

# ExtendedRelation 1.50

## Verzeichnis der Funktionen

Funktion	<b>ts_dtos</b>	(double to string)
Aufruf	ts_dtos( FLOAT [, FORMAT] )	
	FLOAT: Zu konvertierende reelle Zahl	Typ reelle Zahl
	FORMAT: Formatierungsanweisung	Typ Zeichenkette, optional
Rückgabe	Formatierte Zeichenkette	Typ Zeichenkette
Beschreibung	<p>Verwandelt eine reelle Zahl in eine Zeichenkette. Über den Parameter FORMAT kann die Formatierung näher festgelegt werden. Beispiele für Format:</p> <ul style="list-style-type: none"> <li>"%f": Fließkommawert</li> <li>"%e": Exponentenschreibweise</li> <li>"%g": wie %f oder %e, je nachdem was kürzer ist.</li> <li>"%7.3f" : Fließkomma, Feldbreite 10, 3 Nachkommastellen, rechtsbündig</li> <li>"%-7.3f" : Fließkomma, Feldbreite 10, 3 Nachkommastellen, linksbündig</li> <li>"%f %%": Fließkommawert mit Prozentzeichen</li> </ul> <p>Beispiele:</p> <pre>s_value=ts_dtos(10.349, "%.1f") → s_value hat den (abgerundeten) Wert "10.3" s_value=ts_dtos(10.350, "%.1f%%") → s_value hat den (aufgerundeten) Wert "10.4%"</pre>	

Funktion	<b>ts_env</b>	(environment)
Aufruf	ts_env ( ENV )	
	ENV: Umgebungsvariable	Typ Zeichenkette
Rückgabe	Wert der Umgebungsvariable	Typ Zeichenkette
Beschreibung	<p>Liefert den Wert einer Umgebungsvariablen. Existiert dieser Wert nicht, wird eine leere Zeichenkette zurückgegeben. Beispiel:</p> <pre>ERSTELLER = ts_env("USERNAME")</pre>	

Funktion	<b>ts_itos</b>	(integer to string)
Aufruf	ts_itos( INTEGER [, FORMAT] )	
	INTEGER: Zu konvertierende Ganzzahl	Typ Ganzzahl
	FORMAT: Formatierungsanweisung	Typ Zeichenkette, optional
Rückgabe	Formatierte Zeichenkette	Typ Zeichenkette
Beschreibung	<p>Verwandelt eine Ganzzahl in eine Zeichenkette. Über den Parameter FORMAT kann die Formatierung näher festgelegt werden. Beispiele für Format:</p> <ul style="list-style-type: none"> <li>"%d" : Dezimalwert</li> <li>"%o" : Oktalwert</li> <li>"%x" : Hexadezimalwert (kleine Buchstaben)</li> <li>"%X" : Hexadezimalwert (große Buchstaben)</li> <li>"%c" : einzelnes Zeichen</li> </ul>	

	<p>"%10d" : Dezimalwert, Feldbreite 10, rechtsbündig  "%-10d" : Dezimalwert, Feldbreite 10, linksbündig  "%d %% " : Dezimalwert mit Prozentsymbol</p> <p>Beispiele:</p> <pre>s_value= ts_itos( 255, "%7d") → s_value hat den Wert " 255" s_value = ts_itos(255, "%X") → s_value hat den Wert "FF"</pre>
--	---

Funktion	<b>ts_model_name()</b> <b>ts_model_type()</b> <b>ts_model_path()</b> <b>ts_model_device()</b> <b>ts_model_host()</b> <b>ts_model_version()</b> <b>ts_model_subclass()</b>	
Aufruf	ts_model_<bezeichner>()	
	Keine Übergabeparameter	
Rückgabe	Modeldatei-Information	Typ Zeichenkette und Ganzzahl
Beschreibung	ts_model_name(): Modelname in Kleinbuchstaben. Beispiel: welle  ts_model_type(): Datei Suffix in Kleinbuchstaben. Beispiel: prt  ts_model_path(): Dateipfad. Beispiel: \users\proe\  ts_model_device(): Laufwerk, wenn vorhanden: Beispiel: d  ts_model_host(): Hostname, wenn vorhanden (UNC Pfade). Beispiel: cad01  ts_model_version(): Versionsnummer der Datei. Beispiel: 4  ts_model_subclass(): Untertyp. Beispiel Blech: 22	

