|  |  |
| --- | --- |
| 实验名称： | 实验6：PHP综合实验 |
| 班级： | 信息安全224 |
| 学号： | 2022132003 |

1. **实验过程：**

（实验的具体过程，可以分成两个部分，第一个部分写思路，第二个部分写具体的实现代码。）

1. 实验思路：

(1)需求分析：首先，确定博客系统的基本功能需求，如用户注册与登录、文章发布与浏览、评论管理、标签管理等。

(2)设计数据库结构：根据功能需求设计数据库表结构，如用户表、文章表、评论表、标签表等，并确定它们之间的关系。根据功能需求设计数据库表结构。创建数据库和表，并设置主键、外键等约束条件。写数据库操作脚本（如创建表、插入数据等）。

(3)编写PHP代码：使用PHP编写后端逻辑，包括与数据库的交互、用户认证、数据处理等。创建PHP文件，用于处理用户请求和数据交互。连接MySQL数据库，并编写数据库查询、插入、更新和删除等操作的PHP代码。现用户注册与登录功能，包括表单验证、密码加密等。实现文章发布与浏览功能，包括文章列表显示、文章详情显示等。实现评论管理功能，包括评论的添加、显示和删除等。

(4)创建前端界面：使用HTML、CSS和JavaScript（可选）创建博客系统的前端界面，如登录页、注册页、文章列表页、文章详情页等。

(5)集成前后端：将前端界面与后端逻辑进行集成，实现数据的动态显示和交互。使用HTML编写基本页面结构、CSS对页面进行样式设计.编写表单用于用户输入数据，如注册表单、登录表单、评论表单等。在前端页面中嵌入PHP代码，实现数据的动态显示和交互。

(6)测试与调试：对博客系统进行测试，包括功能测试、性能测试和安全性测试等。使用调试工具查找并修复潜在的问题和错误。

(7)优化与扩展：对系统进行性能优化和功能扩展，如添加搜索功能、使用缓存技术等。

1. 代码实现：

重要代码片段说明：

**register.php**（允许注册，注册名需要判断是否已经存在，如果存在，进行提示。）

<?php

session\_start();

include('config.php');

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

    $username = $\_POST['username'];

    $realname = $\_POST['realname'];

    $password = password\_hash($\_POST['password'], PASSWORD\_BCRYPT);

    $checkUserQuery = "SELECT \* FROM users WHERE username='$username'";

    $result = $conn->query($checkUserQuery);

    if ($result->num\_rows > 0) {

        echo "用户名已存在，请选择其他用户名。";

    } else {

        $insertQuery = "INSERT INTO users (username, realname, password) VALUES ('$username', '$realname', '$password')";

        if ($conn->query($insertQuery) === TRUE) {

            echo "注册成功！";

        } else {

            echo "注册失败: " . $conn->error;

        }

    }

}

?>

<!DOCTYPE html>

<html>

<head>

    <title>注册</title>

</head>

<body>

    <h2>注册</h2>

    <form method="post">

        用户名: <input type="text" name="username" required><br>

        真实姓名: <input type="text" name="realname" required><br>

        密码: <input type="password" name="password" required><br>

        <input type="submit" value="注册">

    </form>

    <a href="index.php">登录</a>

</body>

</html>

index.php（登录页面）

<?php

session\_start();

include('config.php');

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

    $username = $\_POST['username'];

    $password = $\_POST['password'];

    $query = "SELECT \* FROM users WHERE username='$username'";

    $result = $conn->query($query);

    if ($result->num\_rows == 1) {

        $user = $result->fetch\_assoc();

        if (password\_verify($password, $user['password'])) {

            $\_SESSION['user\_id'] = $user['id'];

            header("Location: personal.php");

            exit();

        } else {

            echo "密码错误。";

        }

    } else {

        echo "用户不存在。";

    }

}

?>

<!DOCTYPE html>

<html>

<head>

    <title>登录</title>

</head>

<body>

    <h2>登录</h2>

    <form method="post">

        用户名: <input type="text" name="username" required><br>

        密码: <input type="password" name="password" required><br>

        <input type="submit" value="登录">

    </form>

    <a href="register.php">注册</a>

</body>

</html>

}

?>

personal.php（可以修改个人界面）

<?php

session\_start();

include('config.php');

if (!isset($\_SESSION['user\_id'])) {

    header("Location: index.php");

    exit();

}

$user\_id = $\_SESSION['user\_id'];

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

    $realname = $\_POST['realname'];

    $password = password\_hash($\_POST['password'], PASSWORD\_BCRYPT);

    $updateQuery = "UPDATE users SET realname='$realname', password='$password' WHERE id='$user\_id'";

    if ($conn->query($updateQuery) === TRUE) {

        echo "个人信息更新成功！";

    } else {

        echo "更新失败: " . $conn->error;

    }

}

$query = "SELECT \* FROM users WHERE id='$user\_id'";

$result = $conn->query($query);

$user = $result->fetch\_assoc();

?>

<!DOCTYPE html>

<html>

<head>

    <title>个人信息</title>

</head>

<body>

    <h2>个人信息</h2>

    <form method="post">

        用户名: <input type="text" name="username" value="<?php echo $user['username']; ?>" readonly><br>

        真实姓名: <input type="text" name="realname" value="<?php echo $user['realname']; ?>" required><br>

        密码: <input type="password" name="password" required><br>

        <input type="submit" value="更新">

    </form>

    <a href="new\_article.php">发布博客</a>

    <a href="article\_list.php">查看博客列表</a>

    <a href="logout.php">退出</a>

</body>

</html>

}

?>

new\_article.php（发布博客）

<?php

session\_start();

include('config.php');

if (!isset($\_SESSION['user\_id'])) {

    header("Location: index.php");

    exit();

}

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

    $title = $\_POST['title'];

    $content = $\_POST['content'];

    $user\_id = $\_SESSION['user\_id'];

    $insertQuery = "INSERT INTO articles (user\_id, title, content) VALUES ('$user\_id', '$title', '$content')";

    if ($conn->query($insertQuery) === TRUE) {

        echo "博客发布成功！";

    } else {

        echo "发布失败: " . $conn->error;

    }

}

?>

<!DOCTYPE html>

<html>

<head>

    <title>发布博客</title>

</head>

<body>

    <h2>发布博客</h2>

    <form method="post">

        标题: <input type="text" name="title" required><br>

        内容: <textarea name="content" required></textarea><br>

        <input type="submit" value="发布">

    </form>

    <a href="article\_list.php">查看博客列表</a>

    <a href="personal.php">个人信息</a>

</body>

</html>

// 检查文章标题和内容是否为空

if (empty($\_POST['title']) || empty($\_POST['content'])) {

echo "文章标题和内容不能为空。";

