



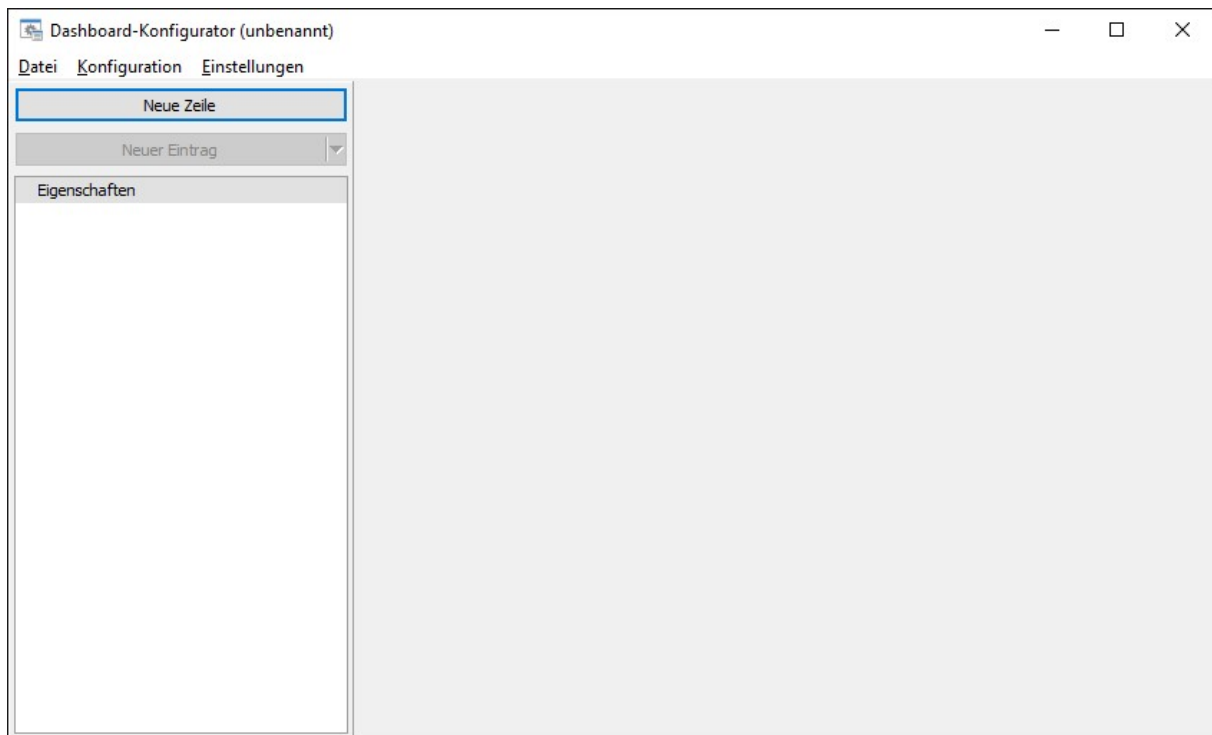
AFS Dashboard-Konfigurator - Schnelleinführung

Inhaltsverzeichnis

Das Hauptfenster.....	3
Das Hauptmenü.....	3
Menü „Datei“	3
Menü „Konfiguration“	3
Menü „Einstellungen“	3
Anlegen einer Konfiguration	4
Verschieben von Einträgen	6
Konfigurieren von Einträgen	7
Spezielle Eigenschaftseditoren.....	8
Einfügen von Einträgen aus Templates	10
Speichern eines Eintrags als Template	11
Ändern einer Konfiguration.....	11
Zeilen hinzufügen	11
Zeilen verschieben.....	11
Ändern der Art eines Eintrags	11
Löschen eines Eintrags	11
Löschen einer Zeile.....	11
Vorschau.....	11
Aufbau einer Konfigurationsdatei	12
Knoten „Dashboard“	12
Knoten „Row“	12
Knoten „Item“	12
Übersicht über die Eintragstypen.....	13
Säulendiagramm (einfach)	13
Säulendiagramm (mehrere Werte)	14
Bereichsdiagramm.....	15
Liniendiagramm.....	16
Tortendiagramm.....	17

Tabelle	18
Kalender	19
Externe Inhalte	20
Benutzerdefiniert	21
Zusammenfassung.....	22

Das Hauptfenster



Das Hauptfenster besteht aus dem Hauptmenü, dem Einstellungsbereich und dem Übersichtsbereich.

Das Hauptmenü

Menü „Datei“

- Neu – Erstellt eine neue Konfiguration
- Vorschau – Zeigt die aktuelle Konfiguration im Webbrowser an
- Beenden – Beendet das Programm

Menü „Konfiguration“

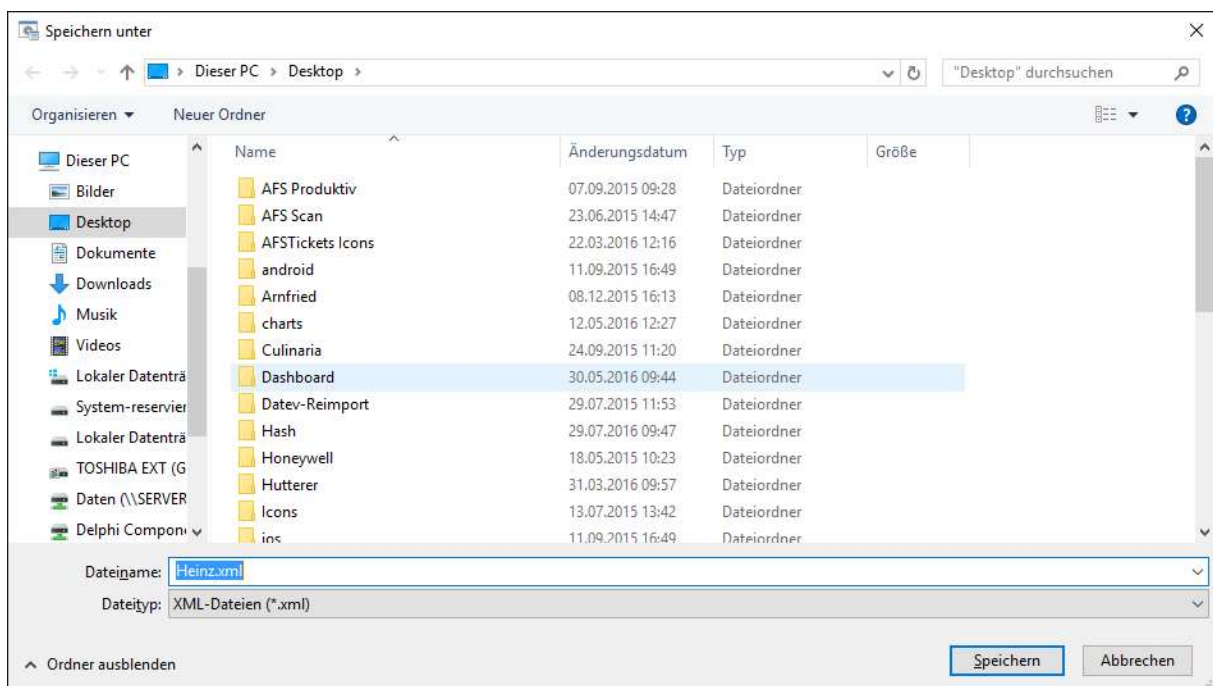
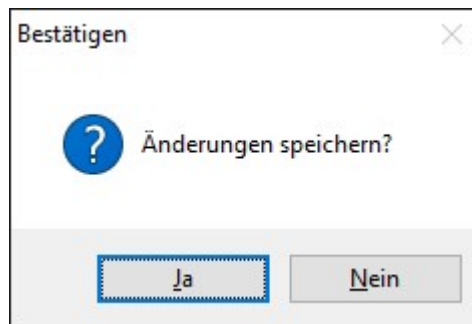
- Laden – Lädt eine bestehende Konfigurationsdatei
- Speichern – Speichert die aktuelle Konfiguration unter einem anzugebenden Namen ab