Funktion	<b>ts_param_by_name</b>	
Aufruf	ts_param_by_name ( PARAMETERNAME)	
	PARAMATERNAME: Name des Parameters	Typ Zeichenkette
Rückgabe	Wert des Parameters	Typ abhängig von Parameter
Beschreibung	Liefert den Wert eines Parameters der als Zeichenkette angegeben wurde. Dieser Funktion kann auch ein Wert zugeordnet werden, um den Parameter zu setzen	

	<b>Beispiel:</b> VERSION = ts_param_by_name("VERSION" + "_" + real_model_name) oder ts_param_by_name("VERSION" + "_" + real_model_name)= "B"
--	---

<b>Funktion</b>	<b>ts_outline_compute</b>	
<b>Aufruf</b>	ts_outline( [CSYS] )	
	CSYS: Name des Bezugskoordinatensystems	Typ Zeichenkette, optional
<b>Rückgabe</b>	TRUE, wenn erfolgreich FALSE, bei Fehler	Typ Ja/Nein
<b>Beschreibung</b>	Berechnet die Abmessungen in X, Y und Z bezogen auf das Standardkoordinatensystem, bzw. bezogen auf CSYS. Sollte das Bezugskoordinatensystem nicht gefunden werden, so bricht diese Funktion nicht ab, sondern liefert den Wert FALSE zurück. Darauf folgende Aufrufe von ts_outline_x etc. würden dann jedoch fehlschlagen. <b>Beispiel:</b> if ts_outline_compute("CS0") HAUPTMASSE = ts_dtos( ts_outline_x , "%.1f" ) + "x" \ + ts_dtos( ts_outline_y , "%.1f" ) else if ts_outline_compute("C_0") HAUPTMASSE = ts_dtos( ts_outline_x , "%.1f" ) + "x" \ + ts_dtos( ts_outline_y , "%.1f" ) else HAUPTMASSE = "Fehler: Weder CS0 noch C_0 gefunden." endif endif	

<b>Funktion</b>	<b>ts_outline_x, ts_outline_y, ts_outline_z</b>	
<b>Aufruf</b>	ts_outline_x,ts_outline_y, ts_outline_z	
<b>Rückgabe</b>	Hauptmaß in X, Y oder Z	Typ reelle Zahl
<b>Beschreibung</b>	Liefert die Abmessungen in X, Y bzw. Z zurück, die vorher mit ts_outline_compute berechnet wurden. Wird ts_outline_compute nicht vorher aufgerufen, so schlagen die Aufrufe dieser Funktionen fehl. <b>Beispiel:</b> if ts_outline_compute("CS0") HAUPTMASSE = ts_dtos( ts_outline_x , "%.1f" ) + "x" \ + ts_dtos( ts_outline_y , "%.1f" ) endif	

<b>Funktion</b>	<b>ts_smtcontour_compute</b>	
<b>Aufruf</b>	ts_smtcontour_compute()	
	Kein Übergabeparameter	
<b>Rückgabe</b>	TRUE wenn Blechmodel False wenn kein Blech	Typ Ja/Nein
<b>Beschreibung</b>	Berechnet der äußere und innere Kontourlänge eines Bleches. <b>Beispiel:</b> If ts_smtcontour_compute() OUTER = ts_smtcontour_outer() INNER = ts_smtcontour_inner() endif	

Funktion	<b>ts_smtcontour_outer,</b> <b>ts_smtcontour_inner</b>	
Aufruf	ts_smtcontour_outer() ts_smtcontour_inner()	
	Kein Übergabeparameter	
Rückgabe	Kontourlängen	Typ reelle Zahl
Beschreibung	Liefert die äußere und innere Kontourlänge eines Bleches. Es wird die „weisse“ Seite des Bleches berechnet. Sinnvoll eigentlich nur im abgewickelten Zustand. Beispiel: <pre>If ts_smtcontour_compute()     OUTER = ts_smtcontour_outer()     INNER = ts_smtcontour_inner() Endif</pre> Es wird die „weisse“ Seite des Bleches berechnet. Sinnvoll eigentlich nur im abgewickelten Zustand.	

