* [](https://www.csdn.net/)
* [首页](https://www.csdn.net/" \o "首页)
* [博客](https://blog.csdn.net/" \o "博客)
* [学院](https://edu.csdn.net/" \o "学院)
* [下载](https://download.csdn.net/" \o "下载)
* [论坛](https://bbs.csdn.net/" \o "论坛)
* [问答](https://ask.csdn.net/" \o "问答)
* [商城](https://h5.youzan.com/v2/showcase/homepage?alias=BUj3rrGa2J&ps=760" \o "商城" \t "/Users/liu123/Documents\\x/_blank)
* [活动](https://huiyi.csdn.net/" \o "活动)
* [专题](https://spec.csdn.net/" \o "专题)
* [招聘](http://job.csdn.net/" \o "招聘)
* [ITeye](http://www.iteye.com/" \o "ITeye" \t "/Users/liu123/Documents\\x/_blank)
* [GitChat](https://gitbook.cn/?ref=csdn" \o "GitChat)
* [APP](https://www.csdn.net/apps/download?code=pc_1555579859" \o "APP)
* [VIP会员](https://mall.csdn.net/vip" \o "VIP会员)续费8折

[](https://so.csdn.net/so/)

* [IMG_258](https://mp.csdn.net/postedit" \t "/Users/liu123/Documents\\x/_blank)[写博客](https://mp.csdn.net/postedit" \t "/Users/liu123/Documents\\x/_blank)
* [IMG_259](https://i.csdn.net/#/msg/index)
* [登录/注册](https://passport.csdn.net/account/login)

IMG_260

原创

# Spring-事务的传播属性（六）

2018-08-16 16:04:27 [piaoslowly](https://me.csdn.net/piaoslowly" \t "/Users/liu123/Documents\\x/_blank) 阅读数 3531更多

分类专栏： [spring](https://blog.csdn.net/piaoslowly/article/category/7943289" \t "/Users/liu123/Documents\\x/_blank)

版权声明：本文为博主原创文章，遵循[CC 4.0 BY-SA](http://creativecommons.org/licenses/by-sa/4.0/" \t "/Users/liu123/Documents\\x/_blank)版权协议，转载请附上原文出处链接和本声明。

本文链接：<https://blog.csdn.net/piaoslowly/article/details/81743658>

# **Spring-事务的传播属性（六）**

## **事务的传播属性概述**

所谓spring事务的传播属性，就是定义在存在多个事务同时存在的时候，spring应该如何处理这些事务的行为。这些属性在TransactionDefinition中定义，具体常量的解释见下：

* Propagation.REQUIRED（required）：支持当前事务，如果当前有事务， 那么加入事务， 如果当前没有事务则新建一个(默认情况)
* Propagation.NOT\_SUPPORTED（not\_supported) ： 以非事务方式执行操作，如果当前存在事务就把当前事务挂起，执行完后恢复事务（忽略当前事务）；
* Propagation.SUPPORTS (supports) ：如果当前有事务则加入，如果没有则不用事务。
* Propagation.MANDATORY (mandatory) ：支持当前事务，如果当前没有事务，则抛出异常。（当前必须有事务）
* PROPAGATION\_NEVER (never) ：以非事务方式执行，如果当前存在事务，则抛出异常。（当前必须不能有事务）
* Propagation.REQUIRES\_NEW (requires\_new) ：支持当前事务，如果当前有事务，则挂起当前事务，然后新创建一个事务，如果当前没有事务，则自己创建一个事务。
* Propagation.NESTED (nested 嵌套事务) ：如果当前存在事务，则嵌套在当前事务中。如果当前没有事务，则新建一个事务自己执行（和required一样）。嵌套的事务使用保存点作为回滚点，当内部事务回滚时不会影响外部事物的提交；但是外部回滚会把内部事务一起回滚回去。（这个和新建一个事务的区别）

事务开启前奏：入库后抛异常与正常入库

public class UserInfoDaoImpl implements UserInfoDao {

private JdbcTemplate jdbcTemplate;

public Integer save(UserInfoVo userInfoVo) {

SimpleJdbcInsert insert = new SimpleJdbcInsert(jdbcTemplate)

.withTableName("user\_info").usingColumns("user\_name", "age").usingGeneratedKeyColumns("id");

Map<String, Object> params = new HashMap<String, Object>();

params.put("user\_name", userInfoVo.getUserName());

params.put("age", userInfoVo.getAge());

int id = insert.executeAndReturnKey(params).intValue();

throw new RuntimeException("测试异常");

}

public Integer save2(UserInfoVo userInfoVo) {

SimpleJdbcInsert insert = new SimpleJdbcInsert(jdbcTemplate)

.withTableName("user\_info").usingColumns("user\_name", "age").usingGeneratedKeyColumns("id");

Map<String, Object> params = new HashMap<String, Object>();

params.put("user\_name", userInfoVo.getUserName());

params.put("age", userInfoVo.getAge());

int id = insert.executeAndReturnKey(params).intValue();

*//throw new RuntimeException("测试异常");*

return id;

}

public void setJdbcTemplate(JdbcTemplate jdbcTemplate) {

this.jdbcTemplate = jdbcTemplate;

}

}

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18
* 19
* 20
* 21
* 22
* 23
* 24
* 25
* 26
* 27
* 28
* 29
* 30
* 31
* 32
* 33

### **required有则有无则创建（默认）**

说明： 如果当前已经存在事务，那么加入该事务，如果不存在事务，创建一个事务，这是默认的传播属性值。

*//service1中*

public class UserInfoExtendServiceImpl{

@Transactional(propagation = Propagation.REQUIRED)

public void serviceA() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(100);

infoVo.setUserName("ceshi1");

userInfoDao.save(infoVo);

}

@Transactional(propagation = Propagation.REQUIRED)

public void serviceB() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(200);

infoVo.setUserName("ceshi2");

userInfoDao.save2(infoVo);

}

}

*//sercie2中：*public class UserInfoServiceImpl{

private UserInfoExtendService userInfoExtendService;

public void setUserInfoExtendService(UserInfoExtendService userInfoExtendService) {

this.userInfoExtendService = userInfoExtendService;

}

@Transactional

public void service() {

userInfoExtendService.serviceA();

userInfoExtendService.serviceB();

}

}

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18
* 19
* 20
* 21
* 22
* 23
* 24
* 25
* 26
* 27
* 28
* 29
* 30
* 31
* 32
* 33
* 34
* 35
* 36

说明：默认情况下，propagation=PROPAGATION\_REQUIRED，整个service调用过程中，只存在一个共享的事务，当有任何异常发生的时候，所有操作回滚。  
sercieA，serviceB，service，他们三个将为同一个事务。

如果当前有事务则加入事务中，如果当前没有事务则自己创建一个事务，上面的例子中service中有事务了，serviceA，serviceB中自己就不会创建事务了，而是service，serviceA，serviceB为一个事务。  
如果service中没有事务，则sercieA，serviceB会各自创建一个事务，互不影响哦！  
“当前事务”:对于serviceA来说，当前事务就是sercie里面的事务，相当于调用sercieA的那个方法，如果调用它之前就有事务了，则…..

****错误提示****

@Transactional(propagation = Propagation.REQUIRED)

public void serviceA() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(100);

infoVo.setUserName("ceshi1");

userInfoDao.save(infoVo);

}

@Transactional(propagation = Propagation.REQUIRED)

public void serviceB() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(200);

infoVo.setUserName("ceshi2");

userInfoDao.save2(infoVo);

}

@Transactional

public void service() {

serviceA();

serviceB();

}

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18
* 19
* 20
* 21

当时为了写demo所以就写成上面的样子了，导致后面几个事务测试都无效；要知道spring aop内部方法调用会丢失代理的哦。service，serviceA，serviceB在一个类里面，service调用serviceA，serviceB，会产生serviceA，serviceB上面的事务无效，只有service有效。这设计到spring AOP内部调用的问题，会单独拿出一片文章来讲解的。  
可以参考：[1.spring AOP内部调用无效](http://blog.csdn.net/nieyanshun_me/article/details/74898401)  
[2.spring aop之对象内部方法间的嵌套失效](http://blog.csdn.net/doctor_who2004/article/details/51814476)

### **supports有则有无则无事务**

如果当前已经存在事务，那么加入该事务，否则创建一个所谓的空事务（可以认为无事务执行）。

*//Sercie1里面：*

/\*\*

\* 如果当前有事务则加入事务中，如果没有则什么都不做，相当于没事务。

\*/@Transactional(propagation = Propagation.SUPPORTS)

public void serviceC() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(1000);

infoVo.setUserName("ceshi2");

userInfoDao.save(infoVo); *//没有被回滚哦*

}

*//Sercie2里面：*

*//@Transactional,开启事务后，serviceC会和C1公用一个事务，如果这里没有开启，则serviceC不会自己创建事务。*

public void serviceC1() {

Sercie1.serviceC();

}

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18

说明：  
C1在调用C的过程中  
1. C1没有事务，则C也没有事务。  
2. C1有事务，则C加入到C1的事务中。

### **not\_supported忽略事务**

NOT\_SUPPORTED:这个的意思是一直处于无事务状态中执行，如果当前有事务则忽略事务，自己处在一个无事务中执行。  
和上面正好反正，和never事务不一样哦，不会跑异常，自己只是安静的做事。

### **mandatory必须有**

当前必须存在一个事务，否则抛出异常。

*//Sercie1中：*

*//Propagation.MANDATORY当前必须存在一个事务，否则抛出异常。*

@Transactional(propagation = Propagation.MANDATORY)

public void serviceD() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(200);

infoVo.setUserName("D");

userInfoDao.save(infoVo);

}

public void serviceE() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(100);

infoVo.setUserName("E");

userInfoDao.save2(infoVo);

}

*//Sercie2中*

*//这里调用*

public void serviceDE() {

Sercie1.serviceE();*//E正常入库了，*

Sercie1.serviceD();*//D必须要有事务，不然则抛异常了。servcieDE上面没有事务，所有抛异常了。*

}

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18
* 19
* 20
* 21
* 22
* 23
* 24

### **never 必须没有**

如果当前存在事务，则抛出异常，否则在无事务环境上执行代码。

*//Sercie1中：*

*//Propagation.NEVER当前必须没有事务，否则抛出异常。*

@Transactional(propagation = Propagation.NEVER)

public void serviceD() {

System.out.println("执行到了这里。。。。");

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(200);

infoVo.setUserName("D");

userInfoDao.save(infoVo);

}

public void serviceE() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(100);

infoVo.setUserName("E");

userInfoDao.save2(infoVo);

}

*//Sercie2中*

*//这里调用*

public void serviceDE() {

Sercie1.serviceE();*//E正常入库了，*

Sercie1.serviceD();*//D必须要没有事务，不然则抛异常了。*

}

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18
* 19
* 20
* 21
* 22
* 23
* 24
* 25
* 26

说明：servcieDE上面没有事务，D正常以无事务的方式执行，  
sercieDE上面有事务，D抛出异常，sercieD里面直接不回执行就已经往外抛出异常了。D有异常后，如果不处理则会连带sercieE也会滚，别忘记了serviceDE中事务。

### **requires\_new新事务**

REQUIRES\_NEW新建一个事务，不管当前有没有事务，都新建一个独立的事务。  
如果当前存在事务，则把当前事务挂起，自己新创建一个事务，新事务执行完毕后再唤醒当前事务；两个事务没有依赖关系，可以实现自己新事务回滚了，但外部事务继续执行；外部事物回滚也不会影响到新事物的的提交。就是双方互不影响。

*//Sercie1类中*/\*\*

\* REQUIRES\_NEW新建一个事务，不管当前有没有事务，都新建一个独立的事务。

\* 这里面serviceFG创建了一个事务，然后serviceF也创建了一个事务，他们互相独立；

\* 当前方法必须在自己的事务里运行，如果当前存在一个事务，则挂起该事务，这个事务执行完毕后，再唤醒挂起的事务。

\* 挂起事务使用suspend方法将原事务挂起。

\*/

@Transactional(propagation = Propagation.REQUIRES\_NEW)

public void serviceF() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(1000);

infoVo.setUserName("F2");

userInfoDao.save(infoVo); *//这个里面有一个异常哦！*

}

public void serviceG() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(1000);

infoVo.setUserName("G2");

userInfoDao.save2(infoVo);

}

*//Sercie2中*

@Transactional

public void serviceFG() {

Sercie1.serviceG();

try {

Sercie1.serviceF();

} catch (Exception e) {

*//e.printStackTrace();*

}

Sercie1.serviceH();

}

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18
* 19
* 20
* 21
* 22
* 23
* 24
* 25
* 26
* 27
* 28
* 29
* 30
* 31
* 32
* 33
* 34
* 35

