[**GP Coder**](https://gpcoder.com/)

*Trang chia sẻ kiến thức lập trình Java*

[**Trang chủ**](https://gpcoder.com/)[**Java Core**](https://gpcoder.com/category/java-core/)[**Multi-Thread**](https://gpcoder.com/category/java-core/multi-thread/)**Luồng Daemon (Daemon Thread) trong Java**

**Luồng Daemon (Daemon Thread) trong Java**

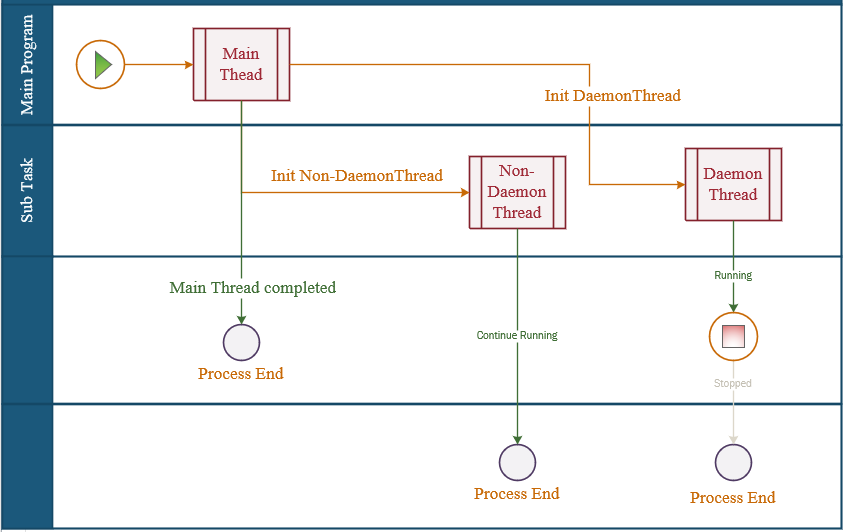
*Đăng vào 14/02/2018 . Được đăng bởi*[**GP Coder**](https://gpcoder.com/author/gpadmin/)*.****10568****Lượt xem . Toàn màn hình*

**Nội dung**[[Ẩn](https://gpcoder.com/3513-luong-daemon-daemon-thread-trong-java/)]

* [1 Luồng Daemon (Daemon Thread) là gì?](https://gpcoder.com/3513-luong-daemon-daemon-thread-trong-java/#Luong_Daemon_Daemon_Thread_la_gi)
* [2 Luồng Daemon thường dùng làm gì?](https://gpcoder.com/3513-luong-daemon-daemon-thread-trong-java/#Luong_Daemon_thuong_dung_lam_gi)
* [3 Ví dụ minh họa](https://gpcoder.com/3513-luong-daemon-daemon-thread-trong-java/#Vi_du_minh_hoa)
  + [3.1 Tạo một luồng WorkingThread](https://gpcoder.com/3513-luong-daemon-daemon-thread-trong-java/#Tao_mot_luong_WorkingThread)
  + [3.2 Chương trình Non-Daemon Thread](https://gpcoder.com/3513-luong-daemon-daemon-thread-trong-java/#Chuong_trinh_Non-Daemon_Thread)
  + [3.3 Chương trình Daemon Thread](https://gpcoder.com/3513-luong-daemon-daemon-thread-trong-java/#Chuong_trinh_Daemon_Thread)

**Luồng Daemon (Daemon Thread) là gì?**

Java chia thread làm 2 loại: một loại thông thường và Daemon Thread. Chúng chỉ khác nhau ở cách thức ngừng hoạt động. Trong một chương trình các luồng thông thường và luồng Daemon chạy song song với nhau. Khi tất cả các luồng thông thường kết thúc, mọi luồng Daemon cũng sẽ bị kết thúc theo bất kể nó đang làm việc gì.

[](https://gpcoder.com/wp-content/uploads/2018/02/daemon-thread.png)

Sử dụng **setDaemon**(boolean) để xác định một luồng là Daemon hoặc không. Chú ý, bạn chỉ có thể gọi hàm setDeamon(boolean) khi thread chưa được chạy. Điều đó có nghĩa là khi thread đã chạy bạn không thể chuyển luồng từ non-daemon sang daemon và ngược lại. Khi bạn cố gắng thay đổi trạng thái của luồng, một ngoại lệ **IllegalThreadStateException** được ném ra và luồng kết thúc xử lý.

