

24.6.2.0 Klasse Webpage (gb.web)

Das Kapitel greift grundlegende Theorien aus dem Kapitel '24.6.10 Entwicklung von Web-Applikationen mit Gambas' auf und ergänzt sie durch die Beschreibung von mehreren Web-Projekten auf der Basis der Klasse Webpage.

Die Klasse Webpage implementiert u.a. eine HTML-Seite. Eine Webpage besteht stets aus zwei Teilen:

- einem HTML-Teil, der in einer *.webpage-Datei gespeichert ist und
- einem Gambas-Klassen-Teil, der in einer *.class-Datei gespeichert ist.

Die Klasse verfügt u.a. über die statische Methode Main(). Diese Methode ermöglicht es Ihnen, eine beliebige Webseite als Start-Klasse zu verwenden. Die Main()-Methode übernimmt mit der Render()-Methode die Ausgabe des kompletten HTML der generierten Webseite.

Der Einsatz der Komponente gb.web macht nur dann wirklich Sinn, wenn auf dem Internet-Webserver

- ein HTTP-Webserver wie Lighttpd oder Apache installiert und für CGI konfiguriert ist und
- mindestens die Laufzeitumgebung von Gambas installiert ist

so dass Sie dort Ihre Webseiten auf der Basis der Klasse Webpage ausführen können!

Den meisten Anwendern bleiben daher nur zwei Möglichkeiten, die entwickelten Web-Anwendungen zu testen:

1. Für den Test von Webseiten in der IDE kann der eingebettete HTTP-Server genutzt werden. Folgen Sie den Hinweisen im Kapitel '24.6.0.1 Exkurs 1' zum Einsatz des HTTP-Servers in der Konsole. Beachten Sie die spezielle Syntax beim Einfügen von CSS (Cascading Style Sheets), JavaScript und den unterschiedlichen multimedialen Objekten in eine Webpage, die sich von der HTML-Syntax unterscheidet!
2. Installation und Konfiguration eines Web-Servers im Intranet, da die Installation von Gambas auf vielen Servern von Anbietern von Webspace gegenwärtig nicht angeboten wird. Eine erprobte Beschreibung der Installation und Konfiguration des Webserver Lighttpd finden Sie im Kapitel '24.6.10.8 Installation und Einrichtung eines Lighttpd Webserver'.

Das Intranet des Autors besteht aus drei Computern (Desktop-PC A und zwei Laptops B und C). Alle Computer besitzen trotz DHCP feste IP-Adressen. Sowohl auf dem PC A als auch auf dem PC B ist jeweils ein Webserver Lighttpd installiert. Der Webserver auf PC B fungiert als Webserver für das Intranet.

Der Einsatz der Programmiersprache Gambas und der Skriptsprache JavaScript ermöglicht es Ihnen, Inhalte dynamisch in eine Webpage einzufügen oder zu ändern und nutzt dafür eine spezielle Syntax. Das Design einer Webseite können Sie mit CSS in vielfältiger Weise gestalten, die Sie hier sehen:

ID	Vorname	Nachname	Wohnort	PLZ	Straße	Geburtstag	Festnetz	Mobil	EMail
1	Arno	Adler	Arneburg	39606	Am Hafen 3	08.12.1981	03937864322	01715749482	arno.adler@arneburg.de
2	Bruno	Bär	Berlin	10404	Bode-Strasse 1	06.12.1986	03094157777	01716771528	bruno.baer@freenet.de
3	Gerda	Geier	Gera	07997	Gartenweg 23	12.09.1980	03657788989	01714472473	gerda.geier@gera.de
4	Lutz	Lama	Leipzig	04103	Lessing-Allee 5	25.02.1989	0641432222	01717346836	lutz.lama@wleipzig.de
5	Maria	Meise	München	80805	Malergasse 10	20.07.1980	0867554324	01716821096	maria.meise@mawa.com

Abbildung 24.6.2.0.1: Ausgabe eines SQLite-Datenbank-Reports in einer Webseite

Das Einfügen eines SQLite-Datenbank-Reports in die Webpage 'IncDBReport.webpage' erfolgt zum Beispiel mit einer Gambas-Funktion, die in der Gambas-Klasse 'DBReport.class' definiert wird.