注意虽然sercieF新建了一个事务，但是如果serviceF抛出异常，还是需要捕获 不然serviceFG里面里面发现有异常抛出，就会把serviceG也给回滚了。

sercieFG有事务：  
1. sercieG()提交成，serviceF新建一个事务，serviceF内部有错，则自己回滚自己的；serviceFG捕获了sercieF的异常则不会影响到serviceFG的事务提交。  
如果不处理serviceF中的异常，serviceFG发现异常会回滚自己的事务，这个时候serviceG也会被回滚。  
2. serviceG()正常执行，sercieF()正常执行后，自己就提交事务了；serviceH抛异常了，会导致serviceG和serviceH中的数据库操作回滚，但是不会影响到serviceF中的提交。

sercieFG中无事务  
sercieG(),sercieH该咋的就咋的，不存在回滚的情况，也不会影响到serviceF；serviceF会自己新建一个事务，自己处理自己内部的事。

### **nested嵌套事务**

如果当前存在事务，则使用 SavePoint 技术把当前事务状态进行保存，然后底层共用一个连接，当NESTED内部出错的时候，自行回滚到 SavePoint这个状态，只要外部捕获到了异常，就可以继续进行外部的事务提交，而不会受到内嵌业务的干扰，但是，如果外部事务抛出了异常，整个大事务都会回滚。

如果没有，则自己新建一个事务自己处理。

*//Sercie1*

/\*\*

\* 嵌套事务，如果当前有事务则设置保存点，没有则新启一个事务

\*/

@Transactional(propagation = Propagation.NESTED)

public void serviceH() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(1000);

infoVo.setUserName("H");

userInfoDao.save(infoVo); *//这个里面有一个异常哦！*

}

public void serviceI() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(1000);

infoVo.setUserName("I");

userInfoDao.save2(infoVo);

}

*//Sercie2*

@Transactional

public void serviceHI() {

Sercie1.serviceI();

try {

Sercie1.serviceH();

} catch (Exception e) {

e.printStackTrace();

}

}

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18
* 19
* 20
* 21
* 22
* 23
* 24
* 25
* 26
* 27
* 28
* 29

如果当前调用方有事务  
1）serviceH方法内部报错，则只会回滚serviceH里面的。  
2）serviceH方法内部不报错，但是外面的调用方报错了，则serviceH会跟着一起回滚。  
3）serviceH方法内部不报错，外面也不报错，则serviceH和外面事务一起提交。

如果当前没有事务  
serviceH就相当于一个自己新建了一个事务，和外面没有关系了，它内部就一个独立事务了。

在一个事务里面再嵌套一个事务，嵌套的事务就是在一个当前事务中设置一个保存点，保存点内部事务报错，则回滚保存点内部事务，不影响外面的。  
与事务中新建事务的区别：在当前事务中新建事务，如果新事务中报错内部会回滚不影响外面的事务，外面的事务报错了，外面事务的回滚不会把新事务中的事务给回滚掉，而嵌套事务则会跟着外部事务一起回滚。

### **异常被捕获不回滚**

userInfoDao.save(UserInfoVo userinfovo){

int id = insert.executeAndReturnKey(params).intValue();*//插入数据库*

throw new RuntimeException("测试异常");

}

@Transactional(rollbackFor = Exception.class)

public void serviceI() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(102222);

infoVo.setUserName("I");

try {

userInfoDao.save(infoVo); *//这个里面有一个异常哦！*

} catch (Exception e) { *//异常被catch住了，事务不会回滚。*

e.printStackTrace();

}

}

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17

userInfoDao.save(infoVo);这个里面有个异常，抛异常后，被cry catch住了，所以这个事物不会回滚，数据还是插入数据库了。  
异常被catch住了，就相当于没有异常了。

### **多线程事务传播性失效**

@Transactional(propagation = Propagation.NESTED)

public void serviceH() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(10020);

infoVo.setUserName("H");

userInfoDao.save2(infoVo); *//这个里面没有异常*

}

public void serviceI() {

UserInfoVo infoVo = new UserInfoVo();

infoVo.setAge(1012);

infoVo.setUserName("I");

userInfoDao.save(infoVo); *//这个里面有异常*

}

@Transactional

public void serviceHI() {

System.out.println("HI......");

new Thread(new Runnable() {

public void run() {

userInfoExtendService.serviceH(); *//嵌套事务*

}

}).start();

*//serviceI抛异常了，按理说serviceH也应该跟着一起回滚的，但是由于serviceH开启了一个独立的线程，所以serviceH已经和serviceI不是同一个事务了*

*//serviceI抛异常了，serviceH不会跟着回滚*

userInfoExtendService.serviceI();

}

* 1
* 2
* 3
* 4
* 5
* 6
* 7
* 8
* 9
* 10
* 11
* 12
* 13
* 14
* 15
* 16
* 17
* 18
* 19
* 20
* 21
* 22
* 23
* 24
* 25
* 26
* 27
* 28

前面介绍过嵌套事务，如果serviceH嵌套事务没有异常，serviceI有异常，他们是需要一起回滚的；serviceH有异常，serviceI没有异常，则serviceH自己回滚自己，serviceI继续提交。这里serviceI抛异常了，按理serviceH也应该一起回滚的，但是由于serviceH开启了一个独立的线程，所以serviceH已经和serviceI不是同一个事务了。事务的传播性也就断了。

注意：事务必须在同一个线程中才有效，serviceI与serviceH不在同一个线程中，他们之间就没有事务关系了。各自为政，各自提交各自的，自个为一个独立的事务。  
原因很简单，spring事务一个线程绑定一个数据库session，在该线程的数据库session中修改数据库的事务属性，改为手动提交。如果不同线程则为不同的数据库session了，不同session是互相隔离的，所以serviceI与serviceH他们两个是两个线程，也就导致了最后操作数据库是两个session了。  
当然，serviceI本身还是有事务特性的，serviceH本身也还有事务特性的。只是serviceI与serviceH不在是一个事务而已了。

### **总结：**

上面说了很多事务，记住，只要在同一个事务里面要么都成功要么都失败。遇到如：Propagation.NEVER有事务则抛出异常，如果不处理异常，它上层的事务也会因为接收到了一场而导致事务回滚。

## **数据库隔离级别**

### **spring的事务隔离级别**

| **操作名称** | **说明** | **级别** |
| --- | --- | --- |
| @Transactional(isolation = Isolation.DEFAULT) | 默认隔离级别，和数据库的中的4种对应，数据库中用啥，spring事务隔离就用啥 | -1 |
| @Transactional(isolation = Isolation.READ\_UNCOMMITTED) | 读取未提交数据(会出现脏读， 不可重复读，幻读)，基本不使用 | 1 |
| @Transactional(isolation = Isolation.READ\_COMMITTED)(SQLSERVER默认) | 读取已提交数据(会出现不可重复读和幻读) | 2 |
| @Transactional(isolation = Isolation.REPEATABLE\_READ) | 可重复读(会出现幻读) | 4 |
| @Transactional(isolation = Isolation.SERIALIZABLE) | 串行化 | 8 |

### **数据库的事务隔离级别**



* 脏读：一事务对数据进行了增删改，但未提交，另一事务可以读取到未提交的数据。如果第一个事务这时候回滚了，那么第二个事务就读到了脏数据。
* 不可重复读：一个事务中发生了两次读操作，第一次读操作和第二次读操作之间，另外一个事务对数据进行了修改，这时候两次读取的数据是不一致的。
* 幻读：第一个事务对一定范围的数据进行批量修改，第二个事务在这个范围增加一条数据，这时候第一个事务就会丢失对新增数据的修改。
* 串行化：不会出现任何问题。只能一个事务提交后，另外一个事务才能开始。最安全，但效率也是最差的。

隔离级别越高，越能保证数据的完整性和一致性，但是对并发性能的影响也越大。  
大多数的数据库默认隔离级别为 1，比如 SqlServer、Oracle  
少数数据库默认隔离级别为：2 比如： MySQL InnoDB

参考地址：

<http://blog.csdn.net/yanyan19880509/article/details/53041564>  
<http://www.uml.org.cn/j2ee/201610201.asp>  
<https://www.cnblogs.com/a757956132/p/4765599.html>

文章最后发布于: 2018-08-16 16:04:27

有 0 个人打赏

[私信求帮助](https://im.csdn.net/im/main.html?userName=piaoslowly" \t "/Users/liu123/Documents\\x/_blank)

[IMG_263](javascript:void(0);)

窗体顶端

窗体底端

#### [【Spring学习34】Spring事务(4)：事务属性之7种传播行为](https://blog.csdn.net/soonfly/article/details/70305683" \o "【Spring学习34】Spring事务(4)：事务属性之7种传播行为" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 5663](https://blog.csdn.net/soonfly/article/details/70305683" \o "【Spring学习34】Spring事务(4)：事务属性之7种传播行为" \t "/Users/liu123/Documents\\x/_blank)

[事务传播行为什么叫事务传播行为？听起来挺高端的，其实很简单。即然是传播，那么至少有两个东西，才可以发生传播。单体不存在传播这个行为。事务传播行为（propagationbehavior）指的就是当一个...](https://blog.csdn.net/soonfly/article/details/70305683" \o "【Spring学习34】Spring事务(4)：事务属性之7种传播行为" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 程序老兵的博客](https://blog.csdn.net/soonfly" \t "/Users/liu123/Documents\\x/_blank)

#### [看完就明白\_spring事务的7种传播行为](https://blog.csdn.net/weixin_39625809/article/details/80707695" \o "看完就明白_spring事务的7种传播行为" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2万+](https://blog.csdn.net/weixin_39625809/article/details/80707695" \o "看完就明白_spring事务的7种传播行为" \t "/Users/liu123/Documents\\x/_blank)

[原文：https://blog.csdn.net/soonfly/article/details/70305683事务传播行为什么叫事务传播行为？听起来挺高端的，其实很简单。 即然是传播，那么至少有两...](https://blog.csdn.net/weixin_39625809/article/details/80707695" \o "看完就明白_spring事务的7种传播行为" \t "/Users/liu123/Documents\\x/_blank)博文[来自： weixin\_39625809的博客](https://blog.csdn.net/weixin_39625809" \t "/Users/liu123/Documents\\x/_blank)

#### [Spring事务的传播特性](https://blog.csdn.net/lfsf802/article/details/9417095" \o "Spring事务的传播特性" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/lfsf802/article/details/9417095" \o "Spring事务的传播特性" \t "/Users/liu123/Documents\\x/_blank)

[Spring中通过Propagation来设置事务的传播属性的，在这个属性中提供了我们其中关于事务传播的特性：    1.     PROPAGATION\_REQUIRED：支持当前事务，如果当前没有...](https://blog.csdn.net/lfsf802/article/details/9417095" \o "Spring事务的传播特性" \t "/Users/liu123/Documents\\x/_blank)博文[来自： Q](https://blog.csdn.net/lfsf802" \t "/Users/liu123/Documents\\x/_blank)

#### [spring的两种常用事务传播属性说明](https://blog.csdn.net/qianxiaopeng/article/details/82427689" \o "spring的两种常用事务传播属性说明" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 3605](https://blog.csdn.net/qianxiaopeng/article/details/82427689" \o "spring的两种常用事务传播属性说明" \t "/Users/liu123/Documents\\x/_blank)

[常用的两种传播属性：1.REQUIRED 如果存在当前事务则用当前事务，如果不存在当前事务，则新建一个事务2.REQUIRES\_NEW 如果当前存在事务则挂起当前事务，开启一个新事务，新事务执行完毕后...](https://blog.csdn.net/qianxiaopeng/article/details/82427689" \o "spring的两种常用事务传播属性说明" \t "/Users/liu123/Documents\\x/_blank)博文[来自： qianxiaopeng的专栏](https://blog.csdn.net/qianxiaopeng" \t "/Users/liu123/Documents\\x/_blank)

#### [Spring的7种事务传播行为类型](https://blog.csdn.net/hsgao_water/article/details/52860380" \o "Spring的7种事务传播行为类型" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2万+](https://blog.csdn.net/hsgao_water/article/details/52860380" \o "Spring的7种事务传播行为类型" \t "/Users/liu123/Documents\\x/_blank)

[1、PROPAGATION\_REQUIRED：如果当前没有事务，就创建一个新事务，如果当前存在事务，就加入该事务，该设置是最常用的设置。2、PROPAGATION\_SUPPORTS：支持当前事务，如果...](https://blog.csdn.net/hsgao_water/article/details/52860380" \o "Spring的7种事务传播行为类型" \t "/Users/liu123/Documents\\x/_blank)博文[来自： hsgao\_water的博客](https://blog.csdn.net/hsgao_water" \t "/Users/liu123/Documents\\x/_blank)

#### [挑战10个最难的Java面试题（附答案）【上】](https://blog.csdn.net/devcloud/article/details/100698417" \o "挑战10个最难的Java面试题（附答案）【上】" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 4万+](https://blog.csdn.net/devcloud/article/details/100698417" \o "挑战10个最难的Java面试题（附答案）【上】" \t "/Users/liu123/Documents\\x/_blank)