Menü „Einstellungen“

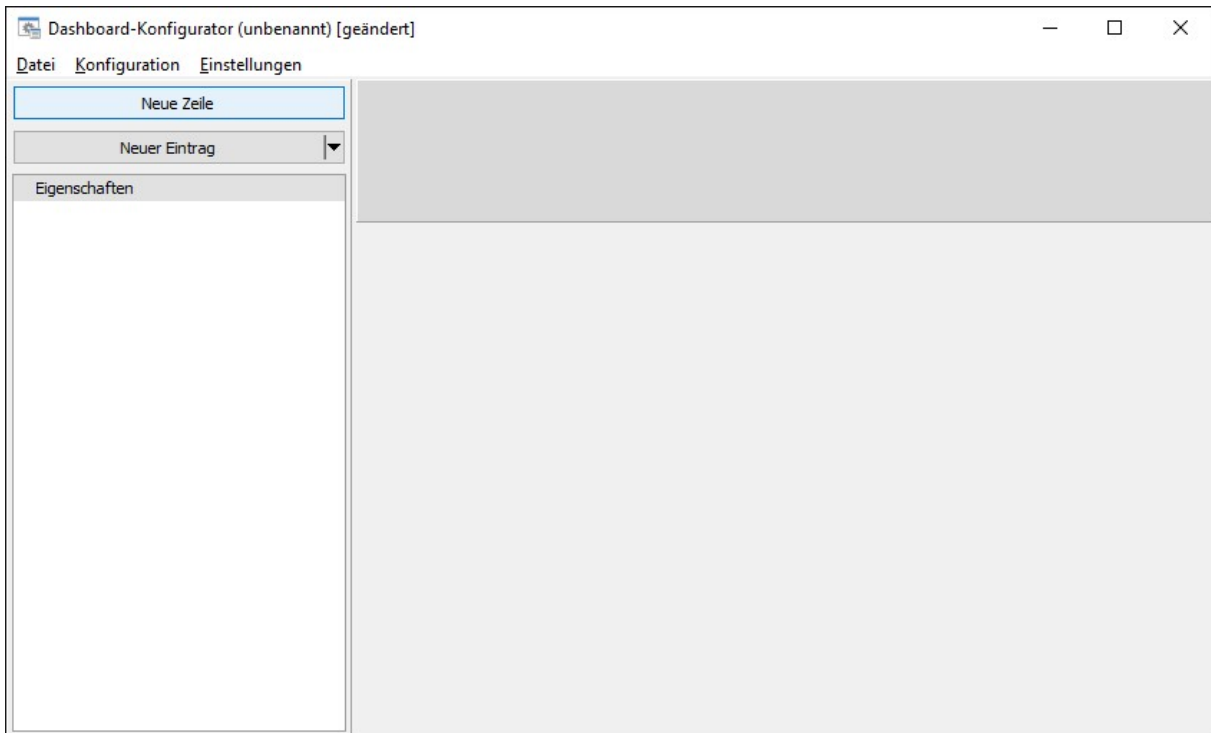
- Einstellungen (Vorschau) – Zeigt einen Dialog an, in dem sich die Pfade zu den benötigten Dateien anpassen lassen

Anlegen einer Konfiguration

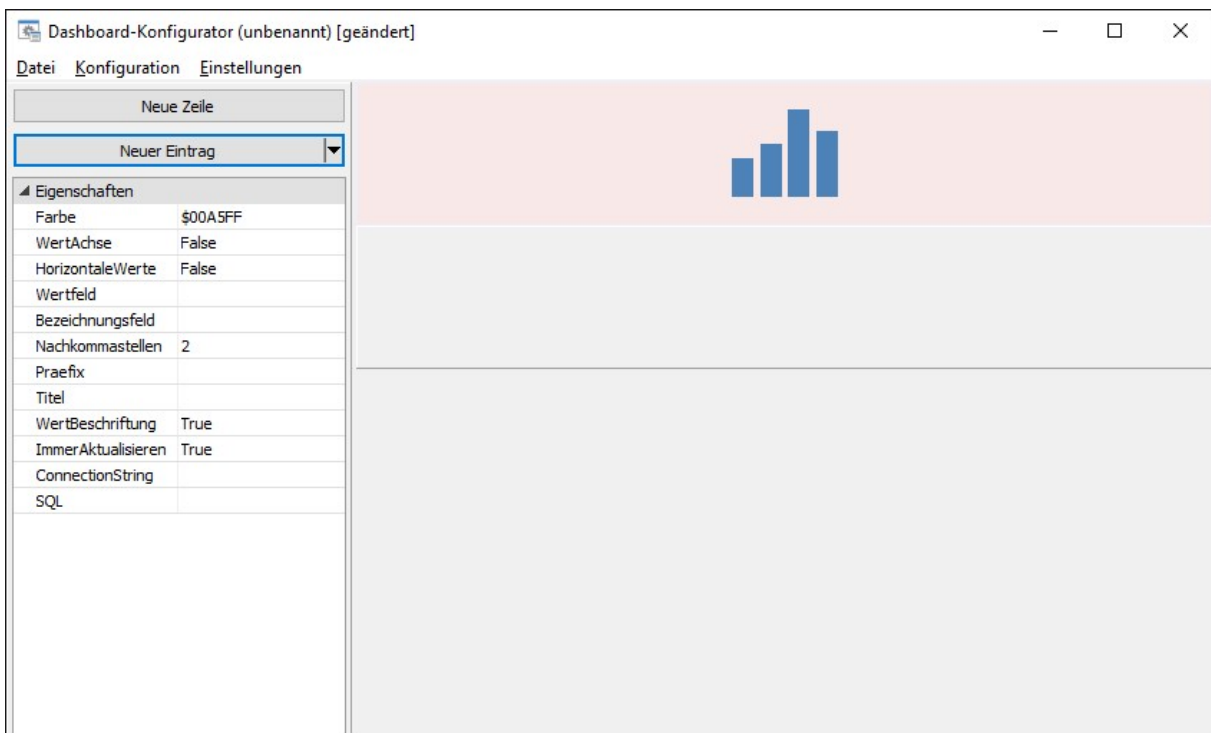
Im Hauptmenü „Datei – Neu“ wählen. Sollte bereits eine geänderte Konfiguration geladen gewesen sein, kann man diese zunächst abspeichern.



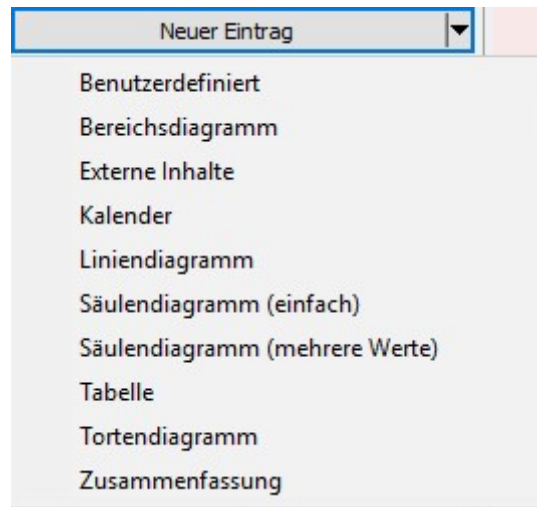
Im Anschluss sind sowohl Einstellungs- als auch Übersichtsbereich leer. Als erster Schritt ist nun mit der entsprechenden Schaltfläche eine neue Zeile anzulegen.