}

?>

view\_article.php（可以查看一篇博客）

<?php

session\_start();

include('config.php');

if (!isset($\_SESSION['user\_id'])) {

    header("Location: index.php");

    exit();

}

$article\_id = $\_GET['id'];

$query = "SELECT \* FROM articles WHERE id='$article\_id'";

$result = $conn->query($query);

$article = $result->fetch\_assoc();

?>

<!DOCTYPE html>

<html>

<head>

    <title>查看博客</title>

</head>

<body>

    <h2><?php echo $article['title']; ?></h2>

    <p><?php echo nl2br($article['content']); ?></p>

    <a href="article\_list.php">返回博客列表</a>

</body>

</html>

article\_list.php （以表格方式查看个人博客的文章，需要进行分页。可以删除、查看）

<?php

session\_start();

include('config.php');

if (!isset($\_SESSION['user\_id'])) {

    header("Location: index.php");

    exit();

}

$user\_id = $\_SESSION['user\_id'];

$page = isset($\_GET['page']) ? $\_GET['page'] : 1;

$limit = 5;

$offset = ($page - 1) \* $limit;

$query = "SELECT \* FROM articles WHERE user\_id='$user\_id' LIMIT $limit OFFSET $offset";

$result = $conn->query($query);

$totalQuery = "SELECT COUNT(\*) AS total FROM articles WHERE user\_id='$user\_id'";

$totalResult = $conn->query($totalQuery);

$total = $totalResult->fetch\_assoc()['total'];

$totalPages = ceil($total / $limit);