Khi một luồng mới được tạo ra, nó được thừa hưởng đặc tính daemon từ luồng cha.  Như vậy khi bạn tạo một luồng trong hàm main của 1 class nó vốn là luồng non-daemon , vì vậy thread tạo ra mặc định cũng là non-daemon . Như vậy nếu bạn tạo một luồng mới trong một luồng Daemon, mặc định nó cũng sẽ là Daemon .

**Luồng Daemon thường dùng làm gì?**

Một trong các luồng Deamon quan trọng của Java đó là luồng gom rác, nghĩa là gom các tài nguyên không còn sử dụng để giải phóng bộ nhớ. Khi tất cả các luồng người dùng không còn hoạt động nữa luồng gom rác cũng bị dừng theo.

**Ví dụ minh họa**

**Tạo một luồng WorkingThread**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21 | **package** com.gpcoder.daemonthread;    **public** **class** WorkingThread **implements** Runnable {        @Override  **public** **void** run() {  **while** (**true**) {              processSomething();          }      }    **private** **void** processSomething() {  **try** {              System.out.println("Processing working thread");              Thread.sleep(500);          } **catch** (InterruptedException e) {              e.printStackTrace();          }      }    } |

**Chương trình Non-Daemon Thread**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | **package** com.gpcoder.daemonthread;    **public** **class** NonDaemonThreadTest {  **public** **static** **void** main(String[] args) **throws** InterruptedException {          Thread dt = **new** Thread(**new** WorkingThread(), "My Non-Daemon Thread");          dt.start();            // continue program          Thread.sleep(3000);          System.out.println(">><< Finishing main program");      }  } |

Thực thi chương trình trên:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | Processing working thread  Processing working thread  Processing working thread  Processing working thread  Processing working thread  Processing working thread  >><< Finishing main program  Processing working thread  Processing working thread  Processing working thread  .... |

Như bạn thấy luồng Non-DaemonThread chạy song song với MainThread. Khi MainThread kết thúc thì luồng Non-DaemonThread vẫn còn tiếp tục xử lý cho đến khi hoàn thành.

**Chương trình Daemon Thread**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | **package** com.gpcoder.daemonthread;    **public** **class** DaemonThreadTest {    **public** **static** **void** main(String[] args) **throws** InterruptedException {          Thread dt = **new** Thread(**new** WorkingThread(), "My Daemon Thread");          dt.setDaemon(**true**);          dt.start();            // continue program          Thread.sleep(3000);          System.out.println(">><< Finishing main program");      }    } |

Thực thi chương trình trên:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | Processing working thread  Processing working thread  Processing working thread  Processing working thread  Processing working thread  Processing working thread  >><< Finishing main program |

Như bạn thấy DaemonThread chạy song song với MainThread. Khi MainThread kết thúc thì tất cả DaemonThread cũng kết thúc.

**4.8**

19

**Nếu bạn thấy hay thì hãy chia sẻ bài viết cho mọi người nhé!**

SHARES

*Chuyên mục:*[**Multi-Thread**](https://gpcoder.com/category/java-core/multi-thread/)*Được gắn thẻ:*[**Multithreading**](https://gpcoder.com/tag/multithreading/)

[Lập trình đa luồng trong Java (Java Multi-threading)](https://gpcoder.com/3484-lap-trinh-da-luong-trong-java-java-multi-threading/)

[Đồng bộ hóa các luồng trong Java](https://gpcoder.com/3514-dong-bo-hoa-cac-luong-trong-java/)

**Có thể bạn muốn xem:**

* [Lập trình đa luồng với CompletableFuture trong Java 8](https://gpcoder.com/4064-lap-trinh-da-luong-voi-completablefuture-trong-java-8/) **(19/07/2018)**
* [Đồng bộ hóa các luồng trong Java](https://gpcoder.com/3514-dong-bo-hoa-cac-luong-trong-java/) **(20/02/2018)**
* [Sử dụng CyclicBarrier trong Java](https://gpcoder.com/3603-su-dung-cyclicbarrier-trong-java/) **(13/03/2018)**
* [Thực thi nhiều tác vụ cùng lúc như thế nào trong Java?](https://gpcoder.com/5100-thuc-thi-nhieu-tac-vu-cung-luc-nhu-the-nao-trong-java/) **(01/02/2019)**
* [Vấn đề Nhà sản xuất (Producer) – Người tiêu dùng (Consumer) và đồng bộ hóa các luồng trong Java](https://gpcoder.com/6180-van-de-nguoi-san-xuat-producer-nguoi-tieu-dung-consumer-va-dong-bo-hoa-cac-luong-trong-java/) **(23/09/2019)**