Sobald mindestens eine Zeile vorhanden und ausgewählt ist, ist auch die Schaltfläche „Neuer Eintrag“ aktiviert. Man kann nun wahlweise zunächst so viele Zeilen anlegen, wie man benötigt, oder erst Zeile für Zeile konfigurieren und bei Bedarf neue Zeilen hinzufügen. Man kann zur Laufzeit beliebig zwischen den vorhandenen Zeilen wechseln, indem man sie mit der linken Maustaste anklickt. Welche Zeile gerade aktiv ist, erkennt man an der dunkleren Färbung (dunkelgrau bei leeren Zeilen, eine Art helles Pink beim aktiven Eintrag).



Um der aktiven Zeile einen Eintrag hinzuzufügen, wählt man die Schaltfläche „Neuer Eintrag“. Daraufhin öffnet sich eine Auswahlliste mit den zur Verfügung stehenden Eintragsarten.



Theoretisch kann man einer Zeile beliebig viele Einträge hinzufügen, in der Praxis sollte man sich aber auf maximal 4 (besser 3) Einträge beschränken, da diese später im Dashboard in der Größe skaliert werden und somit die Lesbarkeit leiden würde.

Verschieben von Einträgen

Einträge lassen sich bequem per Drag&Drop verschieben, auch zeilenübergreifend. Dabei verhält sich das Programm folgendermaßen:

- Drop auf eine leere Zeile verschiebt den Eintrag dorthin
- Drop auf einen vorhandenen Eintrag
 - o In derselben Zeile tauschen beide Einträge ihre Position
 - o Zeilenübergreifend wird der neue Eintrag **vor** den vorhandenen platziert
 - Daraus folgt: um einen Eintrag zeilenübergreifend **hinter** einen vorhandenen zu bekommen, muss man ihn **2 Mal** verschieben.

Konfigurieren von Einträgen

Um einen Eintrag zu konfigurieren, muss dieser aktiviert (d.h. angeklickt) werden. Daraufhin werden im Einstellungsbereich dessen Eigenschaften mit deren aktuellen Werten aufgelistet.

Neue Zeile

Neuer Eintrag ▼

▲ Eigenschaften

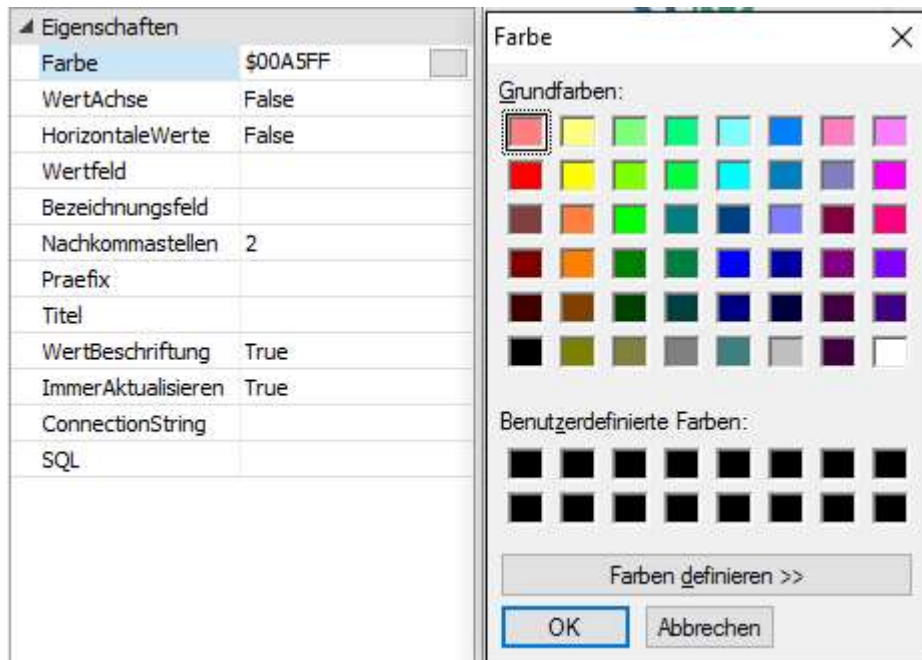
Farbe	\$00A5FF
WertAchse	False
HorizontaleWerte	False
Wertfeld	
Bezeichnungsfeld	
Nachkommastellen	2
Praefix	
Titel	
WertBeschriftung	True
ImmerAktualisieren	True
ConnectionString	
SQL	

Für die Bedeutung der einzelnen Eigenschaften je Eintragstyp bitte im zugehörigen Zusatzdokument nachschauen, das würde hier den Rahmen sprengen.

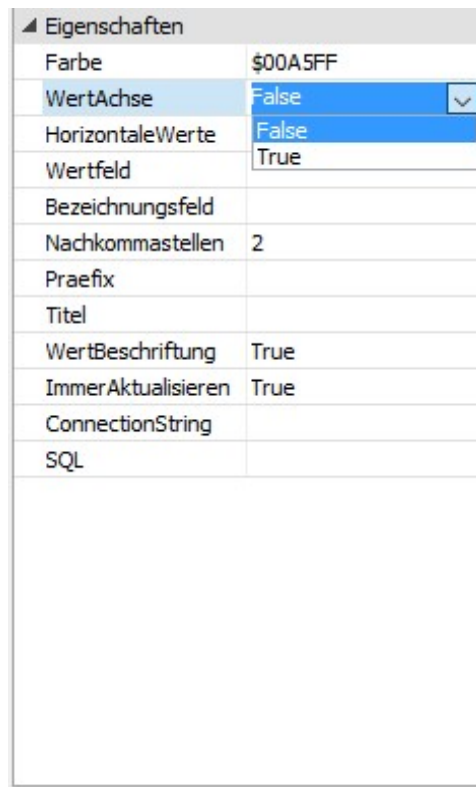
Spezielle Eigenschaftseditoren

Gewisse Eigenschaftstypen verfügen über spezielle Editoren:

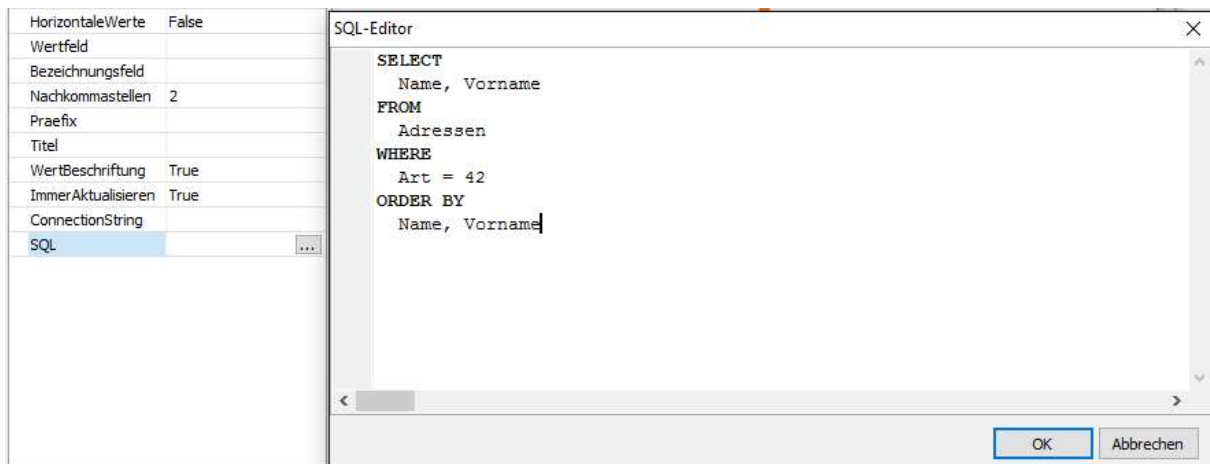
Farben



Wahrheitswerte



Mehrzeile Textwerte (z.B. SQL)



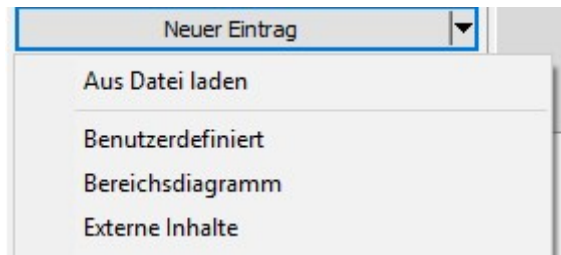
Alle anderen sind reine einzeilige Texteingaben, auch Zahlen (hier werden ungültige Angaben einfach ignoriert).