Über ein JavaScript-Skript wird ein Timer realisiert, mit dem die aktuelle Zeit in der Fußzeile aller Webseiten periodisch angezeigt wird:

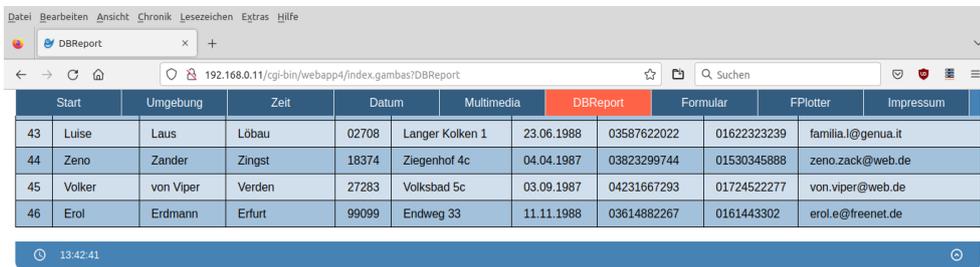


Abbildung 24.6.2.0.2: Anzeige der aktuellen Zeit (Footer) mit einem JavaScript-Timer

Die Webseiten werden vom Webserver auf dem Computer B ausgeliefert und auf dem Computer A im Webbrowser angezeigt.

24.6.2.0.1 Eigenschaften

Die Klasse Webpage besitzt nur zwei Eigenschaften:

- **Webpage.Buffered:** Gibt True zurück, wenn das Webpage-Rendering gepuffert wird oder setzt den Puffer-Modus. Der Standardwert ist False. Wenn Webpage.Buffered False ist, wird die Standardausgabe von Gambas nicht gepuffert. Das bedeutet, dass die generierte Antwort direkt an die Standardausgabe – in der Regel eine HTTP-Server-Pipe – gesendet wird. Wenn Webpage.Buffered dagegen True ist, wird die gesamte Antwort auf die Anfrage gepuffert und in einem Zug an den HTTP-Server übergeben und gesendet. Mit anderen Worten: Wenn Webpage.Buffered True ist, können Sie sicher sein, dass der Client-Browser zum Beispiel die resultierende HTML-Seite so spät wie möglich, aber in einem Zug erhält. Andernfalls könnte der Client-Browser eine Teilantwort auf seine Anfrage erhalten, während das CGI-Skript diese generiert. Eine ungepufferte HTML-Seite kommt freilich schneller an, aber der Browser kann nur einen Teil davon anzeigen, während er auf den Abschluss der Anfrage wartet.
- **Webpage.Parent:** Gibt die übergeordnete Webseite an, wenn Webseiten verschachtelt werden.

24.6.2.0.2 Methoden

Die Klasse Webpage verfügt neben der o.a. statischen Methode Main() über zwei weitere Methoden:

- **Webpage.ToString():** Gibt den Inhalt der Webseite als HTML-String zurück.
- **Webpage.Render():** Gibt den Inhalt der Webseite als HTML-String auf die Standard-Ausgabe aus.

Der Gambas-Compiler generiert mit der Render()-Methode HTML-Quelltext aus einer Webpage-Datei wp_name.webpage und aus der mit ihr verknüpften Klasse wp_name.class.

24.6.2.0.3 Webpage – Syntax

Die Klasse Webpage verwendet eine spezielle Syntax – die an ASP (Active Server Pages) angelehnt ist – um Gambas-Quelltext in eine Webpage *.webpage einzufügen.

Syntax	Beschreibung
<%Quelltext%>	Jeder Gambas-Quelltext wird 'so wie er ist' ausgeführt.
<%=Expression%>	Zuweisung eines Gambas-Ausdrucks, der mit der Html\$-Funktion konvertiert wird.
<!--Comment-->	Ein Webpage-Kommentar wird vollständig ignoriert, während ein HTML-Kommentar wie '<!- HTML-COMMENT ->' im HTML ausgegeben wird.
<%/>	Eine Abkürzung für <%=Application.Root%>.

Syntax	Beschreibung
<<Webpage>>	Fügt den Inhalt einer anderen Webseite in die aktuelle Webseite ein.
<<Webpage arg1="value1" arg2="value2" ... >>	Fügt den Inhalt einer anderen Webseite mit (optionalen) Attributen in die aktuelle Webseite ein. Beispiel: <<AstroWebpage name="sterne" anzahl="7">>
<%!arg_name%>	Liefert den Wert des Attributs 'arg_name.'
<<--CONTENTS-->>	Kennzeichnet die Grenze zwischen der Kopf- und der Fußzeile innerhalb der eingebundenen Webpage.
<</Webpage>>	Einfügen des Footers einer eingefügten Webseite. Die Angabe ist optional, wenn der Footer leer ist.

Tabelle 24.6.2.0.1 : Webpage-Syntax