

Industrial strength Java/JEE Career Companion to open more doors

search here ...

Go

[Home](#)[Java FAQs](#)[600+ Java Q&As](#)[Career](#)[Tutorials](#)[Member](#)[Why?](#)[Can u Debug?](#)[Java 8 ready?](#)[Top X](#)[Productivity Tools](#)[Judging Experience?](#)[Home](#) › [Interview](#) › [JEE Interview Q&A](#) › [JDBC](#) › [NamedParameterStatement versus PreparedStatement](#)

NamedParameterStatement versus PreparedStatement

Posted on [January 5, 2015](#) by [Arulkumaran Kumaraswamipillai](#) — [No Comments](#) ↓

Q. When will you use **NamedParameterStatement** over **PreparedStatement**?

A.

Reason 1: *PreparedStatement* uses anonymous parameters that are accessed by index, which is ok when you have 2 or 3 parameters, but for larger queries with many parameters it is more difficult to keep track of the indices. The developer has to count the number of question marks.

```

1 String sql = "select * from people where (first_n
2 PreparedStatement p = con.prepareStatement(sql);
3 p.setString(1, name);
4 p.setString(2, name);
5 p.setString(3, address);
6

```

600+ Full Stack Java/JEE Interview Q&As ♥Free ♦FAQs

[open all](#) | [close all](#)

✚ [Ice Breaker Interview](#)

✚ [Core Java Interview C](#)

✚ [JEE Interview Q&A \(3](#)

✚ [JEE Overview \(2\)](#)

✚ [Web basics \(8\)](#)

✚ [WebService \(11\)](#)

✚ [JPA \(2\)](#)

✚ [JTA \(1\)](#)

✚ [JDBC \(4\)](#)

✚ [♦ 12 FAQ JDBC](#)

✚ [JDBC Overview](#)

✚ [NamedParamete](#)

✚ [Spring, JavaCon](#)

✚ [JMS \(5\)](#)

✚ [JMX \(3\)](#)

✚ [JNDI and LDAP \(1\)](#)

✚ [Pressed for time? Jav](#)

✚ [SQL, XML, UML, JSC](#)

✚ [Hadoop & BigData Int](#)

✚ [Java Architecture Inte](#)

✚ [Scala Interview Q&As](#)

The above query will require you to renumber the indices, and it can be improved as shown below.

```

1 String sql = "select * from people where (first_n
2 PreparedStatement p = con.prepareStatement(sql);
3 int i = 1;
4 p.setString(i++, name);
5 p.setString(i++, name);
6 p.setString(i++, address);
7

```

The NamedParameter is even more cleaner.

```

1 String sql = "select * from people where (first_n
2 NamedParameterStatement p = new NamedParameterSta
3 p.setString("name", name);
4 p.setString("address", address);
5

```

Reason 2: In some cases parameters make your query more readable when you have combination of parameters and database functions like getdate(), etc. The following example is Spring jdbc based to use parameter names.

```

1
2
3 public int insertOfflineSuperEquityRequest(Trade
4
5     String sql = "insert into trade_request ( ac
6     " security_code,create_date,create_id,update
7     " values (:acc_code,'SYSTEM_A', :ref, :ext
8
9     //Spring JDBC Named Parameter class
10    NamedParameterJdbcTemplate npJdbcTemplate =
11
12
13    Map<String, Object> parameters = new HashMap
14    parameters.put("acc_code", td.accountCode);
15    parameters.put("ref", td.getReference());
16    parameters.put("ext_ref", td.getExternalRefe
17    parameters.put("security", td.getSecurityCod
18
19
20
21    int updateCount = npJdbcTemplate.update(sql,
22    return updateCount;
23 }
24

```

[Spring, Hibernate, & I](#)
[Testing & Profiling/Sa](#)
[Other Interview Q&A 1](#)
[Free Java Interview](#)

16 Technical Key Areas

[open all](#) | [close all](#)

- [Best Practice \(6\)](#)
- [Coding \(26\)](#)
- [Concurrency \(6\)](#)
- [Design Concepts \(7\)](#)
- [Design Patterns \(11\)](#)
- [Exception Handling \(3\)](#)
- [Java Debugging \(21\)](#)
- [Judging Experience It](#)
- [Low Latency \(7\)](#)
- [Memory Management](#)
- [Performance \(13\)](#)
- [QoS \(8\)](#)
- [Scalability \(4\)](#)
- [SDLC \(6\)](#)
- [Security \(13\)](#)
- [Transaction Managen](#)

