



Dokumentation der Klasse "Server"

Klassenname *Server*

Beschreibung Die abstrakte Klasse Server ermöglicht es, Clients Verbindungen über eine selbst festgelegte Portnummer anzubieten. Dabei werden die Verbindungen automatisch aufgebaut und nebenläufig abgewickelt, der Server muss lediglich auf bestimmte Ereignisse reagieren.

Bezugsklassen *ServerVerbindung*, *Liste*

Unterklassen In den Unterklassen können die Dienste *bearbeiteVerbindungsaufbau*, *bearbeiteNachricht*, *bearbeiteVerbindungsverlust* und *bearbeiteVerbindungsende* überschrieben werden.

Protokoll

Auftrag *init*(pPortNummer : GanzeZahl; pMitProtokollieren : Wahrheitswert)

nachher Der Server bietet seinen Dienst über die angegebene Portnummer an. Wenn der Parameter zum Protokollieren wahr war, wird nun jede Kommunikation über diese Verbindung in einem weiteren Fenster mitprotokolliert.

Auftrag *sendeAnAlle*(pNachricht : Zeichenkette)

nachher Die Nachricht wurde an alle Clients geschickt, mit denen der Server verbunden ist.

Auftrag *sendeAnEinen*(pClientIP, pNachricht : Zeichenkette)

vorher Eine Verbindung zu einem Client mit der angegebenen IP-Nummer besteht.

nachher Die Nachricht wurde an den Client mit der angegebenen IP-Nummer verschickt..

Auftrag *melde*(pMeldung : Zeichenkette)

nachher Im Protokollfenster wurde die angegebene Meldung angehängt..

Auftrag *beendeVerbindung*(pClientIP : Zeichenkette)

vorher Eine Verbindung zu einem Client mit der angegebenen IP-Nummer besteht.

nachher Der Server hat die Verbindung zu dem Client mit der angegebenen IP-Nummer beendet.

Anfrage *port* : GanzeZahl

nachher Diese Anfrage liefert die Portnummer des Servers.

Anfrage *clientListe* : Liste

nachher Diese Anfrage liefert die Liste mit den Verbindungen des Servers.

Anfrage *serverVerbindungVonIP*(pClientIP : Zeichenkette) : *ServerVerbindung*

nachher Diese Anfrage liefert die *ServerVerbindung* zu dem Client mit der angegebenen IP-Nummer, falls eine Verbindung besteht, sonst nichts.

Anfrage *neueServerVerbindung* : *ServerVerbindung*

nachher Diese Anfrage liefert eine neue *ServerVerbindung*, die dann mit einem Client verbunden werden kann. Diese Anfrage muss dann überschrieben werden, wenn die *ServerVerbindung* weitere Dienste oder Attribute haben soll. In diesem Fall wird als Ergebnis ein neu erzeugtes Exemplar der entsprechenden Unterklasse von *ServerVerbindung* geliefert.

Anfrage *zahlDerVerbindungen* : GanzeZahl

nachher Diese Anfrage liefert die Anzahl der Verbindungen, die der Server zurzeit betreibt.

Anfrage *mitProtokoll* : Wahrheitswert

nachher Diese Anfrage liefert "wahr", genau dann wenn alle Aktivitäten der Verbindung mitprotokolliert werden.

Auftrag *bearbeiteVerbindungsaufbau*(pClientIP : Zeichenkette)

vorher Eine Verbindung zu dem Client mit der angegebenen IP-Nummer wurde aufgebaut.

nachher (Dieser Dienst ist abstrakt und kann in den Unterklassen überschrieben werden.)
Der Server hat auf den Verbindungsaufbau z. B. mit einer Begrüßungsnachricht reagiert.

Auftrag *bearbeiteNachricht*(pClientIP, pNachricht : Zeichenkette)



vorher Der Client mit der angegebenen IP-Nummer hat dem Server die angegebene Nachricht geschickt.

nachher (Dieser Dienst ist abstrakt und kann in den Unterklassen überschrieben werden.)
Der Server hat auf die Nachricht reagiert.