Funktion	<b>ts_stod</b>	(string to double)
Aufruf	ts_stod ( STRING )	
	STRING: Zu konvertierende Zeichenkette	Typ Zeichenkette
Rückgabe	Konvertierter Wert	Typ Reelle Zahl
Beschreibung	Verwandelt eine Zeichenkette in eine reelle Zahl. Beispiele: <pre>d_value=stod("10.345") → d_value hat den Wert 10.345 d_value=stod("10.3e2") → d_value hat den Wert 1030.0</pre>	

Funktion	<b>ts_stoi</b>	(string to integer)
Aufruf	ts_stoi ( STRING [, BASE])	
	STRING: Zu konvertierende Zeichenkette	Typ Zeichenkette
	BASE: Zahlensystem	Typ Ganzzahl, optional
Rückgabe	Konvertierter Wert	Typ Ganzzahl
Beschreibung	Verwandelt eine Zeichenkette in eine Ganzzahl. Der optionale Parameter BASE gibt das Zahlensystem (2-36) an, das dabei angewendet werden soll. Beispiele: <pre>i_value = ts_stoi("255", 10)     → i_value hat den Dezimalwert 255 i_value = ts_stoi("FF", 16)     → i_value hat den Dezimalwert 255 i_value=ts_stoi("11",2)     → i_value hat den Dezimalwert 3</pre>	

Funktion	<b>ts_time</b>	
Aufruf	ts_time ( FORMAT [, LOCALE])	
	FORMAT: Formatierung der Zeitausgabe	Typ Zeichenkette
	LOCALE: Ländereinstellungen	Typ Zeichenkette
Rückgabe	Zeitangabe	Typ Zeichenkette
Beschreibung	Liefert die aktuelle lokale Zeit als Zeichenkette. Die Formatierung wird über die	

	<p>Zeichenkette FORMAT bestimmt. Folgende Formatanweisung können verwendet werden:</p> <ul style="list-style-type: none"> <li>%a abgekürzter Wochentag</li> <li>%A Wochentag</li> <li>%b abgekürzter Monat</li> <li>%B Monat</li> <li>%c lokale Darstellung von Datum und Uhrzeit</li> <li>%d Tag im Monat</li> <li>%H Stunde (00-23)</li> <li>%I Stunde (01-12)</li> <li>%j Tag im Jahr</li> <li>%m Monat</li> <li>%M Minute</li> <li>%p lokales Äquivalent von AM und PM</li> <li>%S Sekunde</li> <li>%U Woche im Jahr (Sonntag ist erster Wochentag)</li> <li>%w Wochentag (0-6, Sonntag ist 0)</li> <li>%W Woche im Jahr (Montag ist erster Wochentag)</li> <li>%x lokale Darstellung des Datums</li> <li>%X lokale Darstellung der Zeit</li> <li>%y Jahr ohne Jahrhundert</li> <li>%Y Jahr mit Jahrhundert</li> <li>%z Name der Zeitzone</li> <li>%% %</li> </ul> <p>Der optionale Parameter LOCALE legt die Ländereinstellungen fest. Ist er nicht belegt, so werden die aktuellen Ländereinstellungen des Rechners verwendet.</p> <p>Beispiele:</p> <pre> BEARB_DAT = ts_time("%c", "german"); → BEARB_DAT hat den Wert "31.01.2005 22:45" BEARB_DAT = ts_time("%c", "english"); → BEARB_DAT hat den Wert "01/31/2005 10:45 PM" BEARB_DAT = ts_time("%A", "spanish"); → BEARB_DAT hat den Wert "Viernes" </pre>
--	--

Funktion	<b>ts_unit</b>	
Aufruf	ts_unit ( UNITTYPE)	
	UNITTYPE: Art der Einheit	Typ Zeichenkette
Rückgabe	Einheit	Typ Zeichenkette
Beschreibung	<p>Liefert den Namen der vom Model verwendeten Einheit. Für die Einheitenart (Parameter UNITTYPE) sind folgende Werte zulässig:</p> <p>LENGTH  MASS  FORCE  TIME  TEMPERATURE  ANGLE</p> <p>Beispiel:</p> <pre> LAENGEN_EINHEIT = "[" + ts_unit("LENGTH") + "]" KRAFT_EINHEIT = "[" + ts_unit("FORCE") + "]" → in einem Masse, Länge, Zeit System hat KRAFT_EINHEIT den Wert [f(kg,sec,mm)] </pre>	