Wichtiger Hinweis:

Alle Änderungen werden erst dann wirklich übernommen, wenn man im Eigenschaftseditor/Objektinspektor entweder die Enter-Taste drückt oder auf eine andere Eigenschaft klickt.

Einfügen von Einträgen aus Templates

Neu hinzugekommen ist die Möglichkeit, einen Eintrag aus einer vorkonfigurierten Textdatei zu erstellen (Baukastenprinzip). Dazu wählt man aus der Auswahlliste „Neuer Eintrag“ den Punkt „Aus Datei laden“.



Daraufhin öffnet sich ein Dateidialog, mit dem man das einzufügende Template auswählen kann. Als Dateierweiterung ist *.txt voreingestellt, weitere Möglichkeiten sind *.xml oder alle Dateitypen (*.*). Die einzufügende Datei muss inhaltlich gültiges XML aufweisen, unabhängig von der Dateierweiterung. Des Weiteren muss dieses XML der Struktur entsprechen, die auch das Dashboard verwendet, z.B. muss der erste Knoten zwingend „Item“ heißen, ein Attribut namens „Typ“ aufweisen und dieser Typ einem bekannten Typbezeichner entsprechen. Code-Beispiel für ein gültiges Template:

```
<Item type="Pie">
  <Legende>True</Legende>
  <InlineLegende>Ohne</InlineLegende>
  <Typ>Torte</Typ>
  <Wertfeld>VK1</Wertfeld>
  <Bezeichnungsfeld>Bezeichnung</Bezeichnungsfeld>
  <Nachkommastellen>2</Nachkommastellen>
  <Titel>Teuerste Artikel</Titel>
  <WertBeschriftung>True</WertBeschriftung>
  <ImmerAktualisieren>True</ImmerAktualisieren>
  <SQL>SELECT
    VK1, Bezeichnung
  FROM
    Artikel
  ORDER BY
    VK1 DESC
  LIMIT 20</SQL>
</Item>
```

Im Dashstart-Verzeichnis des Managers gibt es übrigens einen Ordner „ITEMS“, der bereits einige vorkonfigurierte Templates enthält.

Speichern eines Eintrags als Template

Im umgekehrten Fall ist es auch möglich, einen Eintrag als Template abzuspeichern. Dazu klickt man mit der rechten Maustaste auf den entsprechenden Eintrag und wählt im Kontextmenü „Als Template speichern“. Im daraufhin erscheinenden Dateialog gibt man nur noch Verzeichnis und Dateinamen (Vorgabeendung *.txt) an, unter dem das Template abgelegt werden soll.

Ändern einer Konfiguration

Zeilen hinzufügen

Ein Klick auf „Neue Zeile“ legt eine solche an und platziert sie an unterster Stelle.

Zeilen verschieben

Dies ist im Konfigurator leider nicht direkt möglich, es gibt aber 2 Möglichkeiten, dieses Ziel zu erreichen:

- Anlegen einer neuen Zeile und Verschieben der einzelnen Einträge per Drag&Drop
- Direkte Manipulation der XML-Datei in einem Texteditor wie z.B. Notepad.

Ändern der Art eines Eintrags

Um die Eintragsart eines bestimmten Eintrags zu ändern, klickt man mit der rechten Maustaste auf denselben. Im Kontextmenü wählt man dann den obersten Punkt „Eintragsart ändern“. Das Untermenü enthält dann eine Liste der verfügbaren Eintragsarten exklusive des aktuell zugewiesenen. Nach Zuweisung einer abweichenden Eintragsart werden die kompatiblen Eigenschaften (also diejenigen, die in beiden Arten vorhanden sind und denselben Datentyp haben) automatisch übernommen, so dass man nicht komplett von vorn beginnen muss.

Löschen eines Eintrags

Hierzu genügt ein Rechtsklick auf den entsprechenden Eintrag und dann der Menüpunkt „Eintrag löschen“. Achtung: es erfolgt keine Sicherheitsabfrage!

Löschen einer Zeile

Rechtsklick auf die betreffende Zeile (bzw. einen der darin enthaltenen Einträge) und im Kontextmenü „Zeile löschen“ wählen. Auch hier erfolgt keine Sicherheitsabfrage.

Vorschau

Im Programm ist ein Webbrowser eingebaut, den man zur Vorschau nutzen kann. Dieser muss allerdings zunächst einmalig konfiguriert werden. Dazu wählt man im Hauptmenü den Punkt „Einstellungen – Einstellungen (Vorschau)“. Hier sind einzustellen:

- Template-Pfad – das Verzeichnis, das die Javascript-Templates enthält (meist ein Unterverzeichnis namens „js/charts/Templates“)
- Ausgabepfad – das Verzeichnis, das die fertig befüllten Javascript-Dateien aufnimmt (im Normalfall das übergeordnete Verzeichnis des Template-Verzeichnisses)
- ConnectionString – der Standard-Verbindungsstring zur abzufragenden Datenbank

Zur Anzeige der Vorschau wählt man dann den Menüpunkt „Datei – Vorschau“ bzw. die F12-Taste.

Einschränkung: wenn ein datenbankgebundener Eintrag kein gültiges SQL enthält, sondern lediglich einen Platzhalter („Makro für Fußgänger“), funktioniert die Vorschau nicht, da die Übersetzung des Platzhalters im Manager geschieht.

Aufbau einer Konfigurationsdatei

Die Konfigurationsdateien werden im XML-Format geschrieben und gelesen. Der Aufbau ist immer folgender:

Knoten „Dashboard“

Dies ist der Wurzelknoten. Er ist einmalig vorhanden und umschließt alle anderen Einträge, d.h. in der ersten Zeile wird er geöffnet („<Dashboard>“) und in der letzten wieder geschlossen („</Dashboard>“).

Knoten „Row“

Dies ist ein Knoten für eine Zeile. Es können beliebig viele Zeilen enthalten sein, die selbst wiederum beliebig viele Einträge enthalten können.

Knoten „Item“

Dies ist ein Knoten für einen Eintrag. Er muss über ein Attribut „type“ verfügen, welches den Eintragsstyp festlegt. Abhängig von diesem Attribut enthält der „Item“-Knoten unterschiedliche Unterknoten, deren Namen mit den Eigenschaftsnamen des entsprechenden Typs korrespondieren.