[欢迎添加华为云小助手微信（微信号：HWCloud002或HWCloud003），验证通过后，输入关键字“加群”，加入华为云线上技术讨论群；输入关键字“最新活动”，获取华为云最新特惠促销。华为云诸多技术...](https://blog.csdn.net/devcloud/article/details/100698417" \o "挑战10个最难的Java面试题（附答案）【上】" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 华为云官方博客](https://blog.csdn.net/devcloud" \t "/Users/liu123/Documents\\x/_blank)

#### [spring-事务管理](https://blog.csdn.net/weixin_33835103/article/details/86219729" \o "spring-事务管理" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 126](https://blog.csdn.net/weixin_33835103/article/details/86219729" \o "spring-事务管理" \t "/Users/liu123/Documents\\x/_blank)

[一个数据库事务是一个被视为单一的工作单元的操作序列。这些操作应该要么完整地执行，要么完全不执行。事务管理是一个重要组成部分，RDBMS 面向企业应用程序，以确保数据完整性和一致性。事务的概念可以描述为...](https://blog.csdn.net/weixin_33835103/article/details/86219729" \o "spring-事务管理" \t "/Users/liu123/Documents\\x/_blank)博文[来自： weixin\_33835103的博客](https://blog.csdn.net/weixin_33835103" \t "/Users/liu123/Documents\\x/_blank)

#### [spring事物传播行为](https://blog.csdn.net/huchangan2005678/article/details/84341573" \o "spring事物传播行为" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 12](https://blog.csdn.net/huchangan2005678/article/details/84341573" \o "spring事物传播行为" \t "/Users/liu123/Documents\\x/_blank)

[PROPAGATION\_REQUIRED如果当前没有事务，就新建一个事务，如果已经存在一个事务中，加入到这个事务中。这是最常见的选择。PROPAGATION\_SUPPORTS支持当前事务，如果当前没有...](https://blog.csdn.net/huchangan2005678/article/details/84341573" \o "spring事物传播行为" \t "/Users/liu123/Documents\\x/_blank)博文[来自： huchangan2005678的博客](https://blog.csdn.net/huchangan2005678" \t "/Users/liu123/Documents\\x/_blank)

#### [面试官：兄弟，说说基本类型和包装类型的区别吧](https://blog.csdn.net/qing_gee/article/details/101670051" \o "面试官：兄弟，说说基本类型和包装类型的区别吧" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 3万+](https://blog.csdn.net/qing_gee/article/details/101670051" \o "面试官：兄弟，说说基本类型和包装类型的区别吧" \t "/Users/liu123/Documents\\x/_blank)

[Java的每个基本类型都对应了一个包装类型，比如说int的包装类型为Integer，double的包装类型为Double。基本类型和包装类型的区别主要有以下4点。...](https://blog.csdn.net/qing_gee/article/details/101670051" \o "面试官：兄弟，说说基本类型和包装类型的区别吧" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 沉默王二](https://blog.csdn.net/qing_gee" \t "/Users/liu123/Documents\\x/_blank)

#### [spring的两种常用事务传播属性说明 - qianxiaopeng的专...\_CSDN博客](http://blog.csdn.net/qianxiaopeng/article/details/82427689" \t "/Users/liu123/Documents\\x/_blank)

[11-21](http://blog.csdn.net/qianxiaopeng/article/details/82427689" \t "/Users/liu123/Documents\\x/_blank)

#### [Spring事务传播属性 - liben0429的博客 - CSDN博客](https://blog.csdn.net/liben0429/article/details/79179450" \t "/Users/liu123/Documents\\x/_blank)

[10-5](https://blog.csdn.net/liben0429/article/details/79179450" \t "/Users/liu123/Documents\\x/_blank)

#### [开发人员提升自己的四种方式](https://blog.csdn.net/csdnnews/article/details/102425401" \o "开发人员提升自己的四种方式" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/csdnnews/article/details/102425401" \o "开发人员提升自己的四种方式" \t "/Users/liu123/Documents\\x/_blank)

[在提升软件开发技能方面，有哪些必备的妙招可以帮助自己呢？作者 |Skill Pathway译者 |弯月，责编 | 屠敏出品 | CSDN（ID：CSDNnews）以下为译文：开发人员之间所谓的不同“级...](https://blog.csdn.net/csdnnews/article/details/102425401" \o "开发人员提升自己的四种方式" \t "/Users/liu123/Documents\\x/_blank)博文[来自： CSDN资讯](https://blog.csdn.net/csdnnews" \t "/Users/liu123/Documents\\x/_blank)

#### [Spring事务传播行为类型 - luozhaohuan的博客 - CSDN博客](https://blog.csdn.net/luozhaohuan/article/details/87797863" \t "/Users/liu123/Documents\\x/_blank)

[11-11](https://blog.csdn.net/luozhaohuan/article/details/87797863" \t "/Users/liu123/Documents\\x/_blank)

#### [spring中七种事务传播属性 - 徐海兴的专栏 - CSDN博客](https://blog.csdn.net/u012326462/article/details/81462446" \t "/Users/liu123/Documents\\x/_blank)

[11-11](https://blog.csdn.net/u012326462/article/details/81462446" \t "/Users/liu123/Documents\\x/_blank)

#### [spring事务传播属性与隔离级别](https://blog.csdn.net/liao756447021/article/details/9272295" \o "spring事务传播属性与隔离级别" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 436](https://blog.csdn.net/liao756447021/article/details/9272295" \o "spring事务传播属性与隔离级别" \t "/Users/liu123/Documents\\x/_blank)

[一、Propagation（事务的传播属性）Propagation：　　key属性确定代理应该给哪个方法增加事务行为。这样的属性最重要的部份是传播行为。有以下选项可供使用：PROPAGATION\_RE...](https://blog.csdn.net/liao756447021/article/details/9272295" \o "spring事务传播属性与隔离级别" \t "/Users/liu123/Documents\\x/_blank)博文[来自： liao756447021的专栏](https://blog.csdn.net/liao756447021" \t "/Users/liu123/Documents\\x/_blank)

[](https://blog.csdn.net/soonfly)关注

##### [唐大麦](https://blog.csdn.net/soonfly" \t "/Users/liu123/Documents\\x/_blank)

188篇文章

排名:4000+

[](https://blog.csdn.net/weixin_39625809)关注

##### [gnixlee](https://blog.csdn.net/weixin_39625809" \t "/Users/liu123/Documents\\x/_blank)

4篇文章

排名:千里之外

[lfsf802](https://blog.csdn.net/lfsf802)关注

##### [lfsf802](https://blog.csdn.net/lfsf802" \t "/Users/liu123/Documents\\x/_blank)

193篇文章

排名:1000+

[河里的树木](https://blog.csdn.net/qianxiaopeng)关注

##### [河里的树木](https://blog.csdn.net/qianxiaopeng" \t "/Users/liu123/Documents\\x/_blank)

29篇文章

排名:千里之外

#### [spring\_事物传播属性理解 - maqingbin8888的专栏 - CSDN博客](https://blog.csdn.net/maqingbin8888/article/details/81739263" \t "/Users/liu123/Documents\\x/_blank)

[10-4](https://blog.csdn.net/maqingbin8888/article/details/81739263" \t "/Users/liu123/Documents\\x/_blank)

#### [Spring的事务传播属性 - jieshaowang1229的专栏 - CSDN博客](https://blog.csdn.net/jieshaowang1229/article/details/50393458" \t "/Users/liu123/Documents\\x/_blank)

[11-3](https://blog.csdn.net/jieshaowang1229/article/details/50393458" \t "/Users/liu123/Documents\\x/_blank)

#### [30秒内便能学会的30个超实用Python代码片段](https://blog.csdn.net/duxinshuxiaobian/article/details/102480959" \o "30秒内便能学会的30个超实用Python代码片段" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 3万+](https://blog.csdn.net/duxinshuxiaobian/article/details/102480959" \o "30秒内便能学会的30个超实用Python代码片段" \t "/Users/liu123/Documents\\x/_blank)

[许多人在数据科学、机器学习、web开发、脚本编写和自动化等领域中都会使用Python，它是一种十分流行的语言。Python流行的部分原因在于简单易学。本文将简要介绍30个简短的、且能在30秒内掌握的代...](https://blog.csdn.net/duxinshuxiaobian/article/details/102480959" \o "30秒内便能学会的30个超实用Python代码片段" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 读芯术的博客](https://blog.csdn.net/duxinshuxiaobian" \t "/Users/liu123/Documents\\x/_blank)

#### [Spring事务传播属性](https://blog.csdn.net/qq_40910541/article/details/88718936" \o "Spring事务传播属性" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1178](https://blog.csdn.net/qq_40910541/article/details/88718936" \o "Spring事务传播属性" \t "/Users/liu123/Documents\\x/_blank)

[事务的传播属性（Propagation）REQUIRED ，这个是默认的属性如果存在一个事务，则支持当前事务。如果没有事务则开启一个新的事务。被设置成这个级别时，会为每一个被调用的方法创建一个逻辑事务...](https://blog.csdn.net/qq_40910541/article/details/88718936" \o "Spring事务传播属性" \t "/Users/liu123/Documents\\x/_blank)博文[来自： RuiDer的博客](https://blog.csdn.net/qq_40910541" \t "/Users/liu123/Documents\\x/_blank)

#### [事务的传播属性都有哪些? - CSDN博客](https://blog.csdn.net/u014698348/article/details/53142246" \t "/Users/liu123/Documents\\x/_blank)

[11-19](https://blog.csdn.net/u014698348/article/details/53142246" \t "/Users/liu123/Documents\\x/_blank)

#### [spring事务传播属性 - weixin\_34199335的博客 - CSDN博客](https://blog.csdn.net/weixin_34199335/article/details/90961170" \t "/Users/liu123/Documents\\x/_blank)

[6-26](https://blog.csdn.net/weixin_34199335/article/details/90961170" \t "/Users/liu123/Documents\\x/_blank)

#### [对计算机专业来说学历真的重要吗？](https://blog.csdn.net/harvic880925/article/details/101511637" \o "对计算机专业来说学历真的重要吗？" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 8万+](https://blog.csdn.net/harvic880925/article/details/101511637" \o "对计算机专业来说学历真的重要吗？" \t "/Users/liu123/Documents\\x/_blank)

[我本科学校是渣渣二本，研究生学校是985，现在毕业五年，校招笔试、面试，社招面试参加了两年了，就我个人的经历来说下这个问题。这篇文章很长，但绝对是精华，相信我，读完以后，你会知道学历不好的解决方案，记...](https://blog.csdn.net/harvic880925/article/details/101511637" \o "对计算机专业来说学历真的重要吗？" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 启舰](https://blog.csdn.net/harvic880925" \t "/Users/liu123/Documents\\x/_blank)

#### [spring事务传播属性和隔离级别 - u010797575的专栏 - CSDN博客](https://blog.csdn.net/u010797575/article/details/87872573" \t "/Users/liu123/Documents\\x/_blank)

[10-31](https://blog.csdn.net/u010797575/article/details/87872573" \t "/Users/liu123/Documents\\x/_blank)

#### [让程序员崩溃的瞬间（非程序员勿入）](https://blog.csdn.net/ybhuangfugui/article/details/100913641" \o "让程序员崩溃的瞬间（非程序员勿入）" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 23万+](https://blog.csdn.net/ybhuangfugui/article/details/100913641" \o "让程序员崩溃的瞬间（非程序员勿入）" \t "/Users/liu123/Documents\\x/_blank)

[今天给大家带来点快乐，程序员才能看懂。来源：https://zhuanlan.zhihu.com/p/470665211. 公司实习生找 Bug2.在调试时，将断点设置在错误的位置3.当我有一个很棒的...](https://blog.csdn.net/ybhuangfugui/article/details/100913641" \o "让程序员崩溃的瞬间（非程序员勿入）" \t "/Users/liu123/Documents\\x/_blank)博文[来自： strongerHuang](https://blog.csdn.net/ybhuangfugui" \t "/Users/liu123/Documents\\x/_blank)

#### [spring 默认事务传播属性](https://blog.csdn.net/AlbertFly/article/details/52682881" \o "spring 默认事务传播属性" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 3925](https://blog.csdn.net/AlbertFly/article/details/52682881" \o "spring 默认事务传播属性" \t "/Users/liu123/Documents\\x/_blank)

[PROPAGATION\_REQUIRED](https://blog.csdn.net/AlbertFly/article/details/52682881" \o "spring 默认事务传播属性" \t "/Users/liu123/Documents\\x/_blank)博文[来自： Flyer的后花园](https://blog.csdn.net/AlbertFly" \t "/Users/liu123/Documents\\x/_blank)