**Bình luận**

5 bình luận

**Tìm kiếm**

Top of Form



Bottom of Form

**Bài viết mới**

* [Giới thiệu CloudAMQP – Một RabbitMQ server trên Cloud](https://gpcoder.com/7219-gioi-thieu-cloudamqp-mot-rabbitmq-server-tren-cloud/) 02/10/2020
* [Kết nối RabbitMQ sử dụng Web STOMP Plugin](https://gpcoder.com/7167-ket-noi-rabbitmq-su-dung-web-stomp-plugin/) 19/06/2020
* [Sử dụng publisher confirm trong RabbitMQ](https://gpcoder.com/6947-su-dung-publisher-confirm-trong-rabbitmq/) 16/06/2020
* [Sử dụng Dead Letter Exchange trong RabbitMQ](https://gpcoder.com/7095-su-dung-dead-letter-exchange-trong-rabbitmq/) 13/06/2020
* [Sử dụng Alternate Exchange trong RabbitMQ](https://gpcoder.com/6944-alternate-exchange-trong-rabbitmq/) 10/06/2020

**Xem nhiều**

* [Hướng dẫn Java Design Pattern – Factory Method *(67525 lượt xem)*](https://gpcoder.com/4352-huong-dan-java-design-pattern-factory-method/)
* [Hướng dẫn Java Design Pattern – Singleton *(66169 lượt xem)*](https://gpcoder.com/4190-huong-dan-java-design-pattern-singleton/)
* [Lập trình đa luồng trong Java (Java Multi-threading) *(61549 lượt xem)*](https://gpcoder.com/3484-lap-trinh-da-luong-trong-java-java-multi-threading/)
* [Giới thiệu Design Patterns *(58620 lượt xem)*](https://gpcoder.com/4164-gioi-thieu-design-patterns/)
* [Giới thiệu về Stream API trong Java 8 *(55261 lượt xem)*](https://gpcoder.com/3923-gioi-thieu-ve-stream-api-trong-java-8/)

**Nội dung bài viết**

* [1 Luồng Daemon (Daemon Thread) là gì?](https://gpcoder.com/3513-luong-daemon-daemon-thread-trong-java/#Luong_Daemon_Daemon_Thread_la_gi)
* [2 Luồng Daemon thường dùng làm gì?](https://gpcoder.com/3513-luong-daemon-daemon-thread-trong-java/#Luong_Daemon_thuong_dung_lam_gi)
* [3 Ví dụ minh họa](https://gpcoder.com/3513-luong-daemon-daemon-thread-trong-java/#Vi_du_minh_hoa)
  + [3.1 Tạo một luồng WorkingThread](https://gpcoder.com/3513-luong-daemon-daemon-thread-trong-java/#Tao_mot_luong_WorkingThread)
  + [3.2 Chương trình Non-Daemon Thread](https://gpcoder.com/3513-luong-daemon-daemon-thread-trong-java/#Chuong_trinh_Non-Daemon_Thread)
  + [3.3 Chương trình Daemon Thread](https://gpcoder.com/3513-luong-daemon-daemon-thread-trong-java/#Chuong_trinh_Daemon_Thread)

