**Daemon Thread**

Daemon thread in Java is a service provider thread that provides services to the user thread. Its life depends on the mercy of the user threads i.e when the entire threads die, JVM terminates this (means Daemon thread) thread automatically.

There are many Java daemon threads running automatically i.e Garbage collector, finalizer etc

Thread A

Other Daemon thread

Eg. Gargbage collector

Main Thread

JVM

Start() start()

Thread B

**Method:**

|  |
| --- |
| **1] public final void *setDaemon(boolean)* :-** marks the thread as daemon  **2] public final boolean *isDaemon()* : -** tests if the thread is a daemon thread. |

|  |
| --- |
| class Test extends Thread  {  public void run()  {  System.out.println(“Child Thread”); //Daemon thread  }  public static void main(String args[])  {  System.out.println(“Main Thread”);  Test t=new Test();  t.setDaemon(true); //create Daemon thread  t1.start();  }  } |

**Note:**

**Case 1:**

1. We have to ***create*** Daemon thread before starting the thread.
2. If we create Daemon thread after the starting the thread, it will create runtime exception i.e **IllegalThreadStateException**.

**Case 2:**

We cannot create main thread as Daemon Thread.

i.e. Thread.currentThread.setDaemon(true);

|  |
| --- |
| class Test extends Thread  {  public void run()  {  System.out.println(“Child Thread”); //Daemon thread  }  Public static void main(String args[])  {  Thread.currentThread.setDaemon(true);  System.out.println(“Main Thread”);  Test t=new Test();  t.setDaemon(true); //create Daemon thread  t1.start();  }  } |

|  |
| --- |
| class Test extends Thread  {  public void run()  {  If(Thread.currentThread().isDaemon())  System.out.println(“DaemonThread”); //Daemon thread  else  System.out.println(“User Thread”);  }  Public static void main(String args[])  {  System.out.println(“Main Thread”);  Test t1=new Test();  Test t2=new Test();  Test t3=new Test();  t1.setDaemon(true); //create Daemon thread  t1.start();  t2.start();  t3.start();  }  } |