#### [JAVA-快速了解线程池的基本原理](https://blog.csdn.net/weixin_44807144/article/details/102510056" \o "JAVA-快速了解线程池的基本原理" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/weixin_44807144/article/details/102510056" \o "JAVA-快速了解线程池的基本原理" \t "/Users/liu123/Documents\\x/_blank)

[前言说起线程池大家肯定不会陌生，在面试中属于必问的问题之一，特别是对于高并发有较高要求的企业，基本是核心打分点。网上关于线程池的文章和视频很多，本篇文章旨在帮助大家快速了解和掌握线程池的原理，对于高级...](https://blog.csdn.net/weixin_44807144/article/details/102510056" \o "JAVA-快速了解线程池的基本原理" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 我在风花雪月里等你](https://blog.csdn.net/weixin_44807144" \t "/Users/liu123/Documents\\x/_blank)

#### [100 个网络基础知识普及，看完成半个网络高手](https://blog.csdn.net/devcloud/article/details/101199255" \o "100 个网络基础知识普及，看完成半个网络高手" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 12万+](https://blog.csdn.net/devcloud/article/details/101199255" \o "100 个网络基础知识普及，看完成半个网络高手" \t "/Users/liu123/Documents\\x/_blank)

[1）什么是链接？链接是指两个设备之间的连接。它包括用于一个设备能够与另一个设备通信的电缆类型和协议。2）OSI 参考模型的层次是什么？有 7 个 OSI 层：物理层，数据链路层，网络层，传输层，会话层...](https://blog.csdn.net/devcloud/article/details/101199255" \o "100 个网络基础知识普及，看完成半个网络高手" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 华为云官方博客](https://blog.csdn.net/devcloud" \t "/Users/liu123/Documents\\x/_blank)

#### [spring中七种事务传播属性](https://blog.csdn.net/u012326462/article/details/81462446" \o "spring中七种事务传播属性" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 224](https://blog.csdn.net/u012326462/article/details/81462446" \o "spring中七种事务传播属性" \t "/Users/liu123/Documents\\x/_blank)

[/\* \* Copyright 2002-2015 the original author or authors. \* \* Licensed under the Apache License, Vers...](https://blog.csdn.net/u012326462/article/details/81462446" \o "spring中七种事务传播属性" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 徐海兴的专栏](https://blog.csdn.net/u012326462" \t "/Users/liu123/Documents\\x/_blank)

#### [Spring事务（二）-事务的传播行为](https://blog.csdn.net/fu123123fu/article/details/86359804" \o "Spring事务（二）-事务的传播行为" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 680](https://blog.csdn.net/fu123123fu/article/details/86359804" \o "Spring事务（二）-事务的传播行为" \t "/Users/liu123/Documents\\x/_blank)

[由来‘“事务的传播行为”，听起来就好高大上的感觉。其实上一篇事务回顾也是为了给这个做铺垫的，我们已经知道事务是怎么回事了，但是得把这个概念真正转换到代码上。我们写代码都知道有controller、se...](https://blog.csdn.net/fu123123fu/article/details/86359804" \o "Spring事务（二）-事务的传播行为" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 长风破浪会有33的博客](https://blog.csdn.net/fu123123fu" \t "/Users/liu123/Documents\\x/_blank)

#### [C语言这么厉害，它自身又是用什么语言写的？](https://blog.csdn.net/coderising/article/details/101731213" \o "C语言这么厉害，它自身又是用什么语言写的？" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 3万+](https://blog.csdn.net/coderising/article/details/101731213" \o "C语言这么厉害，它自身又是用什么语言写的？" \t "/Users/liu123/Documents\\x/_blank)

[这是来自我的星球的一个提问：“C语言本身用什么语言写的？”换个角度来问，其实是：C语言在运行之前，得编译才行，那C语言的编译器从哪里来？用什么语言来写的？如果是用C语......](https://blog.csdn.net/coderising/article/details/101731213" \o "C语言这么厉害，它自身又是用什么语言写的？" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 码农翻身](https://blog.csdn.net/coderising" \t "/Users/liu123/Documents\\x/_blank)

#### [我花了一夜用数据结构给女朋友写个H5走迷宫游戏](https://blog.csdn.net/qq_40693171/article/details/100716766" \o "我花了一夜用数据结构给女朋友写个H5走迷宫游戏" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 16万+](https://blog.csdn.net/qq_40693171/article/details/100716766" \o "我花了一夜用数据结构给女朋友写个H5走迷宫游戏" \t "/Users/liu123/Documents\\x/_blank)

[起因又到深夜了，我按照以往在csdn和公众号写着数据结构！这占用了我大量的时间！我的超越妹妹严重缺乏陪伴而怨气满满！而女朋友时常埋怨，认为数据结构这么抽象难懂的东西没啥作用，常会问道：天天写这玩意，有...](https://blog.csdn.net/qq_40693171/article/details/100716766" \o "我花了一夜用数据结构给女朋友写个H5走迷宫游戏" \t "/Users/liu123/Documents\\x/_blank)博文[来自： bigsai](https://blog.csdn.net/qq_40693171" \t "/Users/liu123/Documents\\x/_blank)

#### [Spring 事务在多线程环境下的传播](https://blog.csdn.net/iteye_15896/article/details/82327238" \o "Spring 事务在多线程环境下的传播" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 503](https://blog.csdn.net/iteye_15896/article/details/82327238" \o "Spring 事务在多线程环境下的传播" \t "/Users/liu123/Documents\\x/_blank)

[有时候需要使用多线程来提高对于CPU，尤其是多核CPU的利用率，来提高性能。Spring的bean一般都是无状态的，单例的。所以也就是线程安全的。那么对于声明式的事务，在多线程环境下又是怎么处理的呢？...](https://blog.csdn.net/iteye_15896/article/details/82327238" \o "Spring 事务在多线程环境下的传播" \t "/Users/liu123/Documents\\x/_blank)博文[来自： JackyCheng2007](https://blog.csdn.net/iteye_15896" \t "/Users/liu123/Documents\\x/_blank)

#### [spring的七种事务传播属性](https://blog.csdn.net/xuSir_1/article/details/72844779" \o "spring的七种事务传播属性" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 3534](https://blog.csdn.net/xuSir_1/article/details/72844779" \o "spring的七种事务传播属性" \t "/Users/liu123/Documents\\x/_blank)

[spring事物的七种事物传播属性行为及五种隔离级别首先,说说什么事务（Transaction）。事务，就是一组操作数据库的动作集合。事务是现代数据库理论中的核心概念之一。如果一组处理步骤或者全部发生...](https://blog.csdn.net/xuSir_1/article/details/72844779" \o "spring的七种事务传播属性" \t "/Users/liu123/Documents\\x/_blank)博文[来自： xuSir\_1的博客](https://blog.csdn.net/xuSir_1" \t "/Users/liu123/Documents\\x/_blank)

#### [漫画 | 外行对程序员误会有多深！](https://blog.csdn.net/ityouknow/article/details/102512836" \o "漫画 | 外行对程序员误会有多深！" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/ityouknow/article/details/102512836" \o "漫画 | 外行对程序员误会有多深！" \t "/Users/liu123/Documents\\x/_blank)

[作者：阿波、纯洁的微笑漫画：宁州枪手程序员如今已经发展成社会的主流职业，以至于街头的王大妈李大爷都能说出一二来，据说他们认为的程序员是这样子的：程序员都是秃头，秃的越狠越......](https://blog.csdn.net/ityouknow/article/details/102512836" \o "漫画 | 外行对程序员误会有多深！" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 纯洁的微笑](https://blog.csdn.net/ityouknow" \t "/Users/liu123/Documents\\x/_blank)

#### [spring事务传播级别](https://blog.csdn.net/qq_36094023/article/details/90544286" \o "spring事务传播级别" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 253](https://blog.csdn.net/qq_36094023/article/details/90544286" \o "spring事务传播级别" \t "/Users/liu123/Documents\\x/_blank)

[1.2 事物特性定义事物最重要的两个特性，是事物的传播级别和数据隔离级别。传播级别定义的是事物的控制范围，事物隔离界别定义的是事物在数据库读写方面的控制范围。今天只写事物的传播级别。第二章 事物的传播...](https://blog.csdn.net/qq_36094023/article/details/90544286" \o "spring事务传播级别" \t "/Users/liu123/Documents\\x/_blank)博文[来自： qq\_36094023的博客](https://blog.csdn.net/qq_36094023" \t "/Users/liu123/Documents\\x/_blank)

#### [反转！物联网火爆，程序员开发技能却有待加强？](https://blog.csdn.net/csdnsevenn/article/details/100787853" \o "反转！物联网火爆，程序员开发技能却有待加强？" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 293](https://blog.csdn.net/csdnsevenn/article/details/100787853" \o "反转！物联网火爆，程序员开发技能却有待加强？" \t "/Users/liu123/Documents\\x/_blank)

[近几年来，物联网发展迅速：据中商产业研究院《2016——2021年中国物联网产业市场研究报告》显示，预计到2020年，中国物联网的整体规模将达2.2万亿元，产业规模比互联......](https://blog.csdn.net/csdnsevenn/article/details/100787853" \o "反转！物联网火爆，程序员开发技能却有待加强？" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 程序人生的博客](https://blog.csdn.net/csdnsevenn" \t "/Users/liu123/Documents\\x/_blank)

#### [别再翻了，面试二叉树看这 11 个就够了~](https://blog.csdn.net/qq_36903042/article/details/100798101" \o "别再翻了，面试二叉树看这 11 个就够了~" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 6万+](https://blog.csdn.net/qq_36903042/article/details/100798101" \o "别再翻了，面试二叉树看这 11 个就够了~" \t "/Users/liu123/Documents\\x/_blank)

[写在前边数据结构与算法：不知道你有没有这种困惑，虽然刷了很多算法题，当我去面试的时候，面试官让你手写一个算法，可能你对此算法很熟悉，知道实现思路，但是总是不知道该在什么地方写，而且很多边界条件想不全面...](https://blog.csdn.net/qq_36903042/article/details/100798101" \o "别再翻了，面试二叉树看这 11 个就够了~" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 一个不甘平凡的码农](https://blog.csdn.net/qq_36903042" \t "/Users/liu123/Documents\\x/_blank)

#### [spring事务的七种传播属性和五种隔离级别](https://blog.csdn.net/m0_38110132/article/details/74573304" \o "spring事务的七种传播属性和五种隔离级别" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2662](https://blog.csdn.net/m0_38110132/article/details/74573304" \o "spring事务的七种传播属性和五种隔离级别" \t "/Users/liu123/Documents\\x/_blank)

[事务定义事务，就是一组操作数据库的动作集合。事务是现代数据库理论中的核心概念之一。如果一组处理步骤或者全部发生或者一步也不执行，我们称该组处理步骤为一个事务。当所有的步骤像一个操作一样被完整地执行，我...](https://blog.csdn.net/m0_38110132/article/details/74573304" \o "spring事务的七种传播属性和五种隔离级别" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 浮生一梦](https://blog.csdn.net/m0_38110132" \t "/Users/liu123/Documents\\x/_blank)

#### [Spring的事务传播属性](https://blog.csdn.net/jieshaowang1229/article/details/50393458" \o "Spring的事务传播属性" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 543](https://blog.csdn.net/jieshaowang1229/article/details/50393458" \o "Spring的事务传播属性" \t "/Users/liu123/Documents\\x/_blank)

[Spring的事务传播属性，可以选择以下几个取值，具体的取值含义如下：Never：指定业务方法绝对不能在事务范围内执行。如果业务方法在某个事务中执行，容器会抛出例外只有业务方法没有关联到任何事务，才能...](https://blog.csdn.net/jieshaowang1229/article/details/50393458" \o "Spring的事务传播属性" \t "/Users/liu123/Documents\\x/_blank)博文[来自： jieshaowang1229的专栏](https://blog.csdn.net/jieshaowang1229" \t "/Users/liu123/Documents\\x/_blank)

#### [学会了这些技术，你离BAT大厂不远了](https://blog.csdn.net/z694644032/article/details/100084287" \o "学会了这些技术，你离BAT大厂不远了" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 16万+](https://blog.csdn.net/z694644032/article/details/100084287" \o "学会了这些技术，你离BAT大厂不远了" \t "/Users/liu123/Documents\\x/_blank)

[每一个程序员都有一个梦想，梦想着能够进入阿里、腾讯、字节跳动、百度等一线互联网公司，由于身边的环境等原因，不知道 BAT 等一线互联网公司使用哪些技术？或者该如何去学习这些技术？或者我该去哪些获取这些...](https://blog.csdn.net/z694644032/article/details/100084287" \o "学会了这些技术，你离BAT大厂不远了" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 平头哥的技术博文](https://blog.csdn.net/z694644032" \t "/Users/liu123/Documents\\x/_blank)

#### [spring-事务详细讲解（五）](https://blog.csdn.net/piaoslowly/article/details/81743640" \o "spring-事务详细讲解（五）" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 75](https://blog.csdn.net/piaoslowly/article/details/81743640" \o "spring-事务详细讲解（五）" \t "/Users/liu123/Documents\\x/_blank)