?>

<!DOCTYPE html>

<html>

<head>

    <title>博客列表</title>

</head>

<body>

    <h2>博客列表</h2>

    <table border="1">

        <tr>

            <th>标题</th>

            <th>操作</th>

        </tr>

        <?php while ($row = $result->fetch\_assoc()) { ?>

        <tr>

            <td><?php echo $row['title']; ?></td>

            <td>

                <a href="view\_article.php?id=<?php echo $row['id']; ?>">查看</a>

                <a href="delete\_article.php?id=<?php echo $row['id']; ?>">删除</a>

            </td>

        </tr>

        <?php } ?>

    </table>

    <div>

        <?php for ($i = 1; $i <= $totalPages; $i++) { ?>

            <a href="article\_list.php?page=<?php echo $i; ?>"><?php echo $i; ?></a>

        <?php } ?>

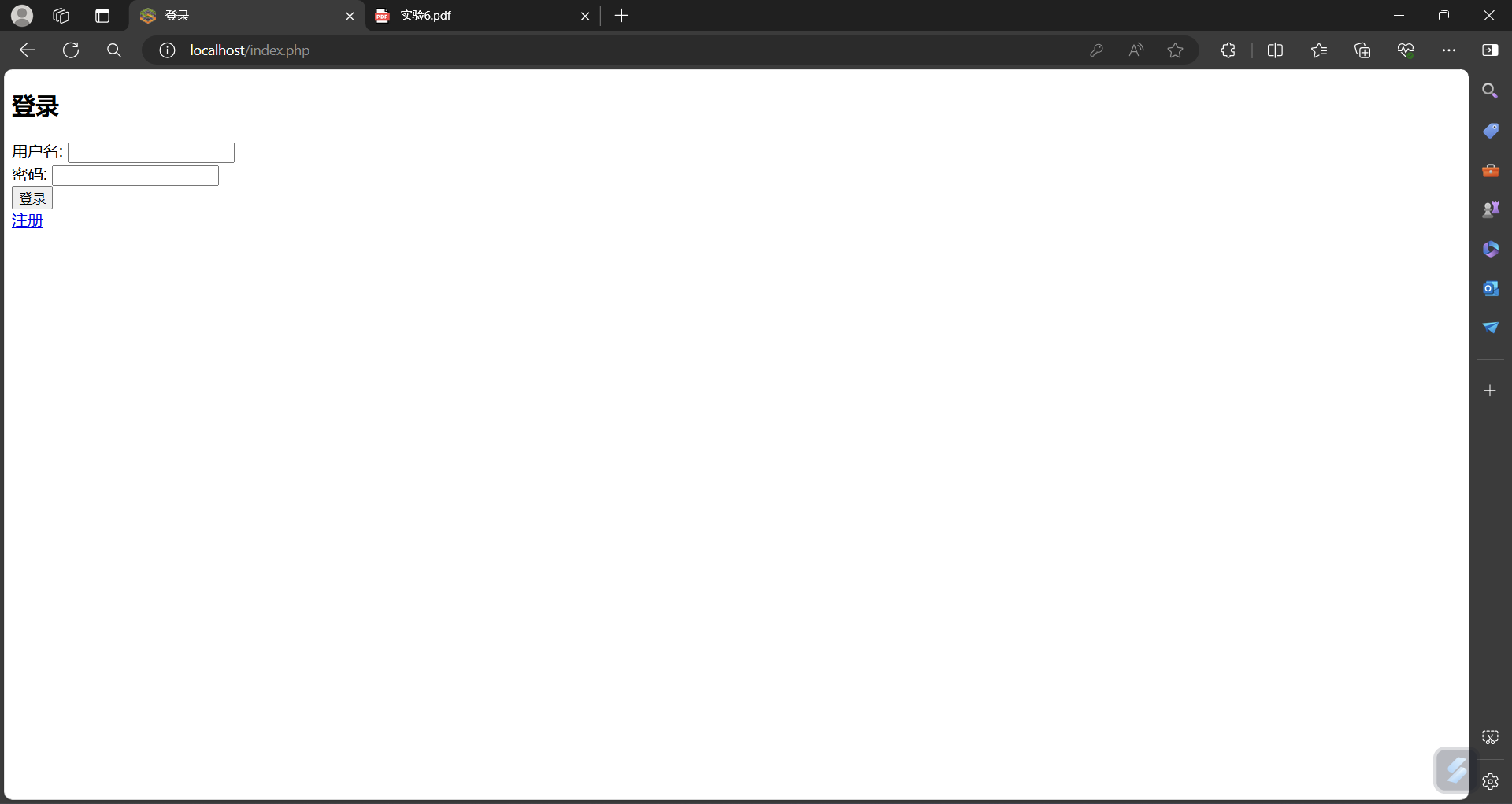
    </div>

    <a href="personal.php">个人信息</a>

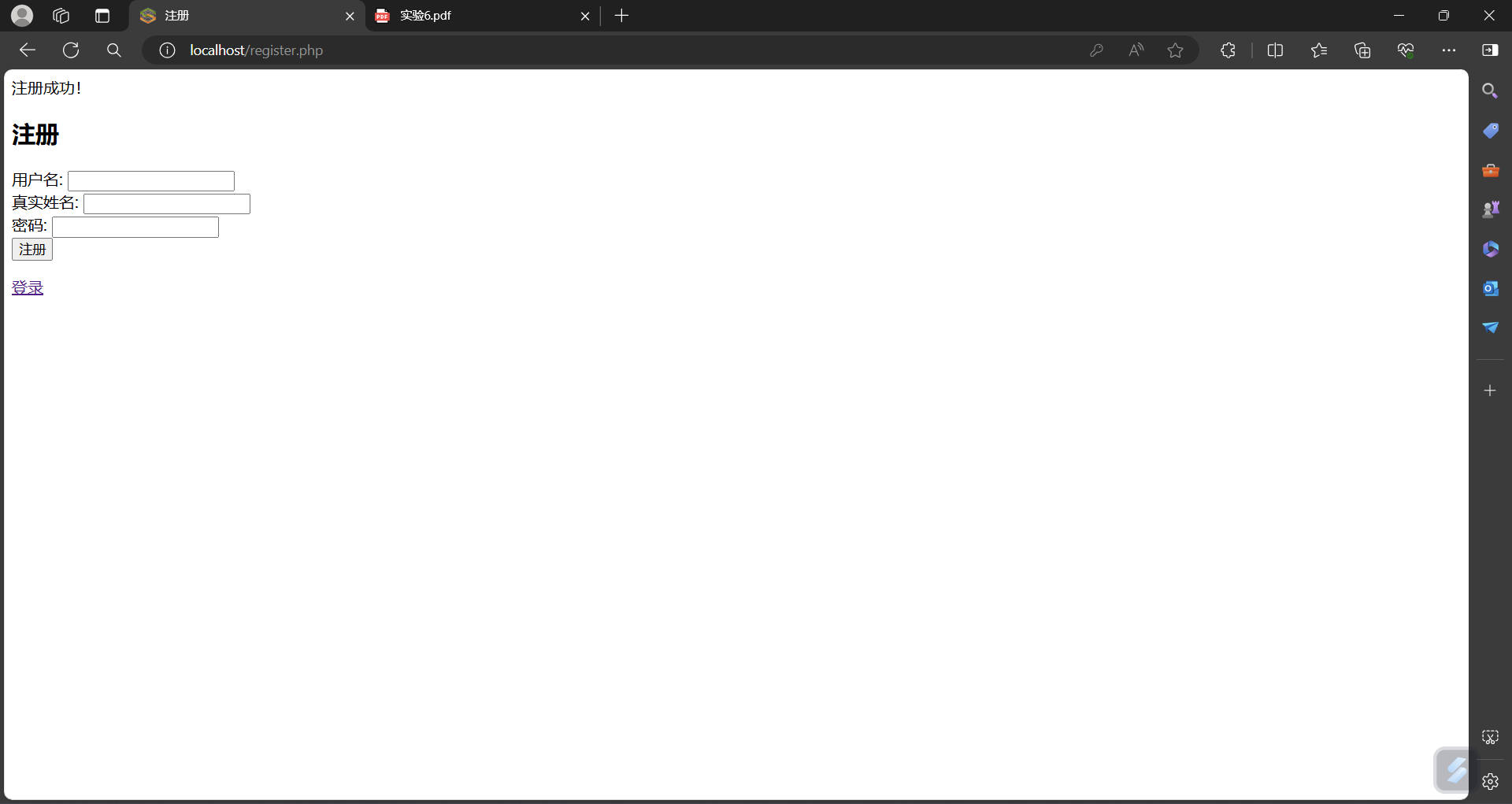
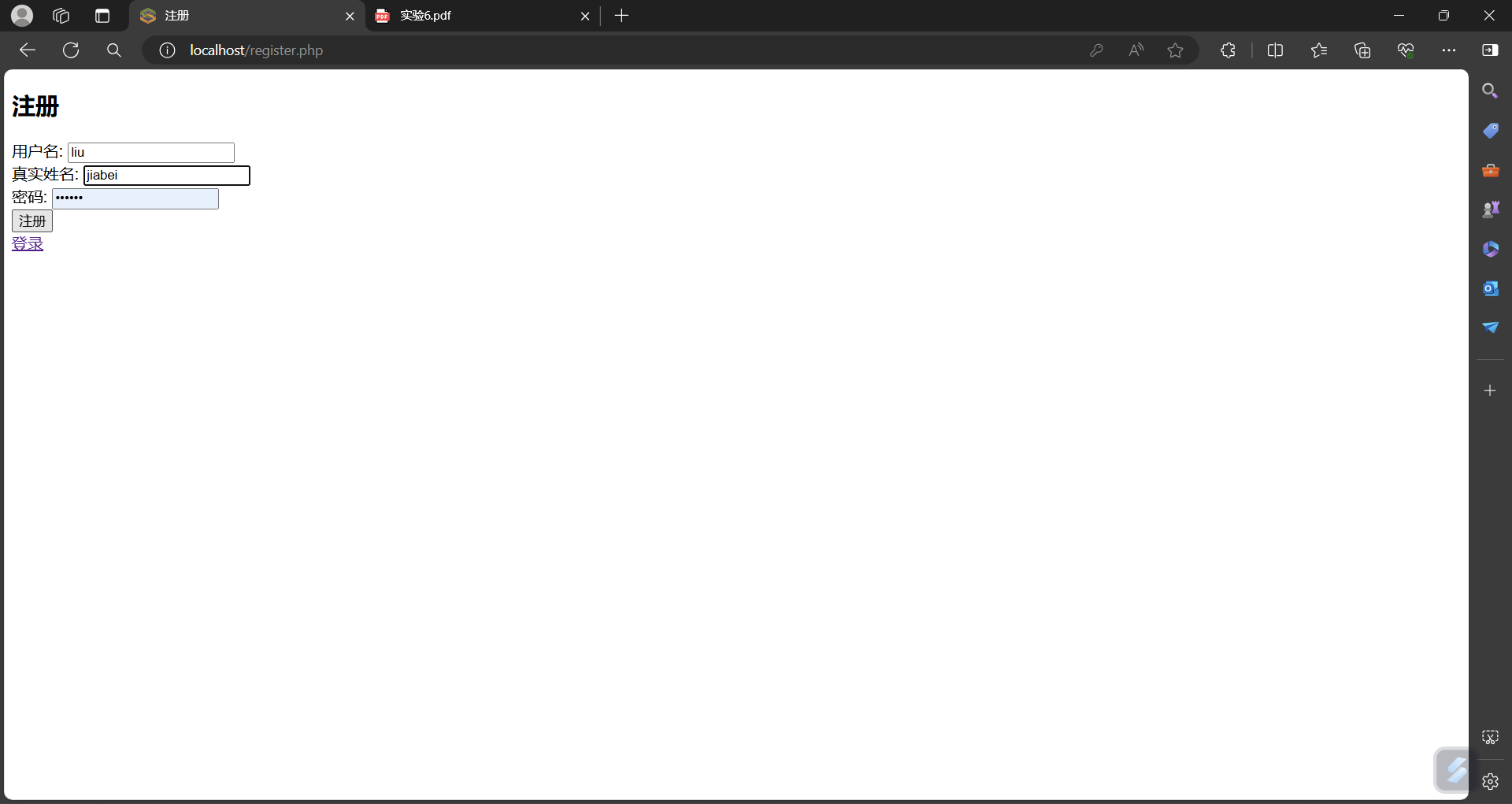
</body>