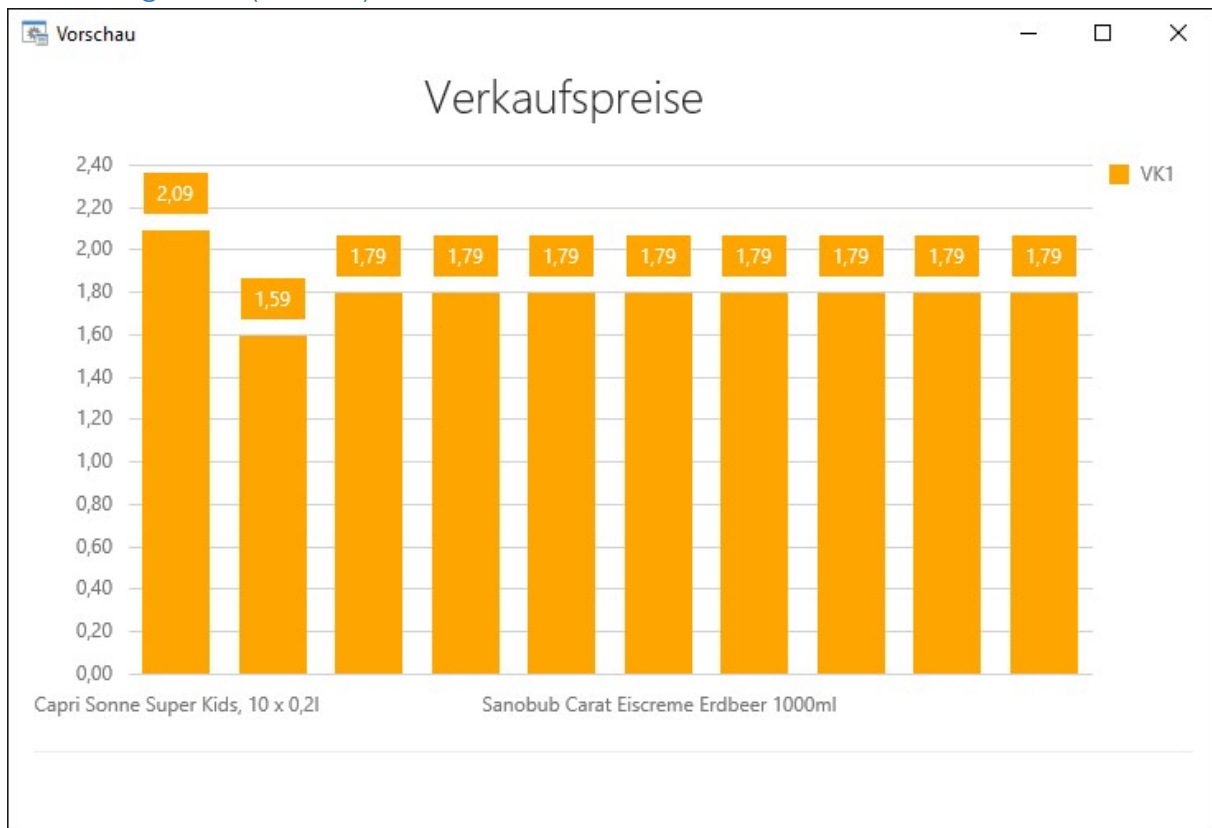
Da es sich um XML handelt, müssen alle Knoten wieder geschlossen werden, sei es in der langen („<Knoten>Wert</Knoten>“) oder der kurzen („<Knoten />“) Schreibweise. Letztere sollte theoretisch nicht vorkommen, da der Konfigurator Standardwerte nicht abspeichert und somit keine leeren Knoten entstehen dürften. Es schadet aber nicht, wenn leere Knoten enthalten sein sollten, solange sie syntaktisch korrekt sind.

Beispiel:

```
<!-- Oberster Knoten -->
<Dashboard>
  <!-- Zeile -->
  <Row>
    <!-- Eintrag -->
    <Item type="ExternalContent">
      <!-- Eigenschaft -->
      <Timeout>0</Timeout>
      <!-- Weitere Eigenschaft -->
      <URL>http://www.afs-software.de/news.html</URL>
    <!-- Ende des Eintrags -->
    </Item>
    <!-- Weiterer Eintrag -->
    <Item type="Custom">
      <!-- Eigenschaft -->
      <Scriptname>Mitarbeiter.txt</Scriptname>
    <!-- Ende des weiteren Eintrags -->
    </Item>
  <!-- Ende der Zeile -->
</Row>
<!-- Ende des obersten Knotens -->
</Dashboard>
```

Übersicht über die Eintragstypen

Säulendiagramm (einfach)



Eigenschaft	Typ	Bedeutung
Farbe	Ganze Zahl	Farbe der Säulen
WertAchse	True/False	Skala (linke Seite) anzeigen
HorizontaleWerte	True/False	Links/rechts oder unten/oben
Wertfeld	String	Name des numerischen Feldes mit den darzustellenden Werten
Bezeichnungsfeld	String	Name des Feldes, das die Bezeichner zu den Werten enthält
Nachkommastellen	Ganze Zahl	Selbsterklärend
Praefix	String	Kann zur Identifikation des Datensatzes verwendet werden
Titel	String	Überschrift („Verkaufspreise“ in der Abbildung)
WertBeschriftung	True/False	Werte am Säulenende anzeigen
ImmerAktualisieren	True/False	Sollen Werte bei jedem Aufruf neu vom Server abgefragt werden?
ConnectionString	String	Wenn belegt, wird unabhängig von der globalen eine eigene Verbindung aufgebaut, z.B. zu einer anderen Datenbank
SQL	String	Enthält die auszuführende Abfrage (Wertfeld und Bezeichnungsfeld müssen natürlich enthalten sein)

Säulendiagramm (mehrere Werte)



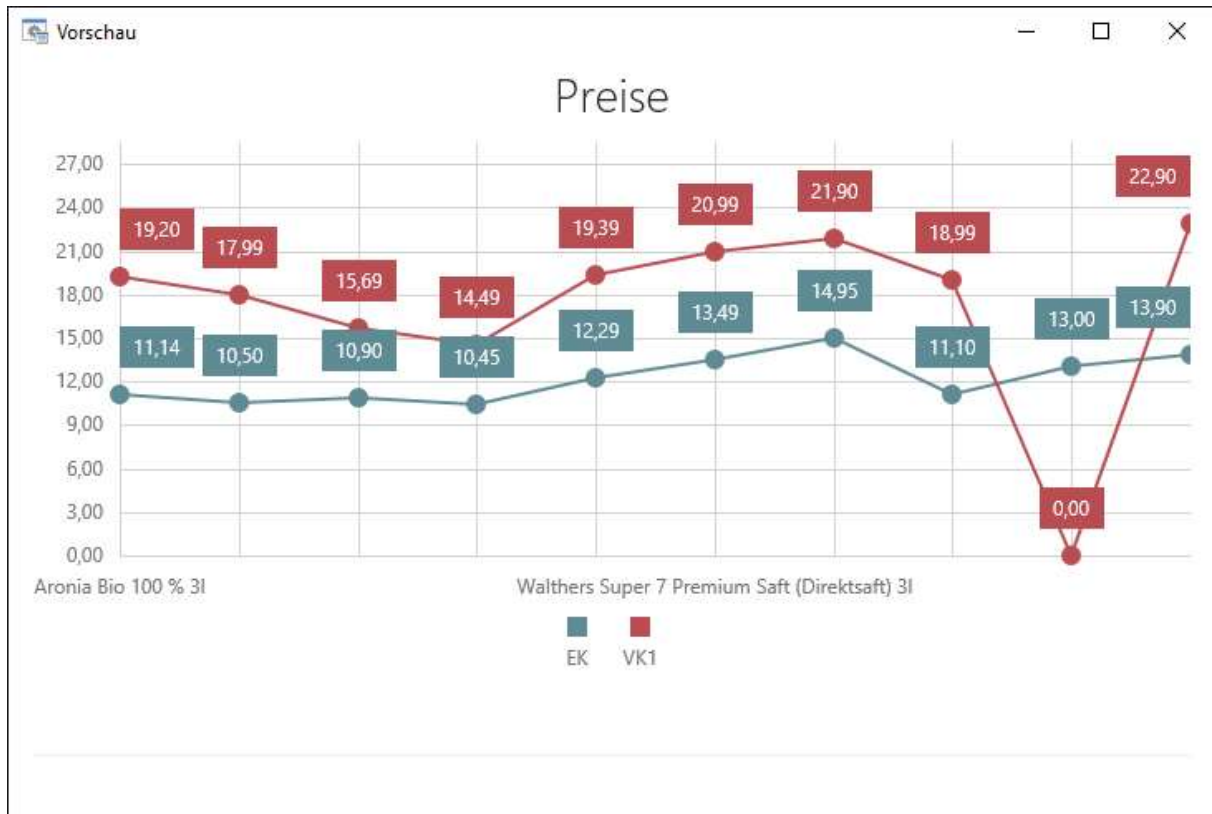
Eigenschaft	Typ	Bedeutung
WertAchse	True/False	Skala (linke Seite) anzeigen
HorizontaleWerte	True/False	Links/rechts oder unten/oben
Wertfelder	String	Namen der numerischen Felder mit den darzustellenden Werten, kommasetrennt
Bezeichnungsfeld	String	Name des Feldes, das die Bezeichner zu den Werten enthält
Nachkommastellen	Ganze Zahl	Selbsterklärend
Praefix	String	Kann zur Identifikation des Datensatzes verwendet werden
Titel	String	Überschrift („Preise“ in der Abbildung)
WertBeschriftung	True/False	Werte am Säulenende anzeigen
ImmerAktualisieren	True/False	Sollen Werte bei jedem Aufruf neu vom Server abgefragt werden?
ConnectionString	String	Wenn belegt, wird unabhängig von der globalen eine eigene Verbindung aufgebaut, z.B. zu einer anderen Datenbank
SQL	String	Enthält die auszuführende Abfrage (Wertfeld und Bezeichnungsfeld müssen natürlich enthalten sein)

