

The Executor (thread pool)

Table of Contents

- [Introduction](#)
- [Attributes](#)
 1. [Common Attributes](#)
 2. [Standard Implementation](#)

Introduction

The **Executor** represents a thread pool that can be shared between components in Tomcat. Historically there has been a thread pool per connector created but this allows you to share a thread pool, between (primarily) connector but also other components when those get configured to support executors

The executor has to implement the `org.apache.catalina.Executor` interface.

The executor is a nested element to the [Service](#) element. And in order for it to be picked up by the connectors, the Executor element has to appear prior to the Connector element in server.xml

Attributes

Common Attributes

All implementations of **Executor** support the following attributes:

Attribute	Description
<code>className</code>	The class of the implementation. The implementation has to implement the <code>org.apache.catalina.Executor</code> interface. This interface ensures that the object can be referenced through its <code>name</code> attribute and that implements Lifecycle, so that it can be started and stopped with the container. The <u>default value for the className</u> is <code>org.apache.catalina.core.StandardThreadExecutor</code>
<code>name</code>	The name used to reference this pool in other places in server.xml. The name is required and must be unique.

Standard Implementation

The default implementation supports the following attributes:

Attribute	Description
<code>threadPriority</code>	(int) The thread priority for threads in the executor, the default is 5 (the value of the <code>Thread.NORM_PRIORITY</code> constant)
<code>daemon</code>	(boolean) Whether the threads should be daemon threads or not, the default is <code>true</code>

maxThreads：是线程池的最大活动线程数，可以把它理解为线程池最多能同时开启的线程数
 minSpareThreads：是保留的最小活动闲线程数，也就是说不管有没有请求，活动线程数最少是25
 maxIdleTime：是指超过25的线程最大空闲时间，超过这个时间线程还没有使用将会被关闭
 maxQueueSize：是指当已在有25个线程在运行中了，如果再有请求进来会将它放到队列中，默认是无上限

namePrefix	(String) The name prefix for each thread created by the executor. The thread name for an individual thread will be <code>namePrefix+threadNumber</code>
maxThreads	(int) The <u>max number of active threads in this pool</u> , default is <code>200</code>
minSpareThreads	(int) The <u>minimum number of threads always kept alive</u> , default is <code>25</code>
maxIdleTime	(int) The <u>number of milliseconds before an idle thread shutdown, unless the number of active threads are less or equal to minSpareThreads</u> . Default value is <code>60000(1 minute)</code>
maxQueueSize	(int) The <u>maximum number of runnable tasks that can queue up awaiting execution before we reject them</u> . Default value is <code>Integer.MAX_VALUE</code> 默认是创建一个无界队列！
prestartminSpareThreads	(boolean) Whether minSpareThreads should be started when starting the Executor or not, the default is <code>false</code>
threadRenewalDelay	(long) If a <code>ThreadLocalLeakPreventionListener</code> is configured, it will notify this executor about stopped contexts. After a context is stopped, threads in the pool are renewed. To avoid renewing all threads at the same time, this option sets a delay between renewal of any 2 threads. The value is in ms, default value is <code>1000</code> ms. If value is negative, threads are not renewed.