</html>

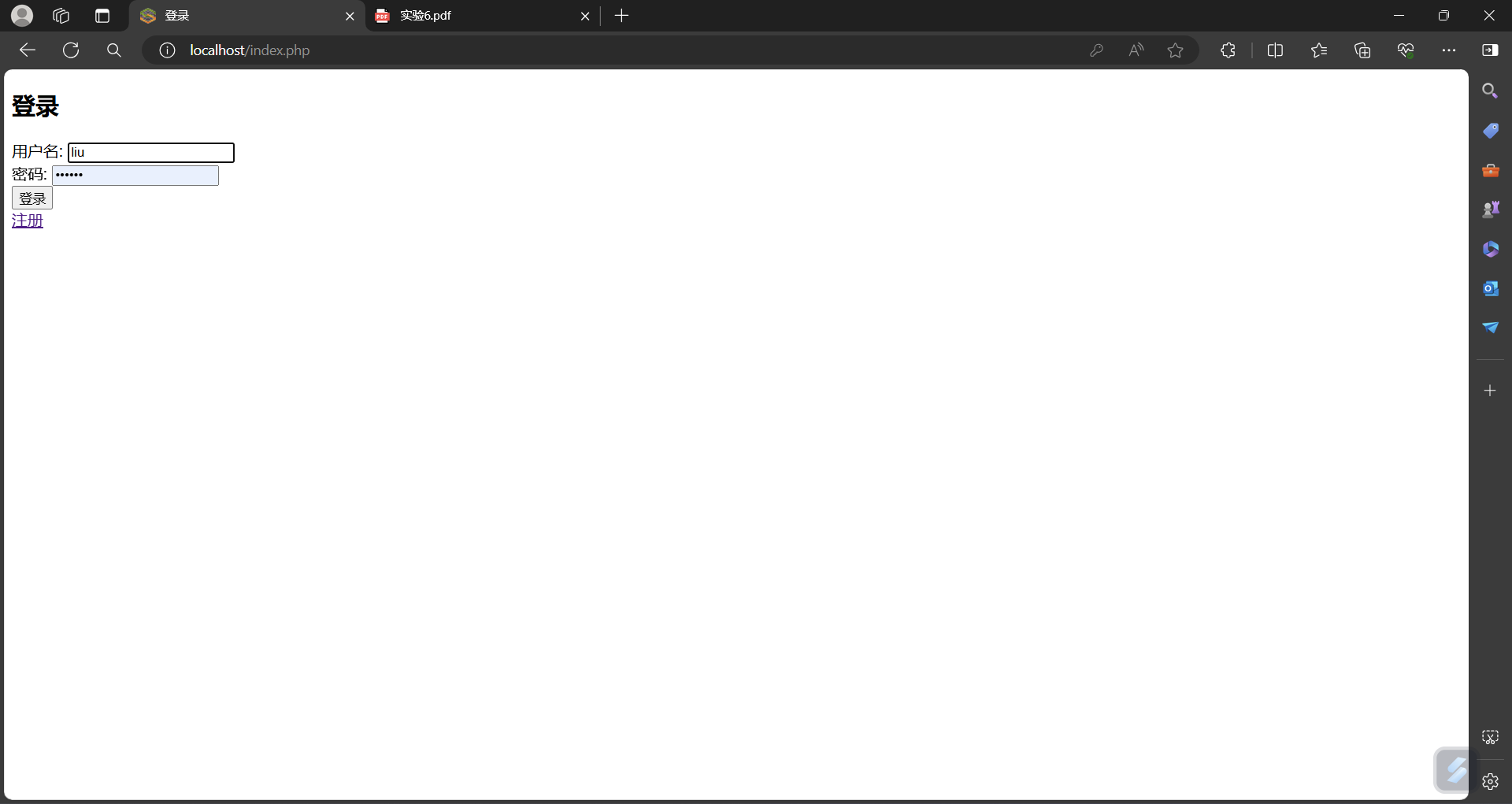
1. **实验结果**
2. 进入登录、注册界面



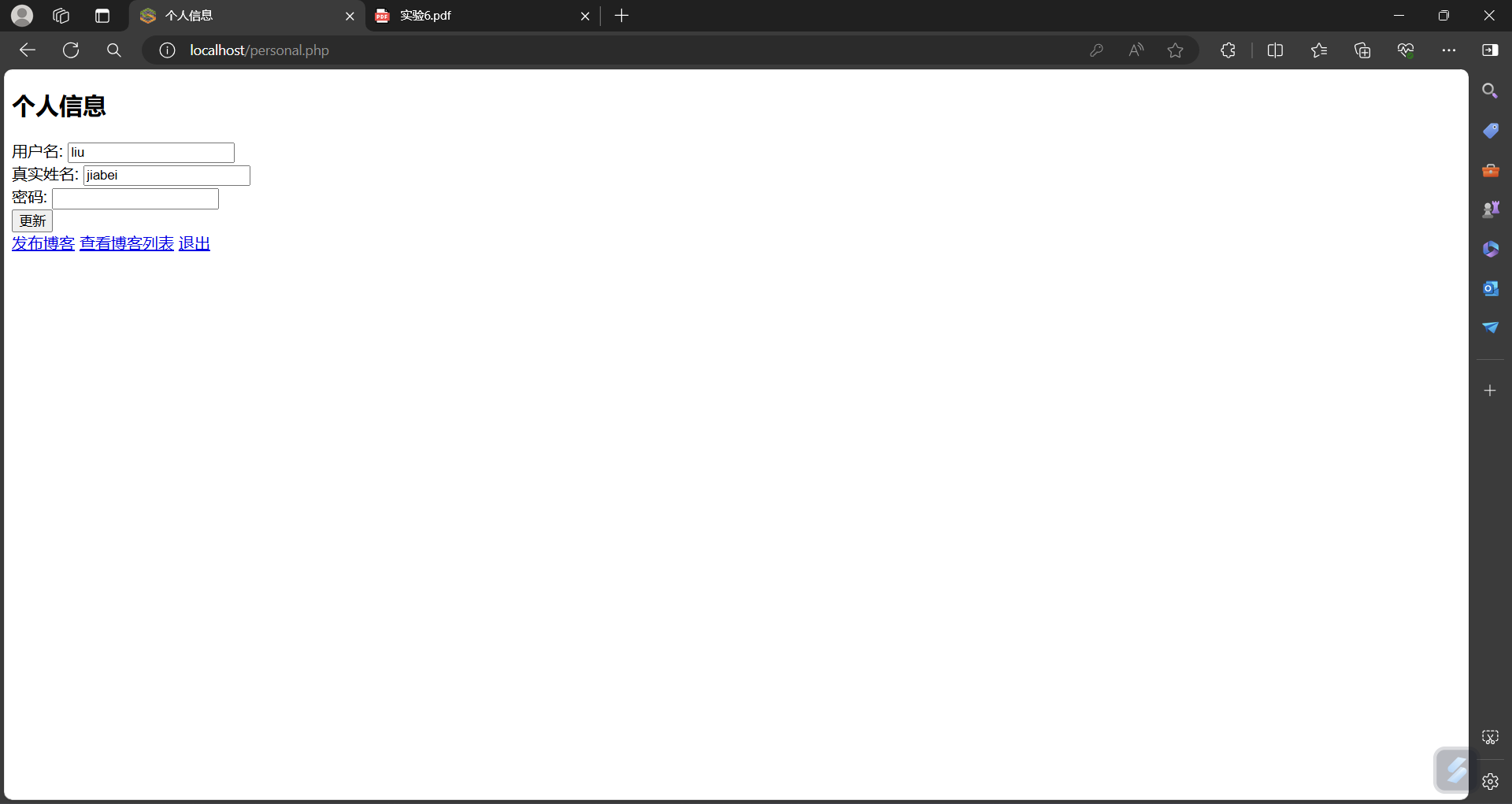
1. 进入注册界面



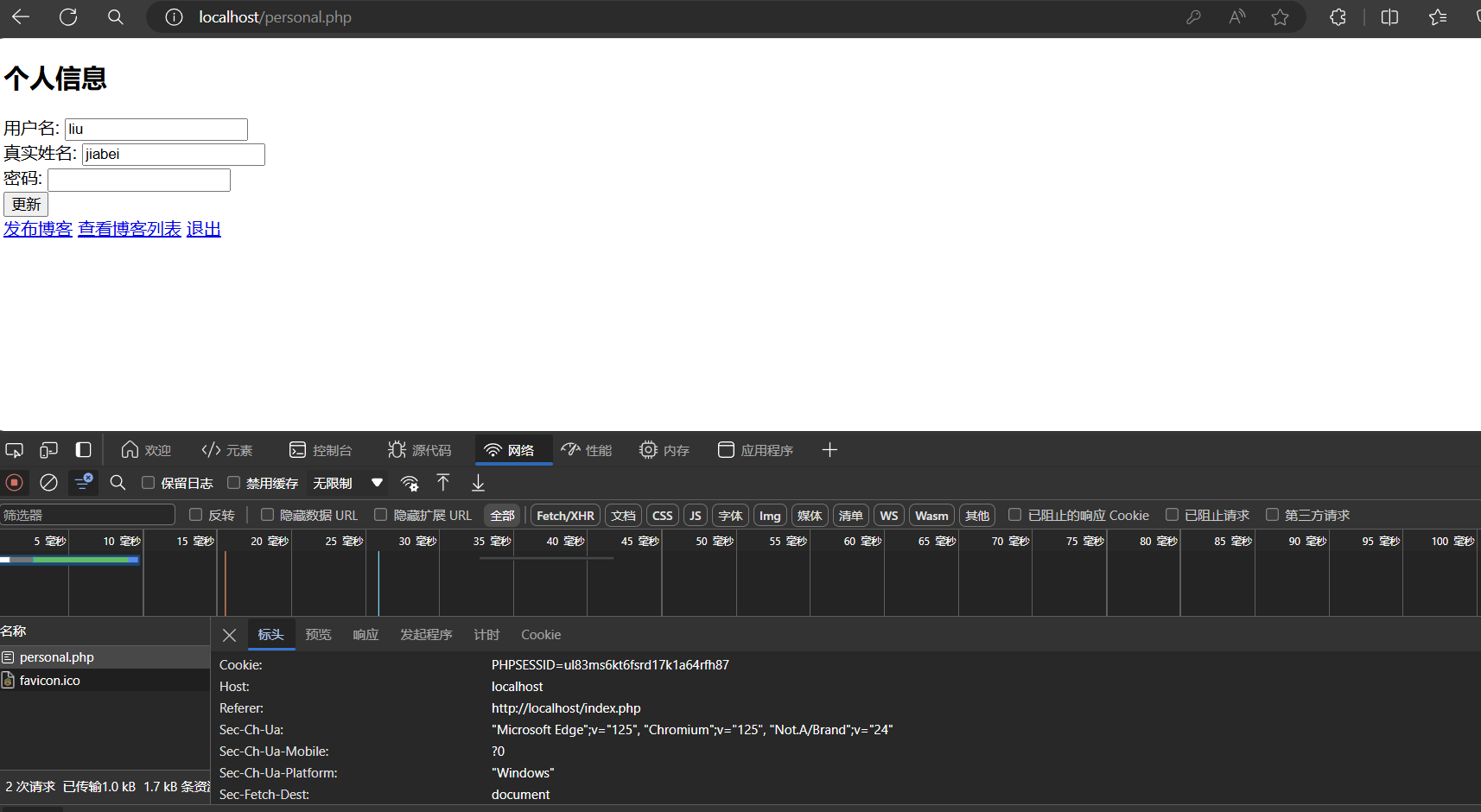
1. 用刚才注册好的信息进行登录