80+ step by step Java Tutorials

[open all](#) | [close all](#)

- [Setting up Tutorial \(6\)](#)
- [Tutorial - Diagnosis \(2\)](#)
- [Akka Tutorial \(9\)](#)
- [Core Java Tutorials \(2\)](#)
- [Hadoop & Spark Tuto](#)
- [JEE Tutorials \(19\)](#)
- [Scala Tutorials \(1\)](#)
- [Spring & Hibernate Ti](#)

As you can see, it has a combination of named parameters (:acc_code, :ref, :ext_ref, and security), a constant (i.e. SYSTEM_A), and Sybase database functions like getDate() to get current date time and suser_name() to get the user id.

Popular Posts

♦ 11 Spring boot interview questions & answers

825 views

♦ Q11-Q23: Top 50+ Core on Java OOP Interview Questions & Answers

766 views

18 Java scenarios based interview Questions and Answers

400 views

001A: ♦ 7+ Java integration styles & patterns interview questions & answers

388 views

01b: ♦ 13 Spring basics Q8 – Q13 interview questions & answers

295 views

♦ 7 Java debugging interview questions & answers

293 views

01: ♦ 15 Ice breaker questions asked 90% of the time in Java job interviews with hints

285 views

♦ 10 ERD (Entity-Relationship Diagrams) Interview Questions and Answers

279 views

♦ Q24-Q36: Top 50+ Core on Java classes, interfaces and generics interview questions & answers

239 views

001B: ♦ Java architecture & design concepts interview questions & answers

201 views

[Tools Tutorials \(19\)](#)
[Other Tutorials \(45\)](#)

100+ Java pre-interview coding tests

[open all](#) | [close all](#)
[Can you write code? \(](#)
[♦ Complete the given](#)
[Converting from A to I](#)
[Designing your classe](#)
[Java Data Structures](#)
[Passing the unit tests](#)
[What is wrong with th](#)
[Writing Code Home A](#)
[Written Test Core Jav](#)
[Written Test JEE \(1\)](#)

How good are your?

[open all](#) | [close all](#)
[Career Making Know-](#)
[Job Hunting & Resum](#)

Bio

Latest Posts

**Arulkumar
Kumaraswamipillai**



Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via Amazon.com in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site. **945+** paid members. [join my LinkedIn Group](#). [Reviews](#)



About [Arulkumaran Kumaraswamipillai](#)

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via Amazon.com in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site. **945+** paid members. [join my LinkedIn Group](#). [Reviews](#)

◀ Spring batch industrial strength tutorial – part2

Java OpenCSV tutorial to work with CSV files ▶

Posted in JDBC

Leave a Reply

Logged in as geethika. [Log out?](#)

Comment

Empowers you to open more doors, and fast-track

Technical Know Hows

☀ [Java generics in no time](#) ☀ [Top 6 tips to transforming your thinking from OOP to FP](#) ☀ [How does a HashMap internally work? What is a hashing function?](#)
☀ [10+ Java String class interview Q&As](#) ☀ [Java auto un/boxing benefits & caveats](#) ☀ [Top 11 slacknesses that can come back and bite you as an experienced Java developer or architect](#)

Non-Technical Know Hows

☀ [6 Aspects that can motivate you to fast-track your career & go places](#) ☀ [Are you reinventing yourself as a Java developer?](#) ☀ [8 tips to safeguard your Java career against offshoring](#) ☀ [My top 5 career mistakes](#)

Prepare to succeed

☀ [Turn readers of your Java CV go from “Blah blah” to “Wow”?](#) ☀ [How to prepare for Java job interviews?](#) ☀ [16 Technical Key Areas](#) ☀ [How to choose from multiple Java job offers?](#)

© Disclaimer

The contents in this Java-Success are copy righted. The author has the right to correct or enhance the current content without any prior notice.

These are general advice only, and one needs to take his/her own circumstances into consideration. The author will not be held liable for any damages caused or alleged to be caused either directly or indirectly by these materials and resources. Any trademarked names or labels used in this blog remain the property of their respective trademark owners. No guarantees are made regarding the accuracy or usefulness of content, though I do make an effort to be accurate. Links to external sites do not imply endorsement of the linked-to sites.

