

Bindungen und JavaScript

Einleitung



Mit dem Beispiel, das über die Schaltfläche Bindungen aufgerufen wird, soll gezeigt werden, wie man programmtechnisch in der Bedienoberfläche der Anwendung auf die nicht sichtbaren Bindungen zugreifen kann: mit Hilfe der Programmiersprache JavaScript ist es möglich, programmtechnisch auf die Bindungen einer Seite zuzugreifen und „Verarbeitungen“ (Umformungen, Berechnungen, andere Darstellungen, usw.) vorzunehmen.

In diesem Beispiel geht es zunächst darum, den JavaScript-Zugriff in Abhängigkeit vom jeweiligen (TD Mobile) Datentyp zu erläutern.

Grundsätzlich gibt es folgende Standard-Datentypen:

- Boolean
- String
- Number und
- DateTime

Hinzu kommen noch komplexe Datentypen, die zunächst als Klassen angelegt und ebenfalls als Bindung (auf einer Seite) angelegt werden können

Datentypen als Bindung

	einfach	komplex
Einzeln	x (Boolean, String, Number, Date/Time)	x (mit Boolean, String, Number, Date/Time)
Array	x (String)	x (mit Boolean, String, Number, Date/Time)

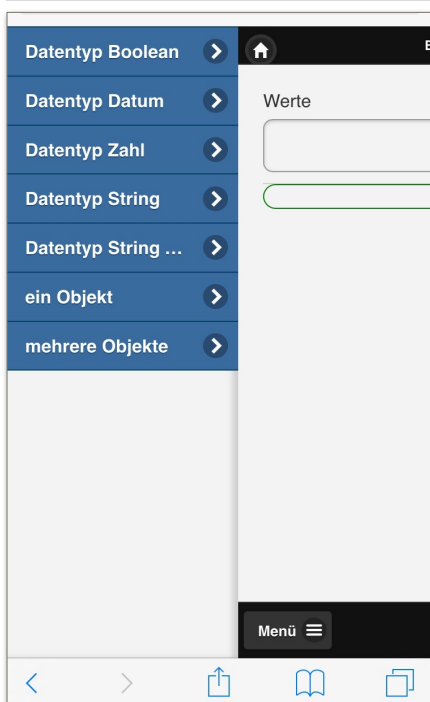
Einfache und komplexe Datentypen können einzeln oder als Array angelegt werden.

Die Tabelle zeigt die in den einzelnen Aufrufen bearbeiteten Datentypen der Bindungen. In der Anwendung werden jeweils die Werte der Datentypen ermittelt und in die Bindung **ERGEBNIS** geschrieben. Dem Datenfeld Werte wurde als Wertebindung **ERGEBNIS** zugewiesen. So wird das über das jeweilige JavaScript ermittelte Ergebnis in dem Datenfeld angezeigt.

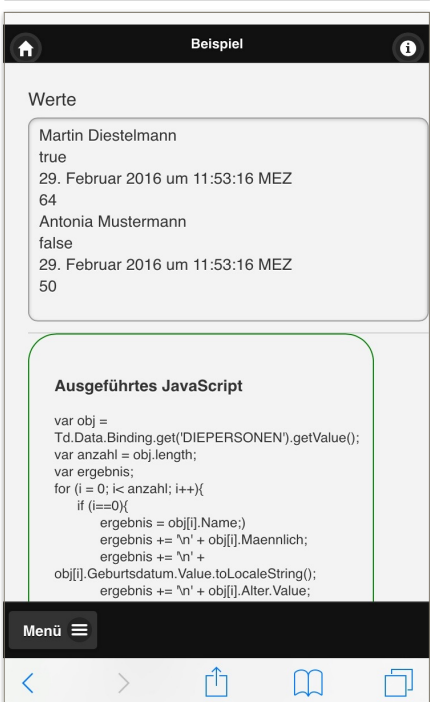
Die Beispielanwendung zeigt die „Behandlung“ von

- einfachen Datentypen Boolean, Datum, Number und String,
- einem einfachen String-Array,
- einem komplexen Datentyp und
- einem Array eines komplexen Datentyps.

Anfangszustand der Anwendung – links ist ein Menü aufgeklappt, in dem die einzelnen JavaScripte aufgerufen werden können.



Die Abbildung zeigt das Ergebnis der Auswertung „mehrere Objekte“. Im unteren Bereich wird der dazu gehörende JavaScript-Code dargestellt.