[Date: 2017-11-15 00:00:01spring-事务（五） 文章说明：本文不会只做复制粘贴的动作，在讨论一个问题的时候，理论部分总是词不达意，所以会从网上找合适的描述文章作为本篇的开...](https://blog.csdn.net/piaoslowly/article/details/81743640" \o "spring-事务详细讲解（五）" \t "/Users/liu123/Documents\\x/_blank)博文[来自： piaoslowly的专栏](https://blog.csdn.net/piaoslowly" \t "/Users/liu123/Documents\\x/_blank)

#### [从入门到精通，Java学习路线导航（附学习资源）](https://blog.csdn.net/java_sha/article/details/100895297" \o "从入门到精通，Java学习路线导航（附学习资源）" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2238](https://blog.csdn.net/java_sha/article/details/100895297" \o "从入门到精通，Java学习路线导航（附学习资源）" \t "/Users/liu123/Documents\\x/_blank)

[引言最近也有很多人来向我"请教"，他们大都是一些刚入门的新手，还不了解这个行业，也不知道从何学起，开始的时候非常迷茫，实在是每天回复很多人也很麻烦，所以在这里统一作个回复吧。Java学习路线当然，这里...](https://blog.csdn.net/java_sha/article/details/100895297" \o "从入门到精通，Java学习路线导航（附学习资源）" \t "/Users/liu123/Documents\\x/_blank)博文[来自： java\_sha的博客](https://blog.csdn.net/java_sha" \t "/Users/liu123/Documents\\x/_blank)

#### [感觉自己不会的东西太多了，不知道如何下手？](https://blog.csdn.net/hollis_chuang/article/details/102542347" \o "感觉自己不会的东西太多了，不知道如何下手？" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2万+](https://blog.csdn.net/hollis_chuang/article/details/102542347" \o "感觉自己不会的东西太多了，不知道如何下手？" \t "/Users/liu123/Documents\\x/_blank)

[GitHub8.8kStar的Java工程师成神之路，不来了解一下吗?GitHub8.8kStar的Java工程师成神之路，真的不来了解一下吗?GitHub8.8kStar的Java工程师成神之路，真...](https://blog.csdn.net/hollis_chuang/article/details/102542347" \o "感觉自己不会的东西太多了，不知道如何下手？" \t "/Users/liu123/Documents\\x/_blank)博文[来自： HollisChuang's Blog](https://blog.csdn.net/hollis_chuang" \t "/Users/liu123/Documents\\x/_blank)

#### [程序员实用工具网站](https://blog.csdn.net/m0_38106923/article/details/100130354" \o "程序员实用工具网站" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 19万+](https://blog.csdn.net/m0_38106923/article/details/100130354" \o "程序员实用工具网站" \t "/Users/liu123/Documents\\x/_blank)

[目录1、搜索引擎2、PPT3、图片操作4、文件共享5、应届生招聘6、程序员面试题库7、办公、开发软件8、高清图片、视频素材网站9、项目开源10、算法11、在线工具宝典大全程序员开发需要具备良好的信息检...](https://blog.csdn.net/m0_38106923/article/details/100130354" \o "程序员实用工具网站" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 不脱发的程序猿](https://blog.csdn.net/m0_38106923" \t "/Users/liu123/Documents\\x/_blank)

#### [成长的第一步是走出舒适区](https://blog.csdn.net/a724888/article/details/102548517" \o "成长的第一步是走出舒适区" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/a724888/article/details/102548517" \o "成长的第一步是走出舒适区" \t "/Users/liu123/Documents\\x/_blank)

[阅读本文大概需要2.8分钟。在温室里呆习惯了，就很难去适应室外环境，在一个圈子呆久了，就会把一切都会当成是理所当然，不想再去适应第二个圈子，即使他们可能更好。很多人觉......](https://blog.csdn.net/a724888/article/details/102548517" \o "成长的第一步是走出舒适区" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 程序员黄小斜](https://blog.csdn.net/a724888" \t "/Users/liu123/Documents\\x/_blank)

#### [终于鸿蒙微内核弄懂了-程序员和鼓励师的合作](https://blog.csdn.net/BEYONDMA/article/details/101915785" \o "终于鸿蒙微内核弄懂了-程序员和鼓励师的合作" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 8205](https://blog.csdn.net/BEYONDMA/article/details/101915785" \o "终于鸿蒙微内核弄懂了-程序员和鼓励师的合作" \t "/Users/liu123/Documents\\x/_blank)

[当鸿蒙OS宣布开源的时候，各种空洞的炒作，几乎把国产操作系统的技术本质掩盖了，虽然笔者没亲眼见过鸿蒙的代码，也没用方舟成功编译什么程序，不过当华为官宣鸿蒙将使用微内核的时候其实这款OS的风格就已经确定...](https://blog.csdn.net/BEYONDMA/article/details/101915785" \o "终于鸿蒙微内核弄懂了-程序员和鼓励师的合作" \t "/Users/liu123/Documents\\x/_blank)博文[来自： Python爱好者的专栏](https://blog.csdn.net/BEYONDMA" \t "/Users/liu123/Documents\\x/_blank)

#### [微信版花呗将上线；苹果在华支持以旧换新；谷歌推出 Flutter1.9 | 极客头条](https://blog.csdn.net/csdnnews/article/details/100815487" \o "微信版花呗将上线；苹果在华支持以旧换新；谷歌推出 Flutter1.9 | 极客头条" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 3386](https://blog.csdn.net/csdnnews/article/details/100815487" \o "微信版花呗将上线；苹果在华支持以旧换新；谷歌推出 Flutter1.9 | 极客头条" \t "/Users/liu123/Documents\\x/_blank)

[快来收听极客头条音频版吧，智能播报由标贝科技提供技术支持。「CSDN极客头条」，是从CSDN网站延伸至官方微信公众号的特别栏目，专注于一天业界事报道。风里雨里，我们将每天为朋友们，播报最新鲜有料的新闻...](https://blog.csdn.net/csdnnews/article/details/100815487" \o "微信版花呗将上线；苹果在华支持以旧换新；谷歌推出 Flutter1.9 | 极客头条" \t "/Users/liu123/Documents\\x/_blank)博文[来自： CSDN资讯](https://blog.csdn.net/csdnnews" \t "/Users/liu123/Documents\\x/_blank)

#### [十大优秀编程项目，让你的简历金光闪闪](https://blog.csdn.net/duxinshuxiaobian/article/details/102218308" \o "十大优秀编程项目，让你的简历金光闪闪" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2万+](https://blog.csdn.net/duxinshuxiaobian/article/details/102218308" \o "十大优秀编程项目，让你的简历金光闪闪" \t "/Users/liu123/Documents\\x/_blank)

[全文共3241字，预计学习时长6分钟被问到如何学习编程时，最常听到的问题就是：“有没有什么新项目的好点子？”一些老套的答案有：“做一个象棋游戏”或者“命令行界面”。这些答案没有错，但这些例子不符合现代...](https://blog.csdn.net/duxinshuxiaobian/article/details/102218308" \o "十大优秀编程项目，让你的简历金光闪闪" \t "/Users/liu123/Documents\\x/_blank)博文[来自： 读芯术的博客](https://blog.csdn.net/duxinshuxiaobian" \t "/Users/liu123/Documents\\x/_blank)

#### [spring事务传播属性和隔离级别](https://blog.csdn.net/songfeihu0810232/article/details/78910438" \o "spring事务传播属性和隔离级别" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 281](https://blog.csdn.net/songfeihu0810232/article/details/78910438" \o "spring事务传播属性和隔离级别" \t "/Users/liu123/Documents\\x/_blank)

[事务的传播属性（Propagation）Propagation支持7种不同的传播机制： REQUIRED：如果存在一个事务，则支持当前事务。如果没有事务则开启一个新的事务。（默认属性） SUPPORT...](https://blog.csdn.net/songfeihu0810232/article/details/78910438" \o "spring事务传播属性和隔离级别" \t "/Users/liu123/Documents\\x/_blank)博文[来自： Flytiger](https://blog.csdn.net/songfeihu0810232" \t "/Users/liu123/Documents\\x/_blank)

#### [Linux文件操作高频使用命令](https://blog.csdn.net/gexiaoyizhimei/article/details/100122368" \o "Linux文件操作高频使用命令" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 4987](https://blog.csdn.net/gexiaoyizhimei/article/details/100122368" \o "Linux文件操作高频使用命令" \t "/Users/liu123/Documents\\x/_blank)

[文章目录0.新建操作：1.查看操作2.删除操作3.复制操作4.移动操作：5.重命名操作：6.解压压缩操作 0.新建操作： mkdir abc #新建一个文件夹 touch abc.sh #新建一...](https://blog.csdn.net/gexiaoyizhimei/article/details/100122368" \o "Linux文件操作高频使用命令" \t "/Users/liu123/Documents\\x/_blank)博文

#### [史上最详细的IDEA优雅整合Maven+SSM框架（详细思路+附带源码）](https://blog.csdn.net/qq_44543508/article/details/100192558" \o "史上最详细的IDEA优雅整合Maven+SSM框架（详细思路+附带源码）" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 3万+](https://blog.csdn.net/qq_44543508/article/details/100192558" \o "史上最详细的IDEA优雅整合Maven+SSM框架（详细思路+附带源码）" \t "/Users/liu123/Documents\\x/_blank)

[网上很多整合SSM博客文章并不能让初探ssm的同学思路完全的清晰，可以试着关掉整合教程，摇两下头骨，哈一大口气，就在万事具备的时候，开整，这个时候你可能思路全无 ~中招了咩~ ，还有一些同学依旧在使用...](https://blog.csdn.net/qq_44543508/article/details/100192558" \o "史上最详细的IDEA优雅整合Maven+SSM框架（详细思路+附带源码）" \t "/Users/liu123/Documents\\x/_blank)博文

#### [分享靠写代码赚钱的一些门路](https://blog.csdn.net/lantian_123/article/details/101488841" \o "分享靠写代码赚钱的一些门路" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 6万+](https://blog.csdn.net/lantian_123/article/details/101488841" \o "分享靠写代码赚钱的一些门路" \t "/Users/liu123/Documents\\x/_blank)

[作者 mezod，译者 josephchang10如今，通过自己的代码去赚钱变得越来越简单，不过对很多人来说依然还是很难，因为他们不知道有哪些门路。今天给大家分享一个精彩......](https://blog.csdn.net/lantian_123/article/details/101488841" \o "分享靠写代码赚钱的一些门路" \t "/Users/liu123/Documents\\x/_blank)博文

#### [nginx学习，看这一篇就够了：下载、安装。使用：正向代理、反向代理、负载均衡。常用命令和配置文件](https://blog.csdn.net/qq_40036754/article/details/102463099" \o "nginx学习，看这一篇就够了：下载、安装。使用：正向代理、反向代理、负载均衡。常用命令和配置文件" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/qq_40036754/article/details/102463099" \o "nginx学习，看这一篇就够了：下载、安装。使用：正向代理、反向代理、负载均衡。常用命令和配置文件" \t "/Users/liu123/Documents\\x/_blank)

[文章目录前言一、nginx简介1. 什么是 nginx 和可以做什么事情2.Nginx 作为 web 服务器3. 正向代理4. 反向代理5. 动静分离6.动静分离二、Nginx 的安装三、 Ngin...](https://blog.csdn.net/qq_40036754/article/details/102463099" \o "nginx学习，看这一篇就够了：下载、安装。使用：正向代理、反向代理、负载均衡。常用命令和配置文件" \t "/Users/liu123/Documents\\x/_blank)博文

#### [【安全】Web渗透测试（全流程）](https://blog.csdn.net/qqchaozai/article/details/102515046" \o "【安全】Web渗透测试（全流程）" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2933](https://blog.csdn.net/qqchaozai/article/details/102515046" \o "【安全】Web渗透测试（全流程）" \t "/Users/liu123/Documents\\x/_blank)

[1 信息收集 1.1域名、IP、端口 域名信息查询：信息可用于后续渗透 IP信息查询：确认域名对应IP，确认IP是否真实，确认通信是否正常 端口信息查询：NMap扫描，确认开放端口 如果不...](https://blog.csdn.net/qqchaozai/article/details/102515046" \o "【安全】Web渗透测试（全流程）" \t "/Users/liu123/Documents\\x/_blank)博文

#### [MySQL数据库—SQL汇总](https://blog.csdn.net/qq_37969433/article/details/102637226" \o "MySQL数据库—SQL汇总" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2万+](https://blog.csdn.net/qq_37969433/article/details/102637226" \o "MySQL数据库—SQL汇总" \t "/Users/liu123/Documents\\x/_blank)

