

## Die Klasse Client

Objekte von Unterklassen der abstrakten Klasse `Client` ermöglichen Netzwerkverbindungen zu einem Server mittels TCP/IP-Protokoll. Nach Verbindungsaufbau können Zeichenketten (Strings) zum Server gesendet und von diesem empfangen werden, wobei der Nachrichtenempfang nebenläufig geschieht. Zur Vereinfachung finden Nachrichtenversand und -empfang zeilenweise statt, d. h., beim Senden einer Zeichenkette wird ein Zeilentrenner ergänzt und beim Empfang wird dieser entfernt. Jede empfangene Nachricht wird einer Ereignisbehandlungsmethode übergeben, die in Unterklassen implementiert werden muss. Es findet nur eine rudimentäre Fehlerbehandlung statt, so dass z.B. Verbindungsabbrüche nicht zu einem Programmabbruch führen. Eine einmal unterbrochene oder getrennte Verbindung kann nicht reaktiviert werden.

## Dokumentation der Klasse Client

### Konstruktor `Client(String pServerIP, int pServerPort)`

Es wird eine Verbindung zum durch `pServerIP` und `pServerPort` spezifizierten Server aufgebaut, so dass Daten (Zeichenketten) gesendet und empfangen werden können. Kann die Verbindung nicht hergestellt werden, kann der Client nicht zum Datenaustausch verwendet werden.

### Anfrage `boolean isConnected()`

Die Anfrage liefert den Wert `true`, wenn der Client mit dem Server aktuell verbunden ist. Ansonsten liefert sie den Wert `false`.

### Auftrag `void send(String pMessage)`

Die Nachricht `pMessage` wird – um einen Zeilentrenner ergänzt – an den Server gesendet. Schlägt der Versand fehl, geschieht nichts.

### Auftrag `void close()`

Die Verbindung zum Server wird getrennt und der Client kann nicht mehr verwendet werden. Ist Client bereits vor Aufruf der Methode in diesem Zustand, geschieht nichts.

### Auftrag `void processMessage(String pMessage)`

Diese Methode wird aufgerufen, wenn der Client die Nachricht `pMessage` vom Server empfangen hat. Der vom Server ergänzte Zeilentrenner wurde zuvor entfernt. Die Methode ist abstrakt und muss in einer Unterklasse der Klasse `Client` überschrieben werden, so dass auf den Empfang der Nachricht reagiert wird. Der Aufruf der Methode erfolgt nicht synchronisiert.