Bereichsdiagramm



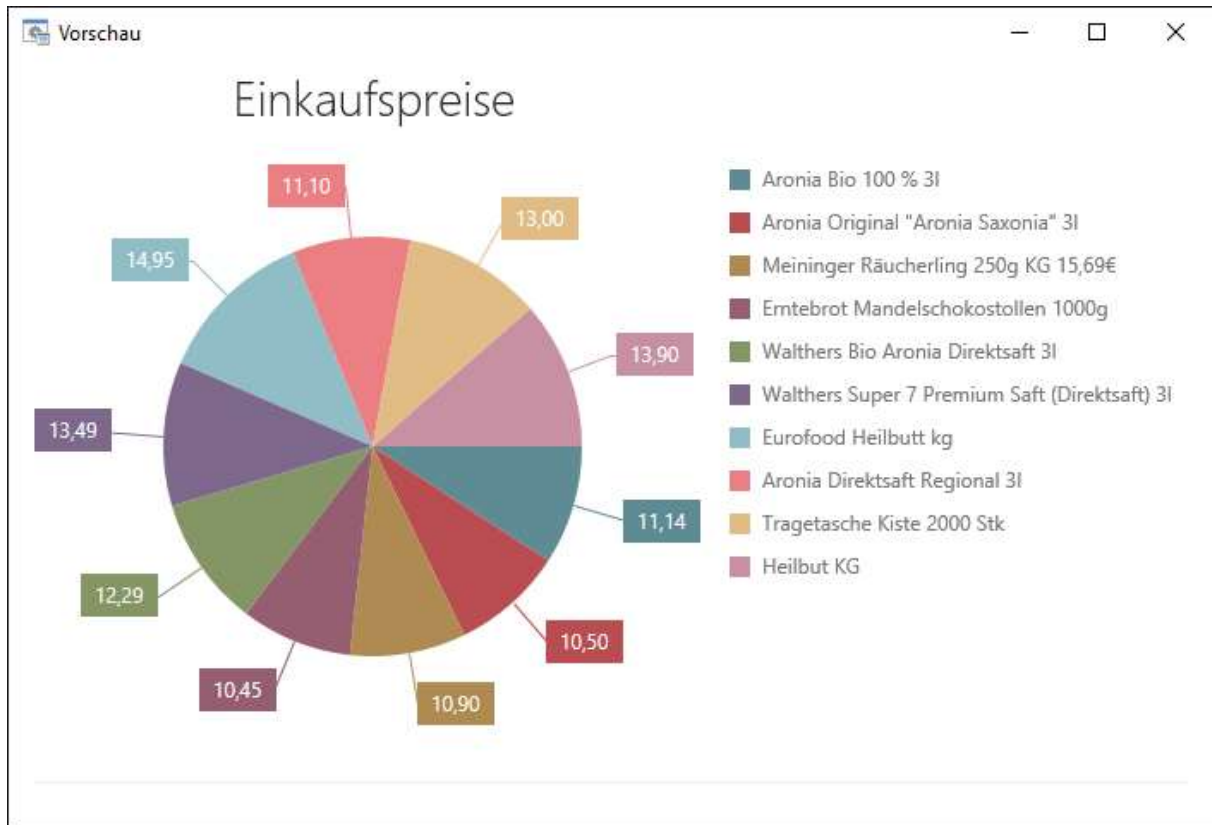
Eigenschaft	Typ	Bedeutung
WertAchse	True/False	Skala (linke Seite) anzeigen
HorizontaleWerte	True/False	Links/rechts oder unten/oben
Wertfelder	String	Namen der numerischen Felder mit den darzustellenden Werten, kommasetrennt
Bezeichnungsfeld	String	Name des Feldes, das die Bezeichner zu den Werten enthält
Nachkommastellen	Ganze Zahl	Selbsterklärend
Praefix	String	Kann zur Identifikation des Datensatzes verwendet werden
Titel	String	Überschrift („Preise“ in der Abbildung)
WertBeschriftung	True/False	Werte in Spitzen/Tälern anzeigen
ImmerAktualisieren	True/False	Sollen Werte bei jedem Aufruf neu vom Server abgefragt werden?
ConnectionString	String	Wenn belegt, wird unabhängig von der globalen eine eigene Verbindung aufgebaut, z.B. zu einer anderen Datenbank
SQL	String	Enthält die auszuführende Abfrage (Wertfeld und Bezeichnungsfeld müssen natürlich enthalten sein)

Liniendiagramm



Eigenschaft	Typ	Bedeutung
WertAchse	True/False	Skala (linke Seite) anzeigen
HorizontaleWerte	True/False	Links/rechts oder unten/oben
Wertfelder	String	Namen der numerischen Felder mit den darzustellenden Werten, kommasetrennt
Bezeichnungsfeld	String	Name des Feldes, das die Bezeichner zu den Werten enthält
Nachkommastellen	Ganze Zahl	Selbsterklärend
Praefix	String	Kann zur Identifikation des Datensatzes verwendet werden
Titel	String	Überschrift („Preise“ in der Abbildung)
WertBeschriftung	True/False	Werte über den Punkten anzeigen
ImmerAktualisieren	True/False	Sollen Werte bei jedem Aufruf neu vom Server abgefragt werden?
ConnectionString	String	Wenn belegt, wird unabhängig von der globalen eine eigene Verbindung aufgebaut, z.B. zu einer anderen Datenbank
SQL	String	Enthält die auszuführende Abfrage (Wertfeld und Bezeichnungsfeld müssen natürlich enthalten sein)

Tortendiagramm



Eigenschaft	Typ	Bedeutung
Legende	True/False	Legende anzeigen
InlineLegende	Aufzählung	Bezeichner zu Werten anzeigen (siehe nachfolgende Tabelle)
Typ	Aufzählung	Torte oder Donut („Loch“ im Kreis)
Wertfeld	String	Name des numerischen Feldes mit den darzustellenden Werten
Bezeichnungsfeld	String	Name des Feldes, das die Bezeichner zu den Werten enthält
Nachkommastellen	Ganze Zahl	Selbsterklärend
Praefix	String	Kann zur Identifikation des Datensatzes verwendet werden
Titel	String	Überschrift („Einkaufspreise“ in der Abbildung)
WertBeschriftung	True/False	Werte zu Kreisausschnitt anzeigen (siehe auch InlineLegende)
ImmerAktualisieren	True/False	Sollen Werte bei jedem Aufruf neu vom Server abgefragt werden?
ConnectionString	String	Wenn belegt, wird unabhängig von der globalen eine eigene Verbindung aufgebaut, z.B. zu einer anderen Datenbank
SQL	String	Enthält die auszuführende Abfrage (Wertfeld und Bezeichnungsfeld müssen natürlich enthalten sein)