[一、准备 下文整理常见SQL语句的用法，使用MySQL5.7测试，参考了尚硅谷MySQL教程及用例。用例sql： 链接: https://pan.baidu.com/s/1tb3-12MRNFjV8d...](https://blog.csdn.net/qq_37969433/article/details/102637226" \o "MySQL数据库—SQL汇总" \t "/Users/liu123/Documents\\x/_blank)博文

#### [五款高效率黑科技神器工具，炸裂好用，省时间](https://blog.csdn.net/loongggdroid/article/details/102656177" \o "五款高效率黑科技神器工具，炸裂好用，省时间" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2万+](https://blog.csdn.net/loongggdroid/article/details/102656177" \o "五款高效率黑科技神器工具，炸裂好用，省时间" \t "/Users/liu123/Documents\\x/_blank)

[loonggg读完需要4分钟速读仅需2分钟感觉我好久好久没有给大家分享高质量的软件和插件了。今天周末，难得在家休息一下，痛下决心，分享一些我认为的高效率工具软件给大家。废......](https://blog.csdn.net/loongggdroid/article/details/102656177" \o "五款高效率黑科技神器工具，炸裂好用，省时间" \t "/Users/liu123/Documents\\x/_blank)博文

#### [黑客入门，从HTB开始](https://blog.csdn.net/neal1991/article/details/102656213" \o "黑客入门，从HTB开始" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2万+](https://blog.csdn.net/neal1991/article/details/102656213" \o "黑客入门，从HTB开始" \t "/Users/liu123/Documents\\x/_blank)

[Hack the box 是国外的一个靶机平台，里面的靶机包含多种系统类型，并且里面可以利用的漏洞类型多种多样，有很多靶机其实非常贴近实战情景。因此 HTB 是一个很好的学习渗透测试靶场。 之前...](https://blog.csdn.net/neal1991/article/details/102656213" \o "黑客入门，从HTB开始" \t "/Users/liu123/Documents\\x/_blank)博文

#### [动画：用动画给女朋友讲解 TCP 四次分手过程](https://blog.csdn.net/qq_36903042/article/details/102656641" \o "动画：用动画给女朋友讲解 TCP 四次分手过程" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 3万+](https://blog.csdn.net/qq_36903042/article/details/102656641" \o "动画：用动画给女朋友讲解 TCP 四次分手过程" \t "/Users/liu123/Documents\\x/_blank)

[作者 | 小鹿 来源 | 公众号：小鹿动画学编程 写在前边 大家好，我们又见面了，做为一个业余的动画师，上次的用动画的形式讲解 TCP 三次握手过程再各大平台收到了广大读者的喜爱，说文章有趣、有...](https://blog.csdn.net/qq_36903042/article/details/102656641" \o "动画：用动画给女朋友讲解 TCP 四次分手过程" \t "/Users/liu123/Documents\\x/_blank)博文

#### [程序员必须掌握的核心算法有哪些？](https://blog.csdn.net/m0_37907797/article/details/102661778" \o "程序员必须掌握的核心算法有哪些？" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 7万+](https://blog.csdn.net/m0_37907797/article/details/102661778" \o "程序员必须掌握的核心算法有哪些？" \t "/Users/liu123/Documents\\x/_blank)

[由于我之前一直强调数据结构以及算法学习的重要性，所以就有一些读者经常问我，数据结构与算法应该要学习到哪个程度呢？，说实话，这个问题我不知道要怎么回答你，主要取决于你想学习到哪些程度，不过针对这个问题，...](https://blog.csdn.net/m0_37907797/article/details/102661778" \o "程序员必须掌握的核心算法有哪些？" \t "/Users/liu123/Documents\\x/_blank)博文

#### [SQL基本语法入门 看这里就够了](https://blog.csdn.net/qq_42080073/article/details/102665305" \o "SQL基本语法入门 看这里就够了" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 8922](https://blog.csdn.net/qq_42080073/article/details/102665305" \o "SQL基本语法入门 看这里就够了" \t "/Users/liu123/Documents\\x/_blank)

[SQL执行顺序 第一步：执行FROM 第二步：WHERE条件过滤 第三步：GROUP BY 分组 第四步：执行SELECT 投影列 第五步：HAVING条件过滤 第六步：执行ORDER B...](https://blog.csdn.net/qq_42080073/article/details/102665305" \o "SQL基本语法入门 看这里就够了" \t "/Users/liu123/Documents\\x/_blank)博文

#### [面试最后一问：你有什么问题想问我吗？](https://blog.csdn.net/dyc87112/article/details/102676581" \o "面试最后一问：你有什么问题想问我吗？" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 3万+](https://blog.csdn.net/dyc87112/article/details/102676581" \o "面试最后一问：你有什么问题想问我吗？" \t "/Users/liu123/Documents\\x/_blank)

[尽管，我们之前分享了这么多关于面试的主题： 高薪必备的一些Spring Boot高级面试题 面试必问：设计模式遵循的面向对象设计原则！ 面试必问：怎么保证缓存与数据库的双写一致性？ 27道高频Spr...](https://blog.csdn.net/dyc87112/article/details/102676581" \o "面试最后一问：你有什么问题想问我吗？" \t "/Users/liu123/Documents\\x/_blank)博文

#### [python 程序员进阶之路：从新手到高手的100个模块](https://blog.csdn.net/xufive/article/details/102676755" \o "python 程序员进阶之路：从新手到高手的100个模块" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 5万+](https://blog.csdn.net/xufive/article/details/102676755" \o "python 程序员进阶之路：从新手到高手的100个模块" \t "/Users/liu123/Documents\\x/_blank)

[在知乎和CSDN的圈子里，经常看到、听到一些 python 初学者说，学完基础语法后，不知道该学什么，学了也不知道怎么用，一脸的茫然。近日，CSDN的公众号推送了一篇博客，题目叫做《迷思：Python...](https://blog.csdn.net/xufive/article/details/102676755" \o "python 程序员进阶之路：从新手到高手的100个模块" \t "/Users/liu123/Documents\\x/_blank)博文

#### [Python——画一棵漂亮的樱花树（不同种樱花+玫瑰+圣诞树喔）](https://blog.csdn.net/weixin_43943977/article/details/102691392" \o "Python——画一棵漂亮的樱花树（不同种樱花+玫瑰+圣诞树喔）" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 4万+](https://blog.csdn.net/weixin_43943977/article/details/102691392" \o "Python——画一棵漂亮的樱花树（不同种樱花+玫瑰+圣诞树喔）" \t "/Users/liu123/Documents\\x/_blank)

[最近翻到一篇知乎，上面有不少用Python（大多是turtle库）绘制的树图，感觉很漂亮，我整理了一下，挑了一些我觉得不错的代码分享给大家（这些我都测试过，确实可以生成） one 樱花树 动...](https://blog.csdn.net/weixin_43943977/article/details/102691392" \o "Python——画一棵漂亮的樱花树（不同种樱花+玫瑰+圣诞树喔）" \t "/Users/liu123/Documents\\x/_blank)博文

#### [还在收集资料？我这里有个github汇总](https://blog.csdn.net/lycyingO/article/details/102693626" \o "还在收集资料？我这里有个github汇总" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/lycyingO/article/details/102693626" \o "还在收集资料？我这里有个github汇总" \t "/Users/liu123/Documents\\x/_blank)

[原创：小姐姐味道（微信公众号ID：xjjdog），欢迎分享，转载请保留出处。国内程序员都喜欢收集资料，但是又不看，github是重灾区。更有莫名其妙fork的，让人不得要......](https://blog.csdn.net/lycyingO/article/details/102693626" \o "还在收集资料？我这里有个github汇总" \t "/Users/liu123/Documents\\x/_blank)博文

#### [阿里编程规范（简化版）](https://blog.csdn.net/lianghecai52171314/article/details/102711958" \o "阿里编程规范（简化版）" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 4016](https://blog.csdn.net/lianghecai52171314/article/details/102711958" \o "阿里编程规范（简化版）" \t "/Users/liu123/Documents\\x/_blank)

[阿里编程规范 及 阿里Java开发规约插件 Alibaba Java Coding Guidelines 统一规范标准将有助于提高行业编码规范化水平，帮助行业人员提高开发质量和效率、大大降低代码维护成...](https://blog.csdn.net/lianghecai52171314/article/details/102711958" \o "阿里编程规范（简化版）" \t "/Users/liu123/Documents\\x/_blank)博文

#### [我们这类学生并不是一无是处](https://blog.csdn.net/qq_38867330/article/details/102712407" \o "我们这类学生并不是一无是处" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 792](https://blog.csdn.net/qq_38867330/article/details/102712407" \o "我们这类学生并不是一无是处" \t "/Users/liu123/Documents\\x/_blank)

[我们这类学生并不是一无是处，更何况，个别学生还很聪明，真的很聪明。 主观能动性在于自己，自己的人生自己把握。其实并不需要给自己那么大的压力，毕竟，你三年后拿到的录取通知书和你当前所处学校的环境及师资...](https://blog.csdn.net/qq_38867330/article/details/102712407" \o "我们这类学生并不是一无是处" \t "/Users/liu123/Documents\\x/_blank)博文

#### [2020年秋招后端面经](https://blog.csdn.net/xianlvfan2224/article/details/102722298" \o "2020年秋招后端面经" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/xianlvfan2224/article/details/102722298" \o "2020年秋招后端面经" \t "/Users/liu123/Documents\\x/_blank)

[主要包含华为，网易互娱，广联达，科大讯飞，浦发，中兴，上海农商行这些已经拿offer的，还有京东（不小心把二面时间换了一下，等通知等三个月了），虾皮（一面挂），顺丰（sp专场一面之后没消息，秋招第一个...](https://blog.csdn.net/xianlvfan2224/article/details/102722298" \o "2020年秋招后端面经" \t "/Users/liu123/Documents\\x/_blank)博文

#### [Linux/C/C++ 不可错过的好书](https://blog.csdn.net/hyb612/article/details/102730648" \o "Linux/C/C++ 不可错过的好书" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/hyb612/article/details/102730648" \o "Linux/C/C++ 不可错过的好书" \t "/Users/liu123/Documents\\x/_blank)

[来源：公众号【编程珠玑】 作者：守望先生 ID：shouwangxiansheng 前言 经常有读者让我推荐书籍，这次我就把我私藏的计算机书单分享给你们！不过由于时间匆忙，不会进行更加详细...](https://blog.csdn.net/hyb612/article/details/102730648" \o "Linux/C/C++ 不可错过的好书" \t "/Users/liu123/Documents\\x/_blank)博文

#### [单点登录（SSO）](https://blog.csdn.net/u011225629/article/details/102756288" \o "单点登录（SSO）" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/u011225629/article/details/102756288" \o "单点登录（SSO）" \t "/Users/liu123/Documents\\x/_blank)

[一、SSO（单点登录）介绍 SSO英文全称Single SignOn，单点登录。SSO是在多个应用系统中，用户只需要登录一次就可以访问所有相互信任的应用系统。它包括可以将这次主要的登录映射到其他应用...](https://blog.csdn.net/u011225629/article/details/102756288" \o "单点登录（SSO）" \t "/Users/liu123/Documents\\x/_blank)博文

#### [漫话：什么是 https ?这应该是全网把 https 讲的最好的一篇文章了](https://blog.csdn.net/m0_37907797/article/details/102759257" \o "漫话：什么是 https ?这应该是全网把 https 讲的最好的一篇文章了" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2万+](https://blog.csdn.net/m0_37907797/article/details/102759257" \o "漫话：什么是 https ?这应该是全网把 https 讲的最好的一篇文章了" \t "/Users/liu123/Documents\\x/_blank)

[今天这篇文章，讲通过对话的形式，让你由浅入深着知道，为什么 Https 是安全的。 一、对称加密 一禅：在每次发送真实数据之前，服务器先生成一把密钥，然后先把密钥传输给客户端。之后服务器给客...](https://blog.csdn.net/m0_37907797/article/details/102759257" \o "漫话：什么是 https ?这应该是全网把 https 讲的最好的一篇文章了" \t "/Users/liu123/Documents\\x/_blank)博文

#### [史上最全的mysql基础教程](https://blog.csdn.net/weixin_45108087/article/details/102766281" \o "史上最全的mysql基础教程" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/weixin_45108087/article/details/102766281" \o "史上最全的mysql基础教程" \t "/Users/liu123/Documents\\x/_blank)

[启动与停止 启动mysql服务 sudo /usr/local/mysql/support-files/mysql.server start 停止mysql服务 sudo /usr/loc...](https://blog.csdn.net/weixin_45108087/article/details/102766281" \o "史上最全的mysql基础教程" \t "/Users/liu123/Documents\\x/_blank)博文

#### [为什么你学不会递归？告别递归，谈谈我的经验](https://blog.csdn.net/m0_37907797/article/details/102767860" \o "为什么你学不会递归？告别递归，谈谈我的经验" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2万+](https://blog.csdn.net/m0_37907797/article/details/102767860" \o "为什么你学不会递归？告别递归，谈谈我的经验" \t "/Users/liu123/Documents\\x/_blank)