In der Kopfzeile links befindet sich die Schaltfläche „Home“, mit der man auf die Startseite gelangt.

Nach Anklicken der Info-Schaltfläche wird ein Dialog angezeigt, auf der die Funktionalität der Seite erläutert wird.

In der Fußzeile befindet sich eine Menü-Schaltfläche, mit der das seitliche Menü an- bzw. abgeschaltet werden kann.

Codierung und Erläuterung

Bei der Erläuterung gehen wir davon aus, dass der Leser Erfahrung mit dem Entwicklungswerkzeug SQLWindows (im Produkt Team Developer), aber keine oder nur geringe Kenntnisse der Entwicklungsumgebung TD Mobile, für die Erstellung von mobilen Geschäftsanwendungen, hat. Aus dieser Sichtweise erfolgen die Erläuterungen, die am Ende dazu führen sollen, dass der Leser auch andere JavaScript-Beispiele für seine Bedürfnisse innerhalb des Produkts TD Mobile anpassen kann.

einfach – Boolean, String

Nach Antippen des Menüpunkts **Datentyp Boolean**, aber auch beim Aufruf der Menüfunktion **Datentyp String** wird die JavaScript-Funktion `einfachenDatentypAnzeigen` mit der booleschen Bindung als Parameter mit dem Namen `wert` übergeben.

```
Funktion: einfachenDatentypAnzeigen
Parameter: wert
Aktionen
    Td.Data.Binding.get('ERGEBNIS').setValue(wert);
```

Die JavaScript-Funktion bedient sich der API von TD Mobile. Aus dem Namensraum `Td.Data.Binding` wird die Funktion `get()` aufgerufen und als Parameter die Bindung `ERGEBNIS` übergeben. Der Rückgabewert, die Bindung `ERGEBNIS`, wird danach mit dem Wert des Übergabeparameters gesetzt: Jetzt steht entweder `true` oder `false` im Datenfeld, da die Bindung `ERGEBNIS` als Wertebindung

einfach – Datum, Zahl

Die beiden Datentypen `Number` und `Date` müssen geringfügig anders behandelt werden, da in diesen beiden Fällen der Wert selbst im Attribut `value` hinterlegt ist.

```
Funktion: einfachenDatentypAnzeigenValue
Parameter: wert
Aktionen
    Td.Data.Binding.get('ERGEBNIS').setValue(wert.Value);
```

Zusammenfassung – einfache Datentypen

Bindungen einer Seite können entweder als Parameter der jeweiligen JavaScript-Funktion übergeben oder innerhalb der JavaScript-Funktion über die TD Mobile API im Namensraum `Td.Data.Binding` ermittelt werden.

Die Wertermittlung selbst unterscheidet sich in Abhängigkeit des jeweiligen, einfachen Datentyps: Bei `Boolean` und `String` steht der Wert direkt in der Variablen, während bei `Number` und `Date/Time` der Wert über das Attribut `Value` ermittelt wird.

String – Array

Array-Bindungen werden in TD Mobile mit der Kennzeichnung `[*]` versehen: in diesem Fall handelt es sich um einen dynamischen Array. Im der nachfolgenden JavaScript-Funktion `ArrayAuslesen` werden alle Werte einer Array-Bindung ausgelesen und das Ergebnis in die Bindung `ERGEBNIS` geschrieben.

```
Funktion: ArrayAuslesen
Aktionen
    var elemente = Td.Data.Binding.get('AUSWAHL').getValue();
```

```

var anzahl = elemente.length;
var ergebnis;
Td.Data.Binding.get('ERGEBNIS').setValue(ergebnis);
for (i=0; i< anzahl; i++){
  if (i==0){
    ergebnis = elemente[i];}
  else {
    ergebnis += '\n' + elemente[i];}
};
Td.Data.Binding.get('ERGEBNIS').setValue(ergebnis);

```

In diesem Fall wird die (Array-) Bindung nicht als Parameter übergeben. Stattdessen wird die Bindung über die TD Mobile API der Variablen `ergebnis` zugewiesen.

Mit der der Zeile `var anzahl = elemente.length` wird die Anzahl der im Array enthaltenen Element ermittelt. Als nächstes durchläuft die Funktion eine `for`-Schleife, in der der jeweilige (String-) Wert der Variablen `ergebnis` zugewiesen wird.