Einstellung	Darstellung
Ohne	Nur Wert
Davor	Bezeichner Wert
Danach	Wert Bezeichner
DavorMitDoppelpunkt	Bezeichner: Wert
DanachMitDoppelpunkt	Wert: Bezeichner
DavorMitKlammern	Bezeichner (Wert)
DanachMitKlammern	Wert (Bezeichner)

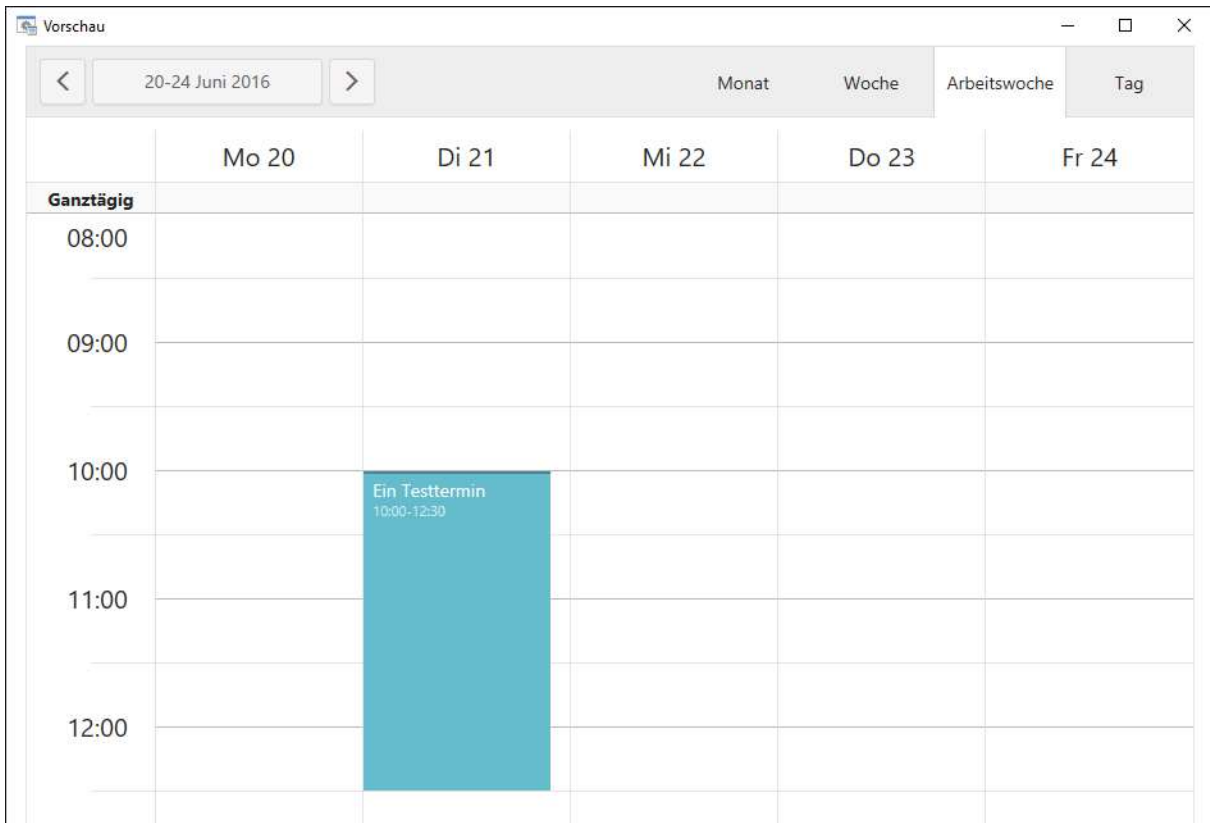
Tabelle

The screenshot shows a window titled 'Vorschau' with a subtitle 'Preise'. It contains a table with the following data:

Artikel	Bezeichnung	EK	VK1
2582	Aronia Bio 100 % 3l	11,14	19,20
2852	Aronia Original "Aronia...	10,50	17,99
2959	Meininger Räucherling...	10,90	15,69
3918	Erntebrot Mandelschok...	10,45	14,49
3300	Walthers Bio Aronia Dir...	12,29	19,39
4260	Walthers Super 7 Premi...	13,49	20,99
4689	Eurofood Heilbutt kg	14,95	21,90
4709	Aronia Direktsaft Regio...	11,10	18,99
4760	Tragetasche Kiste 2000...	13,00	
5025	Heilbut KG	13,90	22,90

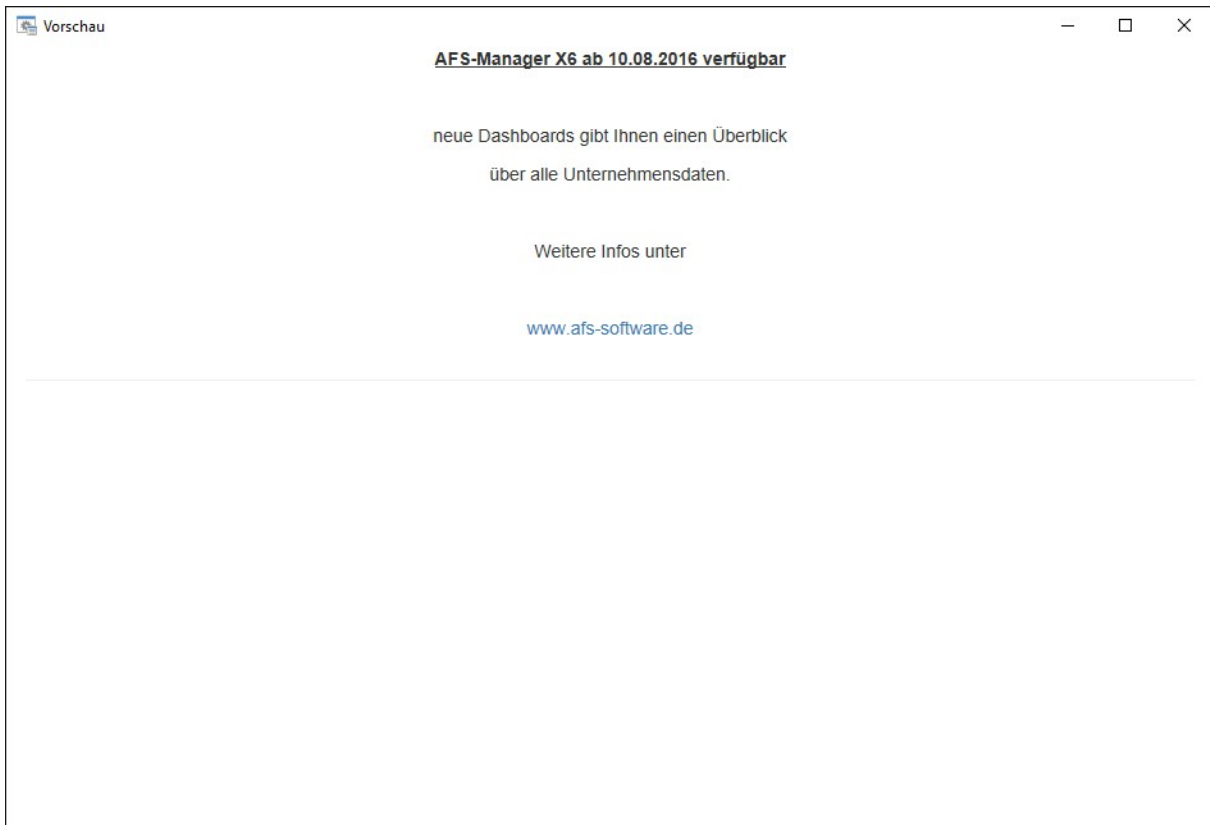
Eigenschaft	Typ	Bedeutung
IDFeld	String	Name des Feldes mit dem Primärschlüssel (optional)
IDFeldAnzeigen	True/False	Primärschlüssel darstellen?
Wertfelder	String	Namen der numerischen Felder mit den darzustellenden Werten, kommasetrennt
Bezeichnungsfeld	String	Name des Feldes, das die Bezeichner zu den Werten enthält
Nachkommastellen	Ganze Zahl	Selbsterklärend
Praefix	String	Kann zur Identifikation des Datensatzes verwendet werden
Titel	String	Überschrift („Preise“ in der Abbildung)
ImmerAktualisieren	True/False	Sollen Werte bei jedem Aufruf neu vom Server abgefragt werden?
ConnectionString	String	Wenn belegt, wird unabhängig von der globalen eine eigene Verbindung aufgebaut, z.B. zu einer anderen Datenbank
SQL	String	Enthält die auszuführende Abfrage (Wertfeld und Bezeichnungsfeld müssen natürlich enthalten sein)