**Lưu trữ**



**Thẻ đánh dấu**

[Annotation](https://gpcoder.com/tag/annotation/) [Authentication](https://gpcoder.com/tag/authentication/) [Basic Java](https://gpcoder.com/tag/basic-java/) [Behavior Pattern](https://gpcoder.com/tag/behavior-pattern/) [Collection](https://gpcoder.com/tag/collection/) [Creational Design Pattern](https://gpcoder.com/tag/creational-design-pattern/) [Cấu trúc điều khiển](https://gpcoder.com/tag/cau-truc-dieu-khien/) [Database](https://gpcoder.com/tag/database/) [Dependency Injection](https://gpcoder.com/tag/dependency-injection/) [Design pattern](https://gpcoder.com/tag/design-pattern/) [Eclipse](https://gpcoder.com/tag/eclipse/) [Exception](https://gpcoder.com/tag/exception/) [Executor Service](https://gpcoder.com/tag/executor-service/) [Google Guice](https://gpcoder.com/tag/google-guice/) [Gson](https://gpcoder.com/tag/gson/) [Hibernate](https://gpcoder.com/tag/hibernate/) [How to](https://gpcoder.com/tag/how-to/) [Interceptor](https://gpcoder.com/tag/interceptor/) [IO](https://gpcoder.com/tag/io/) [Jackson](https://gpcoder.com/tag/jackson/) [Java 8](https://gpcoder.com/tag/java-8/) [Java Core](https://gpcoder.com/tag/java-core/) [JDBC](https://gpcoder.com/tag/jdbc/) [JDK](https://gpcoder.com/tag/jdk/) [Jersey](https://gpcoder.com/tag/jersey/) [JMS](https://gpcoder.com/tag/jms/) [JPA](https://gpcoder.com/tag/jpa/) [json](https://gpcoder.com/tag/json/) [JUnit](https://gpcoder.com/tag/junit/) [JWT](https://gpcoder.com/tag/jwt/) [Message Queue](https://gpcoder.com/tag/message-queue/) [Mockito](https://gpcoder.com/tag/mockito/) [Multithreading](https://gpcoder.com/tag/multithreading/) [OOP](https://gpcoder.com/tag/oop/) [Performance](https://gpcoder.com/tag/performance/) [PowerMockito](https://gpcoder.com/tag/powermockito/) [RabbitMQ](https://gpcoder.com/tag/rabbitmq/) [Reflection](https://gpcoder.com/tag/reflection/) [Report](https://gpcoder.com/tag/report/) [REST](https://gpcoder.com/tag/rest/) [SOAP](https://gpcoder.com/tag/soap/) [Structuaral Pattern](https://gpcoder.com/tag/structuaral-pattern/) [Thread Pool](https://gpcoder.com/tag/thread-pool/) [Unit Test](https://gpcoder.com/tag/unit-test/) [Webservice](https://gpcoder.com/tag/webservice/)

**Liên kết**

* [Clean Code](https://gpcoder.gitbook.io/clean-code/)
* [JavaTpoint](https://www.javatpoint.com/)
* [Refactoring Guru](https://refactoring.guru/)
* [Source Making](https://sourcemaking.com/)
* [TutorialsPoint](http://www.tutorialspoint.com/)
* [W3Schools Online Web Tutorials](https://www.w3schools.com/)

**Giới thiệu**

[GP Coder](https://gpcoder.com/) là trang web cá nhân, được thành lập với mục đích lưu trữ, chia sẽ kiến thức đã học và làm việc của tôi. Các bài viết trên trang này chủ yếu về ngôn ngữ Java và các công nghệ có liên quan đến Java như: Spring, JSF, Web Services, Unit Test, Hibernate, SQL, ...  
Hi vọng góp được chút ít công sức cho sự phát triển cộng đồng Coder Việt.  
  
[Tìm kiếm các bài viết của GP Coder với Google Search](https://gpcoder.com/tim-kiem-gp-coder-voi-google-search/)

**Liên hệ**

Các bạn có thể liên hệ với tôi thông qua:

* Trang [liên hệ](https://gpcoder.com/lien-he)
* Linkedin: [gpcoder](https://www.linkedin.com/in/gpcoder/)
* Email: [contact@gpcoder.com](mailto:contact@gpcoder.com?Subject=Li%C3%AAn%20h%E1%BB%87%20v%E1%BB%81...)
* Skype: [ptgiang56it](skype:ptgiang56it?chat)

**Follow me**

Copyright 2023 © [GP Coder](https://gpcoder.com/) · All Rights Reserved · [Giới thiệu](https://gpcoder.com/gioi-thieu) · [Chính sách](https://gpcoder.com/chinh-sach) · [Điều khoản](https://gpcoder.com/dieu-khoan) · [Liên hệ](https://gpcoder.com/lien-he) ·

1