Auftrag *bearbeiteVerbindungsverlust(pClientIP : Zeichenkette)*

vorher Die Verbindung zu dem Client mit der angegebenen IP-Nummer wurde verloren.

nachher (Dieser Dienst ist abstrakt und kann in den Unterklassen überschrieben werden.)
Der Server hat auf den Verbindungsverlust reagiert.

Auftrag *bearbeiteVerbindungsende(pClientIP : Zeichenkette)*

vorher Die Verbindung zu dem Client mit der angegebenen IP-Nummer wurde vom Server beendet.

nachher (Dieser Dienst ist abstrakt und kann in den Unterklassen überschrieben werden.)
Der Server hat auf das Verbindungsende reagiert.

Auftrag gibFrei

nachher Der Server mit seinen Verbindungen wurde geschlossen und steht nicht mehr zur Verfügung.

Damit es möglich ist, für Testzwecke auch mehrere Clients auf einem Rechner zu starten, muss der Client nicht wie bei den oben aufgeführten Diensten durch die IP-Nummer, sondern durch die Kombination von IP- und Port-Nummer identifiziert werden. Aus diesem Grunde werden die Dienste alternativ auch mit entsprechend erweiterter Parameterliste angeboten:

Auftrag *sendeAnEinen(pClientIP : Zeichenkette; pClientPort: GanzeZahl; pNachricht : Zeichenkette)*

vorher Eine Verbindung zu einem Client mit der angegebenen IP- und Port-Nummer besteht.

nachher Die Nachricht wurde an den Client mit der angegebenen IP- und Port-Nummer verschickt..

Auftrag *beendeVerbindung(pClientIP : Zeichenkette; pClientPort: GanzeZahl)*

vorher Eine Verbindung zu einem Client mit der angegebenen IP- und Port-Nummer besteht.

nachher Der Server hat die Verbindung zu dem Client mit der angegebenen IP- und Port-Nummer beendet.

Anfrage *serverVerbindungVonIP(pClientIP : Zeichenkette; pClientPort: GanzeZahl) : ServerVerbindung*

nachher Diese Anfrage liefert die ServerVerbindung zu dem Client mit der angegebenen IP- und Port-Nummer, falls eine Verbindung besteht, sonst nichts.

Auftrag *bearbeiteVerbindungsaufbau(pClientIP : Zeichenkette; pClientPort: GanzeZahl)*

vorher Eine Verbindung zu dem Client mit der angegebenen IP- und Port-Nummer wurde aufgebaut.

nachher (Dieser Dienst ist abstrakt und kann in den Unterklassen überschrieben werden.)
Der Server hat auf den Verbindungsaufbau z. B. mit einer Begrüßungsnachricht reagiert.

Auftrag *bearbeiteNachricht(pClientIP : Zeichenkette; pClientPort: GanzeZahl; pNachricht : Zeichenkette)*

vorher Der Client mit der angegebenen IP- und Port-Nummer hat dem Server die angegebene Nachricht geschickt.

nachher (Dieser Dienst ist abstrakt und kann in den Unterklassen überschrieben werden.)
Der Server hat auf die Nachricht reagiert.

Auftrag *bearbeiteVerbindungsverlust(pClientIP : Zeichenkette; pClientPort: GanzeZahl)*

vorher Die Verbindung zu dem Client mit der angegebenen IP- und Port-Nummer wurde verloren.

nachher (Dieser Dienst ist abstrakt und kann in den Unterklassen überschrieben werden.)
Der Server hat auf den Verbindungsverlust reagiert.

Auftrag *bearbeiteVerbindungsende(pClientIP : Zeichenkette; pClientPort: GanzeZahl)*

vorher Die Verbindung zu dem Client mit der angegebenen IP- und Port-Nummer wurde vom Server beendet.

nachher (Dieser Dienst ist abstrakt und kann in den Unterklassen überschrieben werden.)
Der Server hat auf das Verbindungsende reagiert.