v 💮 🕃 😤 🗘 🛑 🕲 Userjava 🔏 Userhom.xml x 🌑 Registerservétjava x 🚅 loginjap x 🌑 Loginservétjava x 👚 user x 🚔 registerjap x 🚅 resultjap x 🚅 indexjap x 🚅 indexjap x 🚅 indexjap x
                                                                            wa wegether wegether wegether wegether wegether import javax.servlet.annotain.lebServlet; import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletRequest; import javax.servlet.http.HttpServletResponse; import java.io.10Exception;
       in java

v in org.tty.test5

v in dao

s SessionUtil

s UserDao

model

t User

HelloApplication

HelloApsource

HelloResource

HelloResource

HelloResource

MEPS

Register-Servlet

tesources
                                                                                    @Override
protected void doGet(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
    req.getRequestDispatcher(palls:"/login.jsp").forward(req, resp);
                                                           15 0 0
                                                                                    protected void doPost(HttpServletRequest req, HttpServletResponse resp) throws ServletException, IOException {
    String username = req.getParameter(nime: "username");
    String passored = req.getParameter(nime: "passored");
                                                             28 1 0
          resources

META-INF

org.tty.test5.mode

User.hbm.xml

hibernate.cfg.xml
                                                                                            UserDao userDao = new UserDao();
String result = "";
                                                                                           if (userDao.find8yUsernameAndPassword(username, password).isEmpty()) {
    result = "Operation Failed";
} else {
    result = "Hello." + username;
}
                                                                                            req.setAttribute( name: "result", result);
req.getRequestDispatcher( path: "/result.jsp").forward(req, resp);
```

# 3.4: 进行 JMeter 进行压力测试

### 1、添加用户变量配置文件



### 其中内容如下:

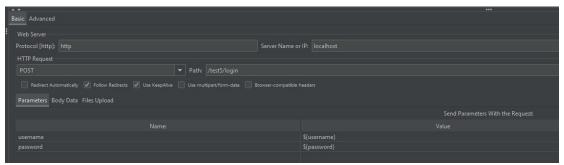




| CSV 数据文件设置           |                        |  |
|----------------------|------------------------|--|
| Name:                |                        |  |
|                      |                        |  |
| Comments:            |                        |  |
| 设置 CSV 数据文件          |                        |  |
| 200 20 1 2000        | 文件名:                   | C:/Users/cht/OneDrive/Workspace/课程/大三下/软件质量保证/实验/5/username.md |
|                      | 文件编码:                  | UTF-8  |
|                      | 变量名称(西文逗号间隔):          |  |
| 忽略首行(只在设置了变量名称后才生效): |                        | False  |
|                      | 分隔符 (用'\t'代替制表符):      |  |
|                      | 是否允许带引号?:              | False  |
|                      | 遇到文件结束符再次循环?:          |  |
|                      | 遇到文件结束符停止线程 <b>?</b> : | False  |
|                      | 线程共享模式:                | 所有现场   |
|                      |                        |  |

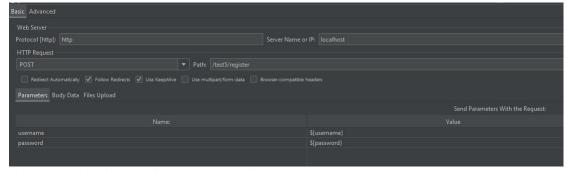
然后在线程组内添加 CSV 的数据文件配置,将刚才创建的两个变量配置上去。

2、添加对应的 Login 和 Register 的 HttpRequest 和 HttpResponse



其中登录的接口是/test5/login,使用 post 的方法进行传参,其中用户名和密码使用变量的方式从.md 文件中读取。

然后以同样的方式创建注册的 HttpRequest, 配置方式和上面的类似。

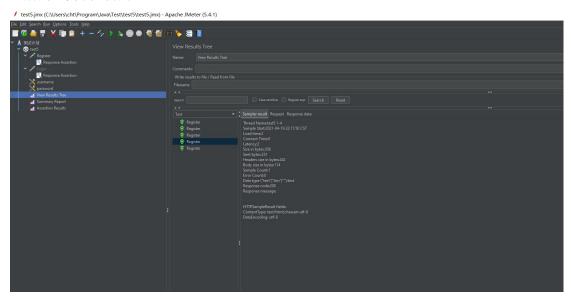


修改配置,线程数设置为5,递增量设为1,并进行测试。

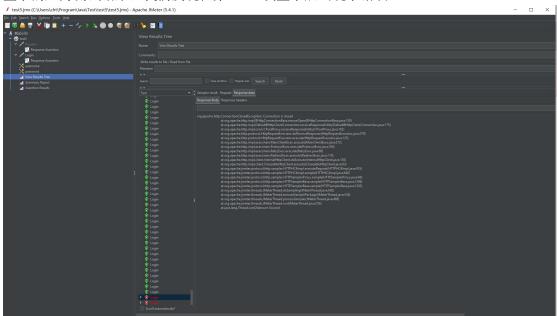


# 3.5: 测试结果分析

注册的压力测试结果

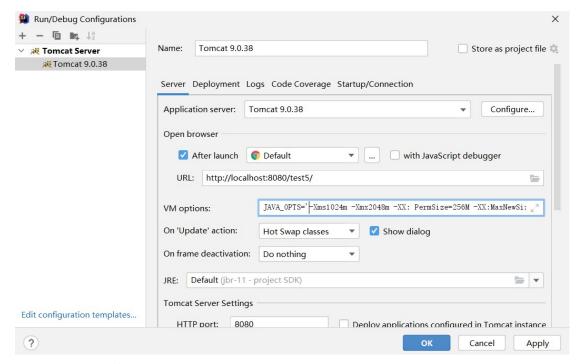


登录的压力测试结果,我们发现在第1007次登录后出现了错误。



# 3.6: 性能优化

我们尝试给 Tomcat 启动指定参数来扩大其启动的内存。



然后重新进行压力测试。



我们能够发现,相对优化前,流量提高了约120%,相对来说这是一次比较好的改善。

### 4、总结与分析

在进行压力测试中,目前发现性能的瓶颈出现在数据库连接中。常常出现的问题有:

- 1、数据库已经断开连接。
- 2、数据库连接超时,因为同时过多的连接而拒绝连接。

相对于内存来说,数据库的压力会更大。当同时出现多个用户访问时,如果没有好的并发策略,非常容易出现长时间无响应,或者更严重的数据库崩溃情况。

一种常见的解决方式是使用连接池的方式进行管理,当然,连接池也会因为无法启用更多的连接导致崩溃。这个时候就需要进行流量限制,并使用数据库长连接来减少创建数据库连接的资源消耗。

同时,也可以使用流量屏蔽技术对大流量访问的 IP 进行屏蔽以减小大流量带来的冲击。