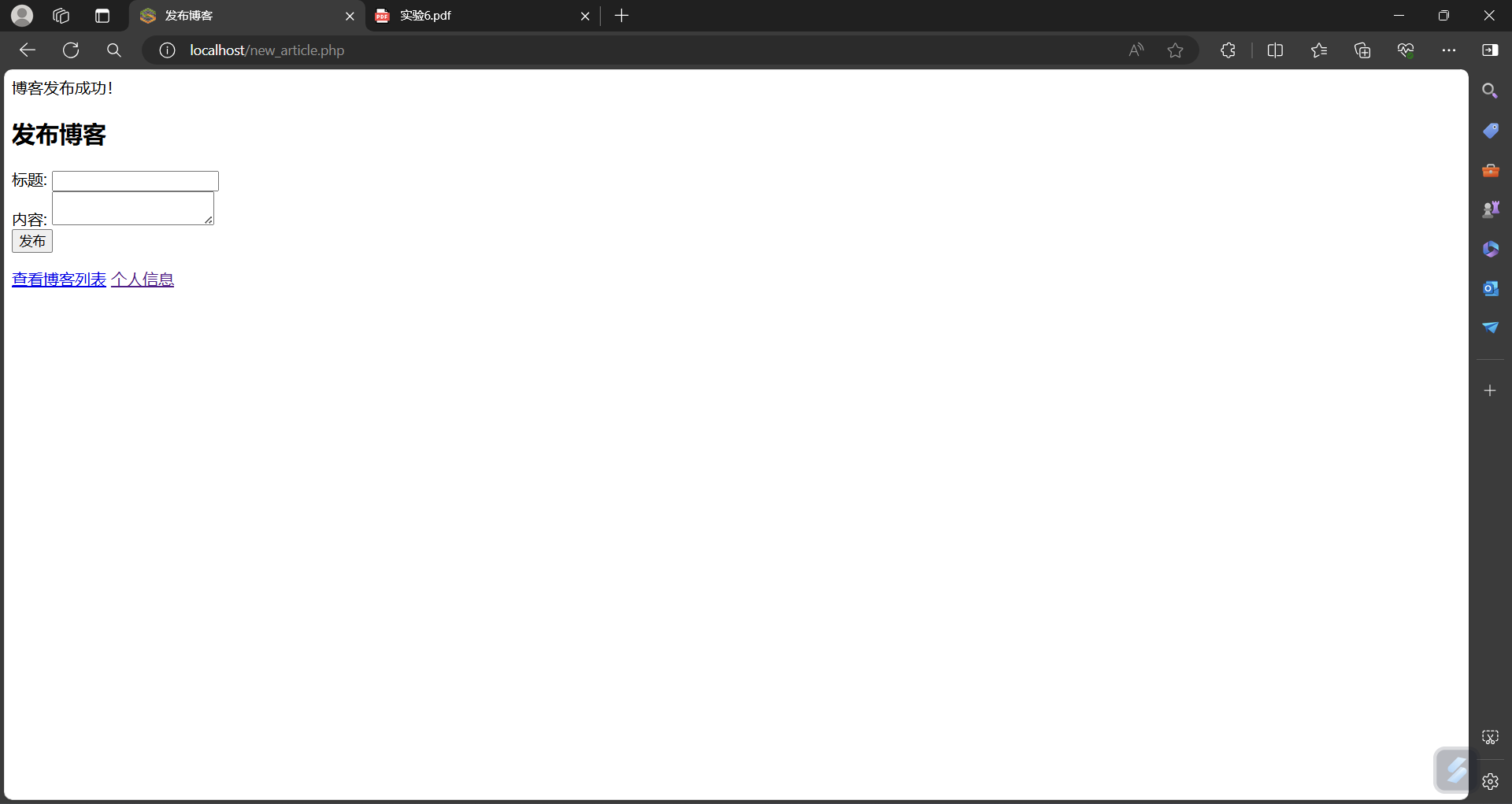
1. 进入个人信息页面

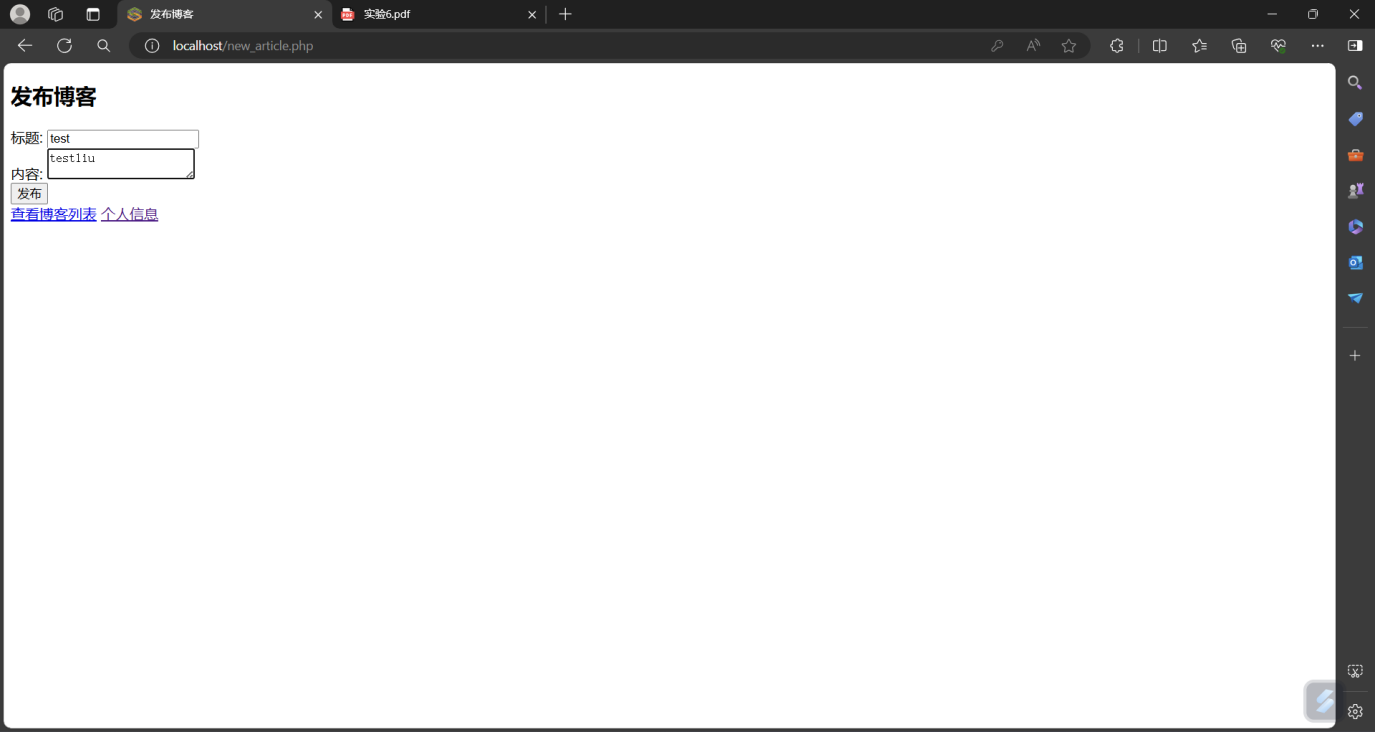


1. 查看页面网络，可以看见我们的cookie值

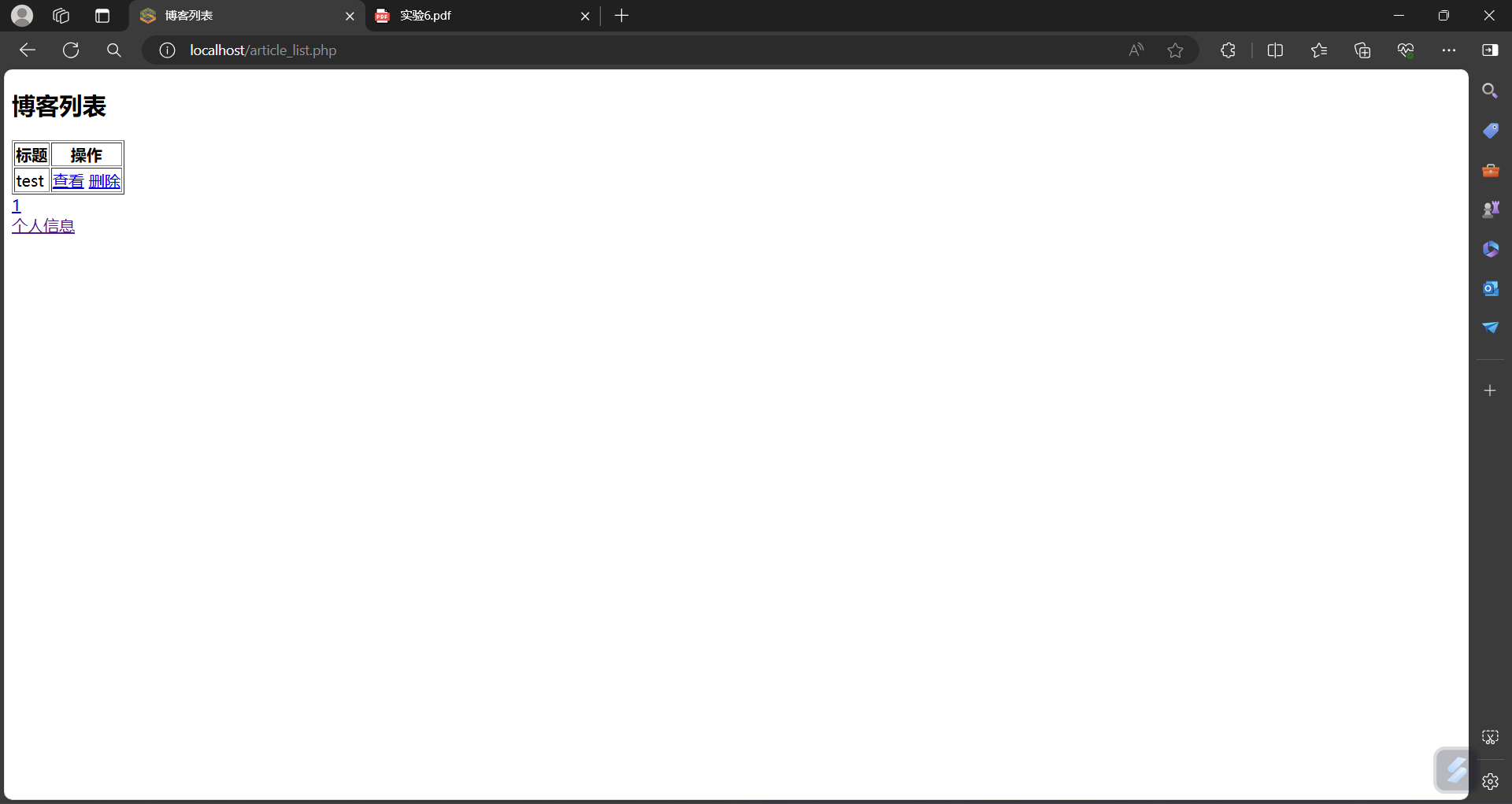


1. 发布博客

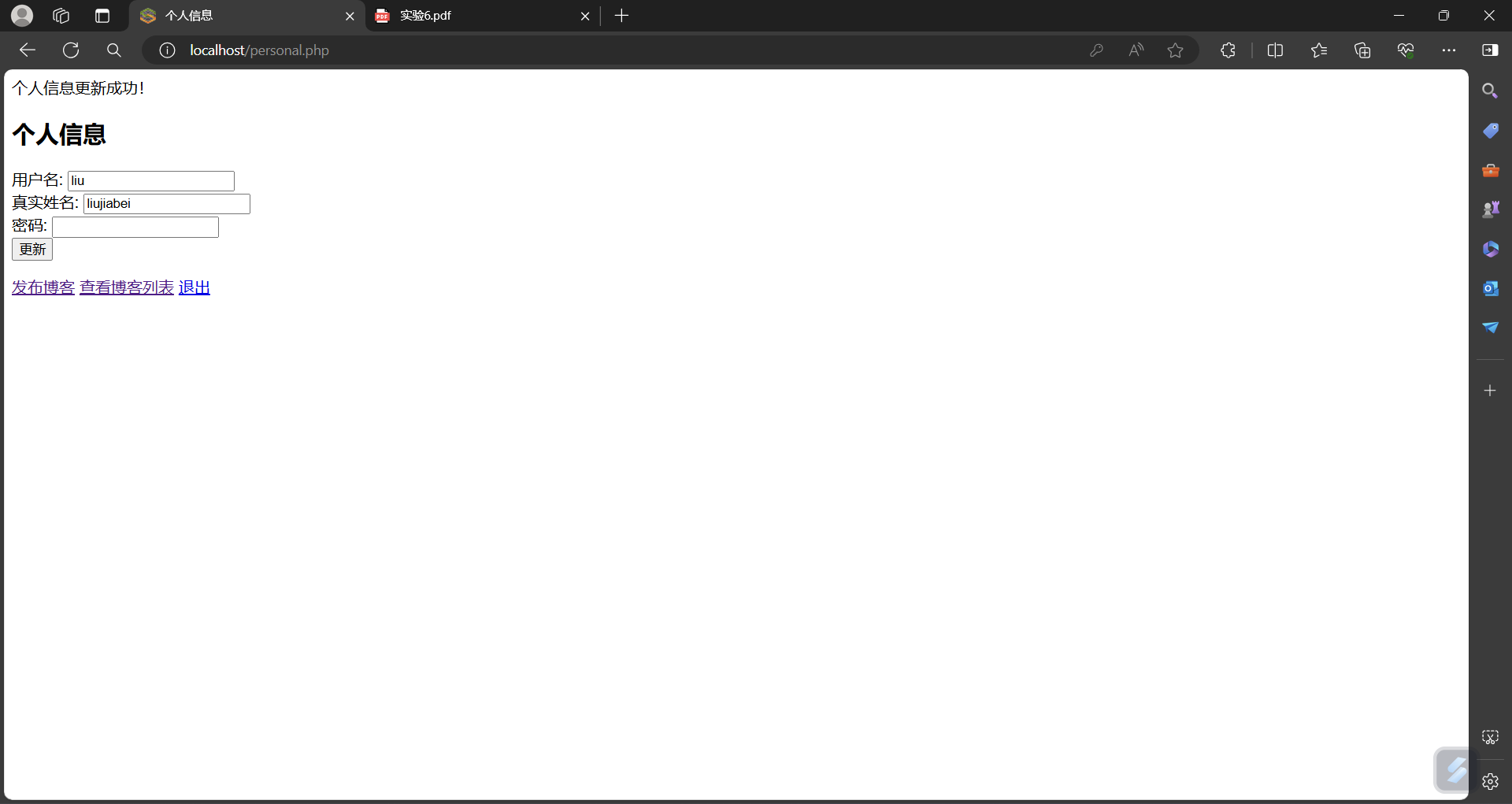




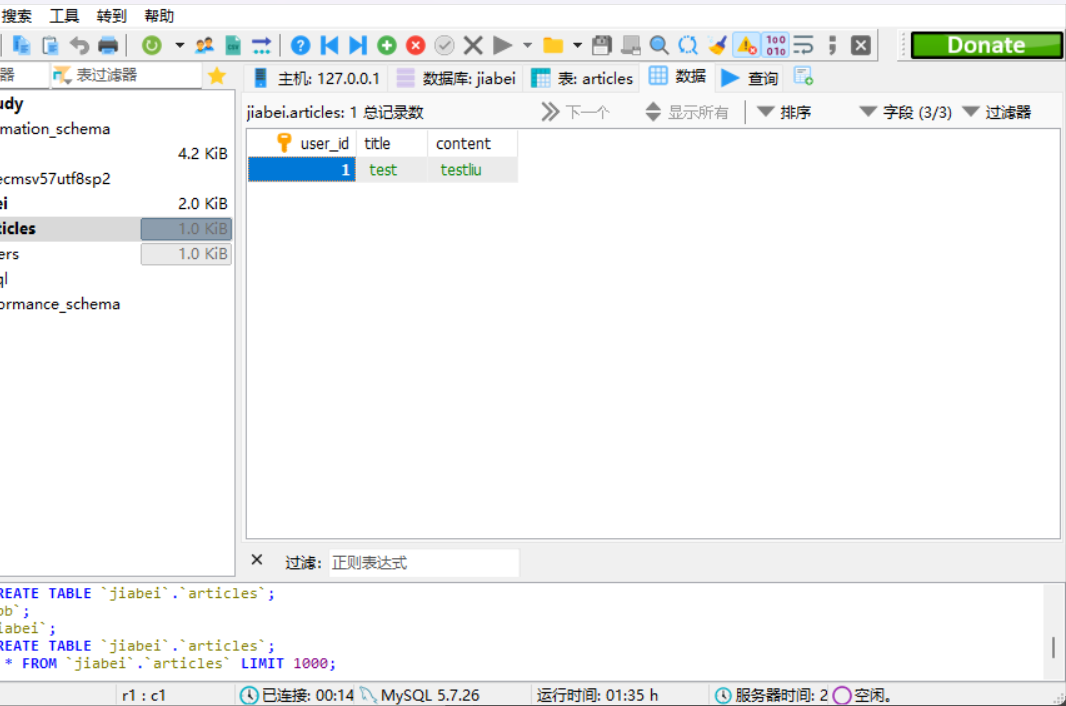
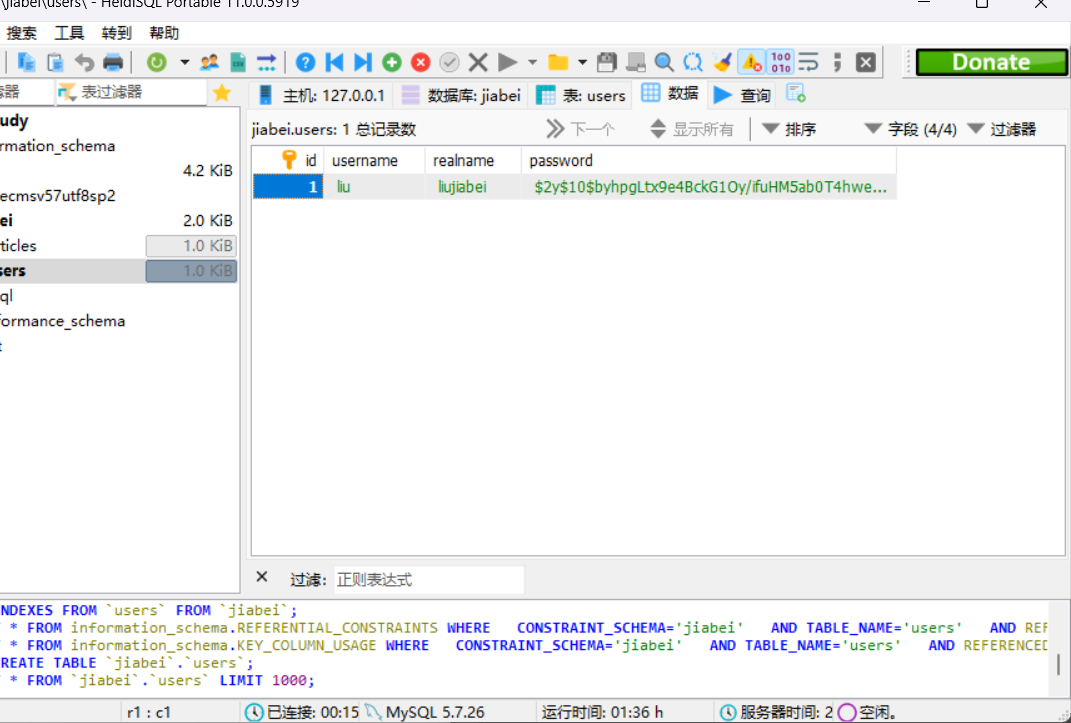
1. 查看博客列表



1. 更新个人用户信息



1. 查看本地数据库信息

****

成功查到新建用户和新建文章的信息。

1. **总结与思考**

在开始实验之前，我复习了PHP的基础语法、数据库操作（特别是MySQL）以及HTML/CSS前端技术。我意识到，这些基础知识是构建博客页面的基石。没有坚实的基础，很难在实验中取得进展。在实验中，我遇到了许多错误和问题。我学会了使用PHP的错误报告机制、调试工具（如Xdebug）以及查看日志文件来定位和解决问题。这些技能在后续的软件开发过程中将非常有用。

通过这次实验，我深刻认识到自己还有很多需要学习和改进的地方。例如，我可以进一步学习前端技术（如JavaScript框架、响应式设计等）来提高博客页面的交互性和美观度；我还可以学习更高级的安全技术来增强博客页面的安全性。总之，这次PHP创建博客页面的实验让我收获颇丰。我不仅巩固了基础知识，还提高了编程技能和解决问题的能力。同时，我也意识到持续学习和改进的重要性。在未来的学习和工作中，我将继续努力提高自己的技能水平，为更好地满足用户需求而努力。