Als letztes schließlich wird die Variable `ergebnis` – die Variable, in der alle Werte aus dem Array stehen – mit Hilfe der TD Mobile API in die Bindung `ERGEBNIS` geschrieben.

komplex - (alle Datentypen)

Als nächstes soll ein komplexer Datentyp behandelt werden. Komplexe Datentypen werden als Klassen definiert und dann auf den jeweiligen Seiten als Bindungen instanziiert. Die nachfolgende JavaScript-Funktion `ObjektAuslesen` zeigt, wie man Werte der unterschiedlichen Datentypen in einem komplexen Objekt ermitteln kann.

```

Funktion: ObjektAuslesen
Aktionen
var obj = Td.Data.Binding.get('OBJEKT').getValue();
var test = '';
Td.Data.Binding.get('ERGEBNIS').setValue(test);
test = obj.Wahr;
test += '\n' + obj.Datum.Value.toLocaleString();
test += '\n' + obj.Zahl.Value;
test += '\n' + obj.String;
Td.Data.Binding.get('ERGEBNIS').setValue(test);

```

In diesem Fall wird über die TD Mobile API das Objekt ermittelt und der Variablen `obj` zugewiesen. Die Variable `test` wird angelegt und in den nachfolgenden Zeilen werden die Werte der Bestandteile des Objekts der Variablen `test` zugewiesen. Die Variable `test` wird am Ende (wieder über die TD Mobile API) der Bindung `ERGEBNIS` zugewiesen.

komplex – Array

Als letztes steht noch die Dokumentation der Wertermittlung von Arrays von komplexen Datentypen an. Dieses wird in der nachfolgenden JavaScript-Funktion `ObjekteAuslesen` gezeigt.

```

Funktion: ObjekteAuslesen
Aktionen
var obj = Td.Data.Binding.get('DIEPERSONEN').getValue();
var anzahl = obj.length;
for (i = 0; i< anzahl; i++){
  if (i==0){
    var ergebnis = obj[i].Name;
    ergebnis += '\n' + obj[i].Maennlich;

```

```
    ergebnis += '\n' + obj[i].Geburtsdatum.Value.toLocaleString();
    ergebnis += '\n' + obj[i].Alter.Value;
  }
  else {
    ergebnis += '\n' + obj[i].Name;
    ergebnis += '\n' + obj[i].Maennlich;
    ergebnis += '\n' + obj[i].Geburtsdatum.Value.toLocaleString();
    ergebnis += '\n' + obj[i].Alter.Value;
  }
};
Td.Data.Binding.get('ERGEBNIS').setValue(ergebnis);
```

In diesem Fall wird die (komplexe Array-)Bindung mithilfe der TD Mobile-API der Variable `obj` zugewiesen und in der nächsten Zeile wird die Anzahl der im Array enthaltenen Elemente ermittelt.

In der `for`-Schleife – unterschieden nach dem ersten und weiteren Werten – werden die Werte in Abhängigkeit von ihrem jeweiligen Datentyp ermittelt und der Variablen `ergebnis` zugewiesen.

Am Ende schließlich wird die Variable `ergebnis` in die Bindung `ERGEBNIS` eingetragen und daher das Ergebnis im Datenfeld der Bedienoberfläche angezeigt.

Zusammenfassung

In diesem Papier ging es um die Klärung der Frage, wie man im Clientteil einer TD Mobile-Anwendung auf die Bindungen der jeweiligen Seite zugreifen kann. Unterschieden wurde dabei, ob es sich um einfache oder komplette Datentypen handelt, die entweder einzeln oder als Array definiert sind.

Es konnte gezeigt werden, dass alle möglichen Bindungen durch eigene JavaScript-Funktionen angesprochen und ausgewertet werden können.

Die in diesem Papier beschriebenen Vorgehensweisen bilden die Grundlage für weitere Beispiele, in denen es beispielsweise darum gehen wird, eine Oberfläche anzulegen, die nicht aus den Standard-Controls von TD Mobile definiert, sondern durch eine entsprechende JavaScript-Funktion gestaltet wird. Auch die erweiterte Integration der Google Maps API kann durch eigene JavaScript-Funktionen vorgenommen werden.

MD Consulting & Informationsdienste GmbH

Michaelisstraße 13a
99084 Erfurt

Berghamer Straße 14
85435 Erding

phone (+49) 8122 97400
email info@md-consulting.de