

24.6.4 Request

Die Klasse `Request` (gb.web) können Sie verwenden, um HTTP-Anfragen (HTTP-Requests) an einen HTTP-Server zu schreiben. Beispiele für die praktische Realisierung finden Sie im Kapitel 24.6.8 Projekte.

24.6.4.1 Eigenschaften

Die Klasse `Request` verfügt über diese Eigenschaften:

Eigenschaft	Datentyp	Beschreibung
<code>Cookies</code>	Collection	Ermittelt eine Übersicht aller Cookies, die in einer HTTP-Anfrage (Request) enthalten sind.
<code>Debug</code>	Boolean	Gibt True zurück, wenn der Debug-Modus eingeschaltet oder setzt den Debug-Modus. Wenn der Debug-Modus eingeschaltet ist, so wird der Inhalt des Post-Request gespeichert und kann über die Eigenschaft <code>Request.Post.Contents</code> ausgelesen werden.
<code>Delimiter</code>	String	Setzt das Zeichen, das die Felder in einer GET- oder application/x-www-form-urlencoded-POST-Anfrage begrenzt oder gibt das Trenn-Zeichen zurück. Das Zeichen '&' ist das Standard-Trennzeichen.
<code>Fields</code>	String[]	Speichert eine Liste der Namen aller Feld-Variablen, die im Request enthalten sind, in einem String-Array.
<code>Files</code>	Collection	Ermittelt eine Auflistung aller Dateien, die dem Request beigelegt sind. Die Schlüssel dieser Collection sind die Datei-Namen, die im Request angegeben sind. Die Werte der Collection ist der Pfad der auf dem Server gespeicherten Dateien. Alle Dateien werden in das temporäre Verzeichnis der Anwendung hochgeladen. Sie werden zerstört, sobald die Anwendung beendet wird.
<code>InternetExplorer</code>	Integer	Gibt die Version des Internet-Explorers als HTTP-Client zurück oder NULL, wenn der Internet-Explorer nicht der HTTP-Client ist.
<code>Path</code>	String	Gibt den Request-Pfad der CGI-Applikation zurück. Ist das Äquivalent zu <code>CGI["PATH_INFO"]</code> .
<code>Query</code>	String	Gibt den Abfrage-Teil der Anfrage zurück. Ist das Äquivalent zu <code>CGI["QUERY_STRING"]</code> .
<code>Method</code>	String	Gibt die Methode "POST" oder "GET" der CGI-Applikation zurück, mit der Anfragen verschickt werden (Äquivalent zu <code>CGI["REQUEST_METHOD"]</code>).

Tabelle 24.6.4.1.1 : Eigenschaften der Klasse Request

Die Eigenschaft `Request.Get` liefert ein virtuelles Objekt `_Request_Get` zurück, um die in der GET-Anfrage gespeicherten Daten abrufen können, während die Eigenschaft `Request.Post` ein virtuelles Objekt `_Request_Post` zurück gibt, um die in der POST-Anfrage gespeicherten Daten abzurufen.

Eigenschaft	Datentyp	Beschreibung
<code>_Get.Fields</code>	String[]	Gibt ein Daten-Array zurück, das alle im GET-Request-URL enthaltenen Felder enthält.
<code>_Post.Contents</code>	String	Gibt den Request-Inhalt zurück, sofern die Eigenschaft 'Request.Debug' auf den Wert True gesetzt ist.
<code>_Post.Fields</code>	String[]	Liefert ein Daten-Array mit allen Feldern im POST-Request-Inhalt zurück.
<code>_Post.Files</code>	Collection	Gibt eine Sammlung (Collection) der Dateien zurück, die mit dem POST-Request geschickt wurden. Der Schlüssel enthält den Dateinamen und der Dateipfad ist als korrespondierender Wert in dieser Sammlung gespeichert.

Tabelle 24.6.4.1.2 : Eigenschaften der virtuellen Klassen `_Request_Get` und `_Request_Post`

Hinweis:

- Jede Datei wird in einer temporären Datei auf der Festplatte gespeichert. Wenn das `/tmp`-Verzeichnis nur eine begrenzte Kapazität hat, könnte es bei zu großen Dateien Probleme geben.

24.6.4.2 Methoden

Die Klasse Request (gb.web) besitzt nur eine Methode:

```
Static Function Exist ( Field As String ) As Boolean
```

Die Funktion gibt True zurück, wenn das angegebene Feld in der HTTP-Anfrage existiert – unabhängig von der verwendeten Methode GET oder POST.

Auch die beiden *virtuellen* Klassen *_Request_Get* und *_Request_Post* besitzen jeweils eine Methode:

```
Function _Request_Get.Exist ( Field As String ) As Boolean  
Function _Request_Post.Exist ( Field As String ) As Boolean
```

Beide Funktionen geben jeweils als Funktionswert True zurück, wenn das angegebene Feld in der GET- oder POST-Anfrage existiert.