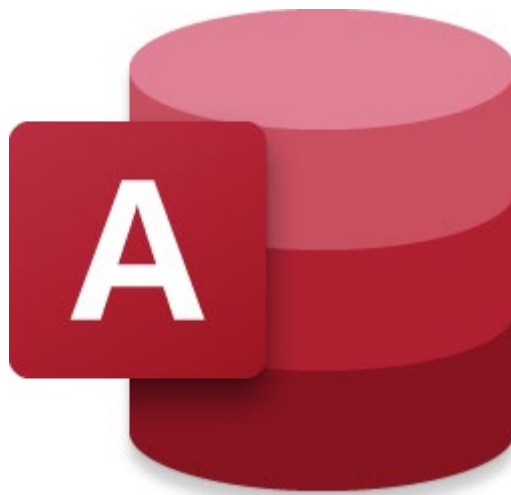




Access für Microsoft 365

Integrierte Funktionen



Inhaltsverzeichnis

Einleitung	2
Vorbemerkung	2
Liste der Funktionen	2
Allgemein	2
Arrays	3
Datenbank.....	3
Datum/Uhrzeit	4
Domänenaggregat.....	6
Fehlerbehandlung	7
Finanzmathematisch.....	8
Inspektion	9
Konvertierung.....	10
Mathematisch	14
Meldungen.....	15
Programmablauf.....	16
SQL-Aggregat	16
Text.....	17
Funktionen mit und ohne Dollarzeichen	21
Fachbegriffe.....	21

Einleitung

In diesem Skript bekommen Sie eine kurze Auflistung über die in Access integrierten Funktionen (inkl. einer Kurzbeschreibung), die Sie z.B. in Abfragen für berechnete Felder oder auch in Formularen und Berichten einsetzen können. Das gilt zwar nicht zwangsläufig für alle integrierten Funktionen, aber doch für die meisten. Die Funktionen sind nach Kategorien geordnet, so wie Sie sie im Dialogfeld **Ausdrucks-Generator** finden (siehe Skript **Access für Microsoft 365 – Abfragen (inkl. Musterlösung)**, Abbildung 6, Seite 7). Die Auflistung gilt für die Version **Access für Microsoft 365**. Sie kann bei älteren Access-Versionen abweichen.

Vorbemerkung

In der Auflistung der integrierten Funktionen sehen Sie neben der Beschreibung der Funktion auch noch die Syntaxschreibweise. Wenn eine Funktion mehrere Funktionsargumente enthält, werden diese jeweils durch ein Semikolon voneinander getrennt (Beispiele siehe Skript **Access für Microsoft 365 - Abfragen**). Bei den Funktionsargumenten kann es sich auch um optionale Argumente handeln, d.h. Sie können bei Verwendung der Funktion das Argument angeben oder auch weglassen. Geben Sie das Argument nicht an, lässt Access in ganz weg oder setzt stattdessen einen Standardwert ein. Optionale Argumente werden in der Syntaxschreibweise in eckige Klammer ([und]) eingeschlossen dargestellt. Wenn Sie bei der späteren Verwendung der Funktion optionale Argumente angeben, dürfen Sie aber nicht die eckigen Klammern mit eingeben (es sei denn, es handelt sich um den Feldnamen einer Tabelle oder Abfrage in der Datenbank). Wie bereits in der Einleitung erwähnt, enthält Skript nur eine Kurzbeschreibung der Funktionen. Wenn Sie Informationen zu den Funktionsargumenten und Beispiele sehen wollen, sollten sich folgende Webseite anschauen:

<https://support.microsoft.com/de-de/office/funktionen-nach-kategorien-b8b136c3-2716-4d39-94a2-658ce330ed83>

Übrigens hat die Groß-/Kleinschreibweise der Funktionsnamen keine Bedeutung für Access. Sie können also bei Verwendung der Funktionen die Namen komplett in Kleinbuchstaben eingeben.

Liste der Funktionen

Allgemein

<i>Funktion</i>	<i>Beschreibung</i>
Farbe	Farbe(Farbnummer) Gibt einen Wert vom Typ <i>Long Integer</i> zurück, der den RGB-Farbcode angibt, der der angegebenen Farbnummer entspricht.
RGB	RGB(Rot;Grün;Blau) Gibt einen Wert vom Typ <i>Long Integer</i> zurück, der einem RGB-Farbwert entspricht.

Arrays

Das englische Wort *Array* steht im Computerbereich für das deutsche Wort *Matrix*. Dabei handelt es sich üblicherweise um eine rechteckige Anordnung von mathematischen Objekten (Zahlen, Variablen, usw.).

Funktion	Beschreibung
LBound	LBound(Array;[Dimension]) Gibt einen Wert vom Typ <i>Long Integer</i> zurück, der den kleinsten verfügbaren Index enthält, den es für die angegebene Dimension eines Arrays gibt.
UBound	UBound(Array;[Dimension]) Gibt einen Wert vom Typ <i>Long Integer</i> zurück, der den größten verfügbaren Index enthält, den es für die angegebene Dimension eines Arrays gibt.

Datenbank

Funktion	Beschreibung
AktuellerBenutzer	AktuellerBenutzer() Gibt den momentan aktiven Benutzer der Datenbank zurück.
AktuellerWebbenutzer	AktuellerWebbenutzer(Typ) Gibt in veröffentlichten Anwendungen einen Wert vom Typ <i>String</i> zurück, der den Benutzer angibt, der momentan angemeldet ist. Gibt in Nur-Client-Anwendungen als Ergebnis NULL zurück.
AktuelleWebbenutzergruppen	AktuelleWebbenutzergruppen(Typ) