

Evan

Only let oneself become strong enough, good enough, can afford the life that you want to.

☰ 目录视图

☰ 摘要视图

🔒 订阅专栏

从创业到再就业，浅述对程序员职业生涯的看法 征文 | 你会为 AI 转型么？ 赠书：7月大咖新书机器学习/Android/python

Android通过php插入查询SQL数据库

2015-11-09 16:24 257人阅读 评论(0) 收藏 举报

☰ 分类： 移动开发 (38) ▼

❗ 版权声明：本文为博主原创文章，未经博主允许不得转载。

PHP代码

```
<?php
header("Content-type: text/html; charset=gb2312");
$serverName = "localhost"; //数据库服务器地址
$uid = "Mssqla"; //数据库用户名
$pwd = "123456"; //数据库密码
$connectionInfo = array("UID"=>$uid, "PWD"=>$pwd, "Database"=>"mysql");
$conn = sqlsrv_connect( $serverName, $connectionInfo);
if($conn == false){
    echo "连接数据库失败";
}else{
    echo "连接数据库成功";
}
$name = $_POST['name'];
$pass = $_POST['pass'];
$security = $_POST['security'];
print($name);

sqlsrv_query($conn, "insert into android (name,pass,security) values('$name','$pass','$security')");

//定义好一个数组。PHP中array相当于一个数据字典
$data = array();
//定义一个类,用到存放从数据库中取出的数据.
class User{
    public $name;
    public $pass;
}
```

```

$query = sqlsrv_query($conn, "select * from myuser");
while($row = sqlsrv_fetch_array($query)){
    $user = new User();
    $user->name = $row['name'];
    $user->pass = $row['pass'];
    $data[] = $user;
}
$json = json_encode($data); //把数据转换为JSON数据
print("{\"user\":\"".$json."\"}");

```

```
sqlsrv_close($conn);
```

```
?>
```

Android代码

```

public static final int ISLOGIN =1;
private void sendRequestWithHttpClient(){
    new Thread(new Runnable() {
        @Override
        public void run() {
            //创建键/值组列表
            List<NameValuePair> pairs = new ArrayList<NameValuePair>();
            pairs.add(new BasicNameValuePair("name", qq_name.getText().toString()));
            pairs.add(new BasicNameValuePair("pass", qq_pass.getText().toString()));
            pairs.add(new BasicNameValuePair("security", security.getText().toString()));
            try{
                HttpClient httpClient = new DefaultHttpClient(); //定义HttpClient
                HttpPost httpPost = new HttpPost("http://...../android/registered.php"); //实例化HTTP方法
                UrlEncodedFormEntity entity = new UrlEncodedFormEntity(pairs,"gb2312");
                httpPost.setEntity(entity);
                HttpResponse httpResponse = httpClient.execute(httpPost);    //执行请求
                //处理返回数据
                HttpEntity httpentity =httpResponse.getEntity();
                InputStream is =httpentity.getContent();
                BufferedReader reader = new BufferedReader(
                    new InputStreamReader(is,"gb2312"));
                StringBuilder sb =new StringBuilder();
                sb.append(reader.readLine() + "\n");
                String line = "0";
                while((line =reader.readLine())!=null){

```

```
        sb.append(line + "\n");
    }
    is.close();
    result = sb.toString();
    Message message = new Message();
    message.what = ISLOGIN;
    handler.sendMessage(message); // 将Message对象发送出去,采用异步处理机制

    if(httpResponse.getStatusLine().getStatusCode() == HttpStatus.SC_OK){
        // String result = EntityUtils.toString(httpResponse.getEntity());
    }else{
    }
}
}catch(UnsupportedEncodingException e){
    e.printStackTrace();
}catch(ClientProtocolException e){
    e.printStackTrace();
}catch (IOException e) {
    e.printStackTrace();
}
}
}).start();
}
```

//异步处理的代码

@SuppressWarnings("HandlerLeak")

```
private Handler handler = new Handler() {
    public void handleMessage(Message msge) {
        switch (msge.what) {
            case ISLOGIN:
                // 在这里可以进行UI操作
                Toast.makeText(LoginActivity.this, "account or password is invalid",
                    Toast.LENGTH_SHORT).show();
                break;
            default:
                break;
        }
    }
};
```

注意：重新开启一个线程处理HttpClient请求，不要在UI线程处理，注意编码方式，如果出现乱码，**数据库**插入操作失败，将返回null值。