

# The Executor (thread pool)

## Table of Contents

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

## Introduction

表示一个线程池

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
className	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 <code>Lifecycle</code> , so that it can be started and stopped with the container. The default value for the className is <code>org.apache.catalina.core.StandardThreadExecutor</code>
name	The name used to reference this pool in other places in <code>server.xml</code> . The name is required and must be unique.

### Standard Implementation

The default implementation supports the following attributes:

Attribute	Description
threadPriority	(int) The thread priority for threads in the executor, the default is 5 (the value of the <code>Thread.NORM_PRIORITY</code> constant)
daemon	(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 <u>name prefix for each thread created by the executor</u> . The thread name for an individual thread will be <u>namePrefix+threadNumber</u>
maxThreads	(int) The <u>max number of active threads in this pool</u> , default is <u>200</u>
minSpareThreads	(int) The <u>minimum number of threads always kept alive</u> , default is <u>25</u>
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 <u>60000(1 minute)</u>
maxQueueSize	(int) The <u>maximum number of runnable tasks that can queue up awaiting execution before we reject them</u> . Default value is <u>Integer.MAX_VALUE</u> 默认是创建一个无界队列！
prestartminSpareThreads	(boolean) <u>Whether minSpareThreads should be started when starting the Executor or not</u> , the default is <u>false</u> 默认是不预先初始化最小空闲线程
threadRenewalDelay	(long) <u>If a ThreadLocalLeakPreventionListener is configured, it will notify this executor about stopped contexts</u> . After a context is stopped, threads in the pool are renewed. <u>To avoid renewing all threads at the same time, this option sets a delay between renewal of any 2 threads</u> . The value is in ms, default value is <u>1000 ms</u> . If value is negative, threads are not renewed.