[可能很多人在大一的时候，就已经接触了递归了，不过，我敢保证很多人初学者刚开始接触递归的时候，是一脸懵逼的，我当初也是，给我的感觉就是，递归太神奇了！ 可能也有一大部分人知道递归，也能看的懂递归，但在实...](https://blog.csdn.net/m0_37907797/article/details/102767860" \o "为什么你学不会递归？告别递归，谈谈我的经验" \t "/Users/liu123/Documents\\x/_blank)博文

#### [大学四年，分享看过的优质书籍](https://blog.csdn.net/qq_36903042/article/details/102770331" \o "大学四年，分享看过的优质书籍" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2万+](https://blog.csdn.net/qq_36903042/article/details/102770331" \o "大学四年，分享看过的优质书籍" \t "/Users/liu123/Documents\\x/_blank)

[数据结构与算法是我在大学里第一次接触到的，当时学了很多其他安卓、网页之类的，一开始就感觉纳闷，数据结构和算法学这个有啥用，再加上上的是一所野鸡大学，老师讲的也是模模糊糊，平时做项目、练习也几乎不用数据...](https://blog.csdn.net/qq_36903042/article/details/102770331" \o "大学四年，分享看过的优质书籍" \t "/Users/liu123/Documents\\x/_blank)博文

#### [有哪些让程序员受益终生的建议](https://blog.csdn.net/harvic880925/article/details/102774769" \o "有哪些让程序员受益终生的建议" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 3万+](https://blog.csdn.net/harvic880925/article/details/102774769" \o "有哪些让程序员受益终生的建议" \t "/Users/liu123/Documents\\x/_blank)

[从业五年多，辗转两个大厂，出过书，创过业，从技术小白成长为基层管理，联合几个业内大牛回答下这个问题，希望能帮到大家，记得帮我点赞哦。 敲黑板！！！读了这篇文章，你将知道如何才能进大厂，如何实现财务自...](https://blog.csdn.net/harvic880925/article/details/102774769" \o "有哪些让程序员受益终生的建议" \t "/Users/liu123/Documents\\x/_blank)博文

#### [最近程序员频繁被抓，如何避免面向监狱编程！？](https://blog.csdn.net/hollis_chuang/article/details/102776191" \o "最近程序员频繁被抓，如何避免面向监狱编程！？" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 6万+](https://blog.csdn.net/hollis_chuang/article/details/102776191" \o "最近程序员频繁被抓，如何避免面向监狱编程！？" \t "/Users/liu123/Documents\\x/_blank)

[最近，有关程序员因为参与某些项目开发导致被起诉，甚至被判刑的事件发生的比较多： 某程序员因为接了个外包，帮别人写了个软件，结果这个软件被用于赌博导致被抓。 某公司利用爬虫抓取用户信息，最后被发现，导致...](https://blog.csdn.net/hollis_chuang/article/details/102776191" \o "最近程序员频繁被抓，如何避免面向监狱编程！？" \t "/Users/liu123/Documents\\x/_blank)博文

#### [一文搞懂什么是TCP/IP协议](https://blog.csdn.net/petterp/article/details/102779131" \o "一文搞懂什么是TCP/IP协议" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2万+](https://blog.csdn.net/petterp/article/details/102779131" \o "一文搞懂什么是TCP/IP协议" \t "/Users/liu123/Documents\\x/_blank)

[什么是TCP/IP协议? 计算机与网络设备之间如果要相互通信,双方就必须基于相同的方法.比如如何探测到通信目标.由哪一边先发起通信,使用哪种语言进行通信,怎样结束通信等规则都需要事先确定.不同的硬件...](https://blog.csdn.net/petterp/article/details/102779131" \o "一文搞懂什么是TCP/IP协议" \t "/Users/liu123/Documents\\x/_blank)博文

#### [大学四年自学走来，这些私藏的实用工具/学习网站我贡献出来了](https://blog.csdn.net/m0_37907797/article/details/102781027" \o "大学四年自学走来，这些私藏的实用工具/学习网站我贡献出来了" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 11万+](https://blog.csdn.net/m0_37907797/article/details/102781027" \o "大学四年自学走来，这些私藏的实用工具/学习网站我贡献出来了" \t "/Users/liu123/Documents\\x/_blank)

[大学四年，看课本是不可能一直看课本的了，对于学习，特别是自学，善于搜索网上的一些资源来辅助，还是非常有必要的，下面我就把这几年私藏的各种资源，网站贡献出来给你们。主要有：电子书搜索、实用工具、在线视频...](https://blog.csdn.net/m0_37907797/article/details/102781027" \o "大学四年自学走来，这些私藏的实用工具/学习网站我贡献出来了" \t "/Users/liu123/Documents\\x/_blank)博文

#### [学习 Java 应该关注哪些网站？](https://blog.csdn.net/qing_gee/article/details/102781522" \o "学习 Java 应该关注哪些网站？" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/qing_gee/article/details/102781522" \o "学习 Java 应该关注哪些网站？" \t "/Users/liu123/Documents\\x/_blank)

[经常有一些读者问我：“二哥，学习 Java 应该关注哪些网站？”，我之前的态度一直是上知乎、上搜索引擎搜一下不就知道了。但读者对我这个态度很不满意，他们说，“我在问你，又不是问知乎，问搜索引擎。”你还...](https://blog.csdn.net/qing_gee/article/details/102781522" \o "学习 Java 应该关注哪些网站？" \t "/Users/liu123/Documents\\x/_blank)博文

#### [哪些 Java 知识不需要再学了](https://blog.csdn.net/qing_gee/article/details/102784577" \o "哪些 Java 知识不需要再学了" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/qing_gee/article/details/102784577" \o "哪些 Java 知识不需要再学了" \t "/Users/liu123/Documents\\x/_blank)

[张无忌在学太极拳的时候，他爹的师父张三丰告诫他一定要把之前所学习的武功全部忘掉，忘得越多就会学得越快。 同样的，自学 Java 的时候一定要先知道哪些 Java 知识不需要再学了，毕竟技术的更新迭代就...](https://blog.csdn.net/qing_gee/article/details/102784577" \o "哪些 Java 知识不需要再学了" \t "/Users/liu123/Documents\\x/_blank)博文

#### [大学四年，我把私藏的自学「学习网站/实用工具」都贡献出来了](https://blog.csdn.net/qq_36903042/article/details/102792114" \o "大学四年，我把私藏的自学「学习网站/实用工具」都贡献出来了" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 4万+](https://blog.csdn.net/qq_36903042/article/details/102792114" \o "大学四年，我把私藏的自学「学习网站/实用工具」都贡献出来了" \t "/Users/liu123/Documents\\x/_blank)

[在分享之前，先说说初学者如何学习编程，这个话题想必非常的重要，要学好编程，给你一些学习网站也好、实用工具也好，但前提是你知道如何去学习它。 见过很多初学者，以及小鹿我刚开始学习的时候，也是自己瞎摸索，...](https://blog.csdn.net/qq_36903042/article/details/102792114" \o "大学四年，我把私藏的自学「学习网站/实用工具」都贡献出来了" \t "/Users/liu123/Documents\\x/_blank)博文

#### [面试：史上最全多线程面试题 - （锁&内存模型&线程）](https://blog.csdn.net/xxyybs/article/details/102793030" \o "面试：史上最全多线程面试题 - （锁&内存模型&线程）" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 2936](https://blog.csdn.net/xxyybs/article/details/102793030" \o "面试：史上最全多线程面试题 - （锁&内存模型&线程）" \t "/Users/liu123/Documents\\x/_blank)

[多线程经典面试题59问。 1.什么是活锁、饥饿、无锁、死锁？ 死锁、活锁、饥饿是关于多线程是否活跃出现的运行阻塞障碍问题，如果线程出现 了这三种情况，即线程不再活跃，不能再正常地执行下去了。 死锁 ...](https://blog.csdn.net/xxyybs/article/details/102793030" \o "面试：史上最全多线程面试题 - （锁&内存模型&线程）" \t "/Users/liu123/Documents\\x/_blank)博文

#### [开题——我的第一个网站(1)](https://blog.csdn.net/yangwohenmai1/article/details/102794305" \o "开题——我的第一个网站(1)" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 5125](https://blog.csdn.net/yangwohenmai1/article/details/102794305" \o "开题——我的第一个网站(1)" \t "/Users/liu123/Documents\\x/_blank)

[每个程序员大概都有一个拥有属于自己网站的梦想，我就是这样。 梦想中属于自己的网站是一个有各种功能的大杂烩： 可以当做自己的随笔记事本； 可以在网页上查看自己感兴趣的内容； 不知道吃什么的时候可...](https://blog.csdn.net/yangwohenmai1/article/details/102794305" \o "开题——我的第一个网站(1)" \t "/Users/liu123/Documents\\x/_blank)博文

#### [中国麻将：世界上最早的区块链项目](https://blog.csdn.net/gao_chun/article/details/102795414" \o "中国麻将：世界上最早的区块链项目" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 3万+](https://blog.csdn.net/gao_chun/article/details/102795414" \o "中国麻将：世界上最早的区块链项目" \t "/Users/liu123/Documents\\x/_blank)

[中国麻将：世界上最早的区块链项目 最近区块链这个玩意又被市场搞的很是火热，相信大部分人都不太清楚这玩意到底是怎么样的一个概念，它来了，它来了，它到底是啥~ 国家都开始发文支持了，下面是一个通俗易懂的...](https://blog.csdn.net/gao_chun/article/details/102795414" \o "中国麻将：世界上最早的区块链项目" \t "/Users/liu123/Documents\\x/_blank)博文

#### [比特币原理详解](https://blog.csdn.net/zcg_741454897/article/details/102796022" \o "比特币原理详解" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/zcg_741454897/article/details/102796022" \o "比特币原理详解" \t "/Users/liu123/Documents\\x/_blank)

[一、什么是比特币 比特币是一种电子货币，是一种基于密码学的货币，在2008年11月1日由中本聪发表比特币白皮书，文中提出了一种去中心化的电子记账系统，我们平时的电子现金是银行来记账，因为银行的背后是...](https://blog.csdn.net/zcg_741454897/article/details/102796022" \o "比特币原理详解" \t "/Users/liu123/Documents\\x/_blank)博文

#### [Python 基础（一）：入门必备知识](https://blog.csdn.net/ityard/article/details/102807071" \o "Python 基础（一）：入门必备知识" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 1万+](https://blog.csdn.net/ityard/article/details/102807071" \o "Python 基础（一）：入门必备知识" \t "/Users/liu123/Documents\\x/_blank)

[Python 入门必备知识，你都掌握了吗？](https://blog.csdn.net/ityard/article/details/102807071" \o "Python 基础（一）：入门必备知识" \t "/Users/liu123/Documents\\x/_blank)博文

#### [兼职程序员一般可以从什么平台接私活？](https://blog.csdn.net/xiyue001/article/details/102816596" \o "兼职程序员一般可以从什么平台接私活？" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 3万+](https://blog.csdn.net/xiyue001/article/details/102816596" \o "兼职程序员一般可以从什么平台接私活？" \t "/Users/liu123/Documents\\x/_blank)

[这个问题我进行了系统性的总结，以下将进行言简意赅的说明和渠道提供，希望对各位小猿/小媛们有帮助~ 根据我们的经验，程序员兼职主要分为三种：兼职职位众包、项目整包和自由职业者驻场。 所谓的兼职职位众...](https://blog.csdn.net/xiyue001/article/details/102816596" \o "兼职程序员一般可以从什么平台接私活？" \t "/Users/liu123/Documents\\x/_blank)博文

#### [《吊打面试官》系列-Redis基础](https://blog.csdn.net/qq_35190492/article/details/102841400" \o "《吊打面试官》系列-Redis基础" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 8636](https://blog.csdn.net/qq_35190492/article/details/102841400" \o "《吊打面试官》系列-Redis基础" \t "/Users/liu123/Documents\\x/_blank)

[你知道的越多，你不知道的越多 点赞再看，养成习惯 前言 Redis在互联网技术存储方面使用如此广泛，几乎所有的后端技术面试官都要在Redis的使用和原理方面对小伙伴们进行360°的刁难。作为一个...](https://blog.csdn.net/qq_35190492/article/details/102841400" \o "《吊打面试官》系列-Redis基础" \t "/Users/liu123/Documents\\x/_blank)博文

#### [图解面试题：如何提高SQL查询的效率？](https://blog.csdn.net/zhongyangzhong/article/details/102849393" \o "图解面试题：如何提高SQL查询的效率？" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 4895](https://blog.csdn.net/zhongyangzhong/article/details/102849393" \o "图解面试题：如何提高SQL查询的效率？" \t "/Users/liu123/Documents\\x/_blank)