Kalender



Eigenschaft	Typ	Bedeutung
Ansicht	Aufzählung	Initial angezeigter Tab-Reiter (Monat/Woche/Arbeitswoche/Tag)
Initialdatum	Datum	Angezeigtes Datum (optional), Voreinstellung aktueller Tag
StartFeld	String	Name des DateTime-Feldes mit dem Startzeitpunkt der Termine
EndeFeld	String	Name des DateTime-Feldes mit dem Endzeitpunkt der Termine
BetreffFeld	String	Name des Feldes, das die Kurzbezeichnung der Termine enthält
Beschreibungfeld	String	Name des Feldes, das den Langtext zum Termin enthält
BenutzerIDFeld	String	Name des Feldes, das den Primärschlüssel des Teilnehmer-Datensatzes enthält
BenutzernameFeld	String	Name des Feldes, das den Namen des Teilnehmers enthält
ImmerAktualisieren	True/False	Sollen Werte bei jedem Aufruf neu vom Server abgefragt werden?
ConnectionString	String	Wenn belegt, wird unabhängig von der globalen eine eigene Verbindung aufgebaut, z.B. zu einer anderen Datenbank
SQL	String	Enthält die auszuführende Abfrage (Wertfeld und Bezeichnungsfeld müssen natürlich enthalten sein)
Titel	String	Überschrift (z.B. „Terminübersicht Meier“)

Externe Inhalte



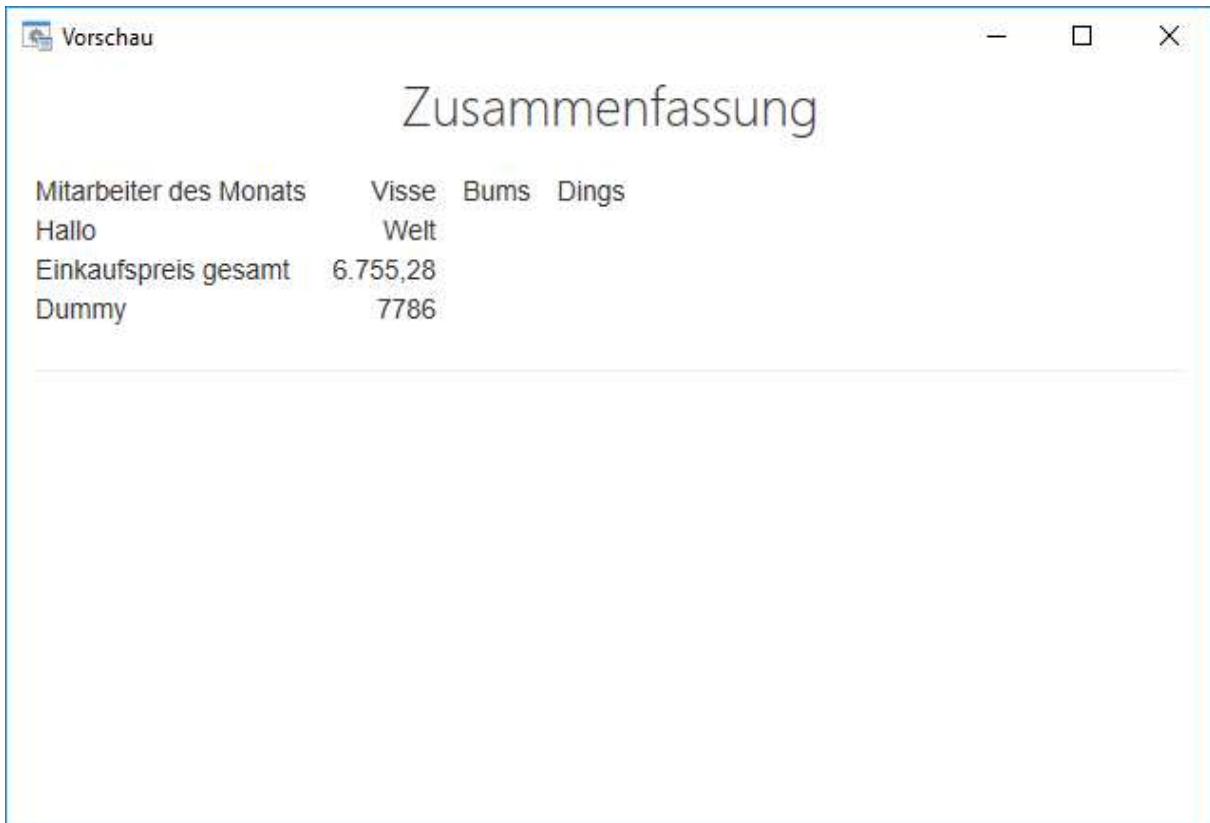
Eigenschaft	Typ	Bedeutung
Timeout	Ganze Zahl	Maximale Wartezeit (im Millisekunden) für Verbindung und Server-Antwort
URL	String	URL des anzuzeigenden HTML-Inhaltes. Sollte kein Protokoll angegeben sein (z.B. nur www.afs-software.com), wird automatisch http verwendet.

Benutzerdefiniert



Eigenschaft	Typ	Bedeutung
Scriptname	String	Name der Datei, die das benutzerdefinierte Javascript enthält. Diese Datei muss zwingend im Template-Verzeichnis liegen, andernfalls wird sie nicht gefunden.

Zusammenfassung



Eigenschaft	Typ	Bedeutung
ImmerAktualisieren	True/False	Sollen Werte bei jedem Aufruf neu vom Server abgefragt werden?
ConnectionString	String	Wenn belegt, wird unabhängig von der globalen eine eigene Verbindung aufgebaut, z.B. zu einer anderen Datenbank
SQLDatei	String	Name der Datei, die die auszuführenden SQL-Befehle enthält. Sie muss sich im Templates-Verzeichnis befinden.
Titel	String	Überschrift („Zusammenfassung“ in der Abbildung)
TabellenVerwenden	True/False	Werte in tabellarischer Form anzeigen (Ausrichtung)
Nachkommastellen	Ganze Zahl	Selbsterklärend

SQL-Befehle in obiger Datei (Abbildung):

```
SELECT MAX(Name) AS 'Mitarbeiter des Monats', 'Dings' AS 'Bums' FROM Adressen
SELECT 'Welt' AS 'Hallo' FROM Adressen LIMIT 1
SELECT SUM(EK) AS 'Einkaufspreis gesamt' FROM Artikel
SELECT MAX(Artikel) AS Dummy FROM Artikel
```

Jede SQL-Abfrage muss sich in einer einzelnen Zeile befinden. Je Feld der Ergebnismenge werden 2 Spalten gebildet: Feldbezeichnung und Feldwert. Gibt eine Abfrage also 3 Werte zurück, erhält man 6 Spalten. Bei tabellarischer Darstellung werden der Feldbezeichner links- und der Feldwert rechtsbündig ausgerichtet.