client

# Class TCPClient.runtimeThr

java.lang.Object
    java.lang.Thread
        client.TCPClient.runtimeThr

**All Implemented Interfaces:**

`java.lang.Runnable`

**Enclosing class:**

`TCPClient`

---

```
public static class TCPClient.runtimeThr
extends java.lang.Thread
```

**Author:**

hector, adonias, gary, henna - The runtime thread will serve as the connection between the upper threads and the local and network threads. The ConcurrentLinkedQueue requestQueue and returnQueue will be stored here. These Queues will store data objects that specify the requesting thread, the requested command, and the response. The runtime thread will execute an infinite loop, checking for new data object requests and spawning either a network thread, or a local thread depending on the request type. Once spawned, the request will be considered as processing and will be de-queued from the Queue.

## *Nested Class Summary*

### Nested classes/interfaces inherited from class java.lang.Thread

`java.lang.Thread.State, java.lang.Thread.UncaughtExceptionHandler`

## *Field Summary*

**Fields**

| Modifier and Type | Field and Description |
|---|---|
| `java.util.concurrent.ConcurrentLinkedQueue<Data>` | `requestQueue` |
| `java.util.concurrent.ConcurrentLinkedQueue<Data>` | `returnQueue` |

### Fields inherited from class java.lang.Thread

MAX_PRIORITY, MIN_PRIORITY, NORM_PRIORITY

## Constructor Summary

**Constructors**

| Constructor and Description |
| --- |
| runtimeThr() |

## Method Summary

**All Methods** | **Instance Methods** | **Concrete Methods**

| Modifier and Type | Method and Description |
| --- | --- |
| void | add(Data command)<br>This function will be used by upper threads to add their data requests to the runtime thread's ConcurrentQueueList. |
| Data | getResponse()<br>This function will be used by upper threads to retrieve their requests from the runtime thread's ConcurrentQueueList responseQueue. |
| void | run() |

### Methods inherited from class java.lang.Thread

activeCount, checkAccess, clone, countStackFrames, currentThread, destroy, dumpStack, enumerate, getAllStackTraces, getContextClassLoader, getDefaultUncaughtExceptionHandler, getId, getName, getPriority, getStackTrace, getState, getThreadGroup, getUncaughtExceptionHandler, holdsLock, interrupt, interrupted, isAlive, isDaemon, isInterrupted, join, join, join, resume, setContextClassLoader, setDaemon, setDefaultUncaughtExceptionHandler, setName, setPriority, setUncaughtExceptionHandler, sleep, sleep, start, stop, stop, suspend, toString, yield

### Methods inherited from class java.lang.Object

equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

## Field Detail

### requestQueue

```
public java.util.concurrent.ConcurrentLinkedQueue<Data> requestQueue
```

#### returnQueue

```
public java.util.concurrent.ConcurrentLinkedQueue<Data> returnQueue
```

## *Constructor Detail*

#### runtimeThr

```
public runtimeThr()
```

## *Method Detail*

#### run

```
public void run()
```

**Specified by:**

```
run in interface java.lang.Runnable
```

**Overrides:**

```
run in class java.lang.Thread
```

#### add

```
public void add(Data command)
```

This function will be used by upper threads to add their data requests to the runtime thread's ConcurrentQueueList.

**Parameters:**

```
command - - The data object containing the information request and information of the requesting upper thread.
```

#### getResponse

```
public Data getResponse()
```

This function will be used by upper threads to retrieve their requests from the runtime thread's ConcurrentQueueList responseQueue.

**Parameters:**

`command - - The data object containing the updated response to the initial information request.`