[【题目】我们公司的数据量非常大，需要的不仅仅是提取数据，要了解SQL方案优化的。一般在写SQL时需要注意哪些问题，可以提高查询的效率？【解题思路】数据量大的情况下，不同的SQL语句，消耗的时间相差很大...](https://blog.csdn.net/zhongyangzhong/article/details/102849393" \o "图解面试题：如何提高SQL查询的效率？" \t "/Users/liu123/Documents\\x/_blank)博文

#### [Python十大装B语法](https://blog.csdn.net/xufive/article/details/102856921" \o "Python十大装B语法" \t "/Users/liu123/Documents\\x/_blank)

[阅读数 4万+](https://blog.csdn.net/xufive/article/details/102856921" \o "Python十大装B语法" \t "/Users/liu123/Documents\\x/_blank)

[Python 是一种代表简单思想的语言，其语法相对简单，很容易上手。不过，如果就此小视 Python 语法的精妙和深邃，那就大错特错了。本文精心筛选了最能展现 Python 语法之精妙的十个知识点，并...](https://blog.csdn.net/xufive/article/details/102856921" \o "Python十大装B语法" \t "/Users/liu123/Documents\\x/_blank)博文

[python](https://blog.csdn.net/yilovexing/article/details/80577510" \t "/Users/liu123/Documents\\x/_blank) [json](https://blog.csdn.net/slwbcsdn/article/details/53458352" \t "/Users/liu123/Documents\\x/_blank) [java](https://blog.csdn.net/csdnnews/article/details/83753246" \t "/Users/liu123/Documents\\x/_blank) [mysql](https://blog.csdn.net/qq_35077512/article/details/88952519" \t "/Users/liu123/Documents\\x/_blank) [pycharm](https://blog.csdn.net/pdcfighting/article/details/80297499" \t "/Users/liu123/Documents\\x/_blank) [android](https://blog.csdn.net/sinyu890807/article/details/97142065" \t "/Users/liu123/Documents\\x/_blank) [linux](https://blog.csdn.net/gexiaoyizhimei/article/details/100122368" \t "/Users/liu123/Documents\\x/_blank) [json格式](https://download.csdn.net/download/xhg_gszs/10978826" \t "/Users/liu123/Documents\\x/_blank) [c#多人合作开发项目策略](https://www.csdn.net/gather_1e/NtSa1gksb3dubG9hZAO0O0OO0O0O.html" \t "/Users/liu123/Documents\\x/_blank) [c# cad 坐标转换](https://www.csdn.net/gather_11/Ntia1gksb3dubG9hZAO0O0OO0O0O.html" \t "/Users/liu123/Documents\\x/_blank) [c# 窗口内运行word](https://www.csdn.net/gather_1c/OtCa1gksb3dubG9hZAO0O0OO0O0O.html" \t "/Users/liu123/Documents\\x/_blank) [c#线程睡眠](https://www.csdn.net/gather_14/OtSa1gksb3dubG9hZAO0O0OO0O0O.html" \t "/Users/liu123/Documents\\x/_blank) [c# 静默打开打印机](https://www.csdn.net/gather_1d/MtTaAgtsZG93bmxvYWQO0O0O.html" \t "/Users/liu123/Documents\\x/_blank) [c#获取当前路径的上一级](https://www.csdn.net/gather_16/MtTaEgtsZG93bmxvYWQO0O0O.html" \t "/Users/liu123/Documents\\x/_blank) [c# 去除线条](https://www.csdn.net/gather_1c/MtTaMgtsZG93bmxvYWQO0O0O.html" \t "/Users/liu123/Documents\\x/_blank) [c# 取整倍数](https://www.csdn.net/gather_29/MtTaUgtsYmxvZwO0O0OO0O0O.html" \t "/Users/liu123/Documents\\x/_blank) [c#窗体最大化控件不变](https://www.csdn.net/gather_17/MtTacgtsZG93bmxvYWQO0O0O.html" \t "/Users/liu123/Documents\\x/_blank) [c# 程序更新程序案例](https://www.csdn.net/gather_16/MtTaggtsZG93bmxvYWQO0O0O.html" \t "/Users/liu123/Documents\\x/_blank)

©️2019 CSDN 皮肤主题: 大白 设计师: CSDN官方博客

[IMG_277](https://blog.csdn.net/piaoslowly)

[piaoslowly](https://blog.csdn.net/piaoslowly" \o "piaoslowly)

[TA的个人主页 >](https://me.csdn.net/piaoslowly" \t "/Users/liu123/Documents\\x/_blank)

[私信](https://im.csdn.net/im/main.html?userName=piaoslowly" \t "/Users/liu123/Documents\\x/_blank)

关注

[原创](https://blog.csdn.net/piaoslowly?t=1)

**[98](https://blog.csdn.net/piaoslowly?t=1)**

粉丝

**65**

获赞

**27**

评论

**4**

访问

**5万+**

等级:

周排名:

**[3万+](https://blog.csdn.net/rank/writing_rank" \t "/Users/liu123/Documents\\x/_blank)**

积分:

**1517**

总排名:

**[4万+](https://blog.csdn.net/rank/writing_rank_total" \t "/Users/liu123/Documents\\x/_blank)**

勋章:



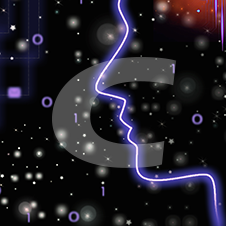


IMG_280

### 最新文章

* [java并发队列之SynchronousQueue(六)](https://blog.csdn.net/piaoslowly/article/details/85005603" \t "/Users/liu123/Documents\\x/_blank)
* [java并发队列之总结(八)](https://blog.csdn.net/piaoslowly/article/details/85005593" \t "/Users/liu123/Documents\\x/_blank)
* [java并发队列之非阻塞队列ConcurrentLinkedQueue(七)](https://blog.csdn.net/piaoslowly/article/details/85005585" \t "/Users/liu123/Documents\\x/_blank)
* [java并发队列之延时队列DelayQueue(五)](https://blog.csdn.net/piaoslowly/article/details/84993409" \t "/Users/liu123/Documents\\x/_blank)
* [java并发队列之优先级队列PriorityBlockingQueue(四)](https://blog.csdn.net/piaoslowly/article/details/84993407" \t "/Users/liu123/Documents\\x/_blank)

### 分类专栏

* [常见bug](https://blog.csdn.net/piaoslowly/article/category/6092253)
* [java1篇](https://blog.csdn.net/piaoslowly/article/category/6096162)
* [jvm11篇](https://blog.csdn.net/piaoslowly/article/category/7912127)
* [Java多线程全面解刨27篇](https://blog.csdn.net/piaoslowly/article/category/7912133)
* [Java集合源码分析8篇](https://blog.csdn.net/piaoslowly/article/category/7912134)
* [分布式7篇](https://blog.csdn.net/piaoslowly/article/category/7925347)
* [rabbitmq3篇](https://blog.csdn.net/piaoslowly/article/category/7931712)
* [spring8篇](https://blog.csdn.net/piaoslowly/article/category/7943289)
* [mysql1篇](https://blog.csdn.net/piaoslowly/article/category/7952968)
* [算法5篇](https://blog.csdn.net/piaoslowly/article/category/7952972)
* [java基础5篇](https://blog.csdn.net/piaoslowly/article/category/7956846)
* [redis3篇](https://blog.csdn.net/piaoslowly/article/category/8254951)
* [Linux2篇](https://blog.csdn.net/piaoslowly/article/category/8255057)
* [shell1篇](https://blog.csdn.net/piaoslowly/article/category/8255058)
* [tomcat1篇](https://blog.csdn.net/piaoslowly/article/category/8255121)

展开

### 归档

* [2018年12月8篇](https://blog.csdn.net/piaoslowly/article/month/2018/12)
* [2018年10月7篇](https://blog.csdn.net/piaoslowly/article/month/2018/10)
* [2018年8月62篇](https://blog.csdn.net/piaoslowly/article/month/2018/08)
* [2016年12月1篇](https://blog.csdn.net/piaoslowly/article/month/2016/12)
* [2016年8月2篇](https://blog.csdn.net/piaoslowly/article/month/2016/08)
* [2016年7月1篇](https://blog.csdn.net/piaoslowly/article/month/2016/07)
* [2016年6月11篇](https://blog.csdn.net/piaoslowly/article/month/2016/06)
* [2016年2月4篇](https://blog.csdn.net/piaoslowly/article/month/2016/02)
* [2016年1月2篇](https://blog.csdn.net/piaoslowly/article/month/2016/01)

展开

### 热门文章

* [关于多线程中的面试题](https://blog.csdn.net/piaoslowly/article/details/81562507)

阅读数 5929

* [top命令查看服务器性能](https://blog.csdn.net/piaoslowly/article/details/51759198)

阅读数 4995

* [redis实战全面讲解](https://blog.csdn.net/piaoslowly/article/details/81563579)

阅读数 4849

* [RabbitMQ的使用](https://blog.csdn.net/piaoslowly/article/details/81625687)

阅读数 4759

* [Spring-事务的传播属性（六）](https://blog.csdn.net/piaoslowly/article/details/81743658)

阅读数 3525

### 最新评论

* [java多线程-SimpleDat...](https://blog.csdn.net/piaoslowly/article/details/81476059" \l "comments" \t "/Users/liu123/Documents\\x/_blank)

[piaoslowly：](https://my.csdn.net/piaoslowly" \t "/Users/liu123/Documents\\x/_blank)[reply]whb3299065[/reply] 第二种方式粘贴的,确实写错了.其实就是在方法里面写SimpleDateFormat simpleDateFormat= new SimpleDateFormat("yyyy-MM-dd"); 而不是在类里面创建的.

* [java多线程-SimpleDat...](https://blog.csdn.net/piaoslowly/article/details/81476059" \l "comments" \t "/Users/liu123/Documents\\x/_blank)

[whb3299065：](https://my.csdn.net/whb3299065" \t "/Users/liu123/Documents\\x/_blank)第二种方式和你提出的问题代码写反了吧

* [RabbitMQ的使用](https://blog.csdn.net/piaoslowly/article/details/81625687" \l "comments" \t "/Users/liu123/Documents\\x/_blank)

[nidaye\_a：](https://my.csdn.net/nidaye_a" \t "/Users/liu123/Documents\\x/_blank)好。

* [top命令查看服务器性能](https://blog.csdn.net/piaoslowly/article/details/51759198" \l "comments" \t "/Users/liu123/Documents\\x/_blank)

[qq\_43336513：](https://my.csdn.net/qq_43336513" \t "/Users/liu123/Documents\\x/_blank)[reply]qq\_43473908[/reply] 1

IMG_296



程序人生

[](https://blog.csdn.net/csdnnews?utm_source=csdn_footer)

CSDN资讯

[kefu@csdn.net](mailto:webmaster@csdn.net" \t "/Users/liu123/Documents\\x/_blank)[QQ客服](http://wpa.b.qq.com/cgi/wpa.php?ln=1&key=XzgwMDE4MDEwNl80ODc3MzVfODAwMTgwMTA2XzJf" \t "/Users/liu123/Documents\\x/_blank)

[客服论坛](http://bbs.csdn.net/forums/Service" \t "/Users/liu123/Documents\\x/_blank)400-660-0108

工作时间 8:30-22:00

**[关于我们](https://www.csdn.net/company/index.html" \l "about" \t "/Users/liu123/Documents\\x/_blank)[招聘](https://www.csdn.net/company/index.html" \l "recruit" \t "/Users/liu123/Documents\\x/_blank)[广告服务](https://www.csdn.net/company/index.html" \l "contact" \t "/Users/liu123/Documents\\x/_blank)** **[网站地图](https://www.csdn.net/gather/A" \t "/Users/liu123/Documents\\x/_blank)**

[百度提供站内搜索](https://zn.baidu.com/cse/home/index" \t "/Users/liu123/Documents\\x/_blank) [京ICP备19004658号](http://www.miibeian.gov.cn/publish/query/indexFirst.action" \t "/Users/liu123/Documents\\x/_blank)

©1999-2019 北京创新乐知网络技术有限公司

[经营性网站备案信息](https://csdnimg.cn/cdn/content-toolbar/csdn-ICP.png" \t "/Users/liu123/Documents\\x/_blank)[网络110报警服务](http://www.cyberpolice.cn/" \t "/Users/liu123/Documents\\x/_blank)

[北京互联网违法和不良信息举报中心](http://www.bjjubao.org/" \t "/Users/liu123/Documents\\x/_blank)

[中国互联网举报中心](http://www.12377.cn/" \t "/Users/liu123/Documents\\x/_blank)[家长监护](https://download.csdn.net/index.php/tutelage/" \t "/Users/liu123/Documents\\x/_blank)[版权申诉](https://blog.csdn.net/blogdevteam/article/details/90369522" \t "/Users/liu123/Documents\\x/_blank)

0

IMG_299举报IMG_300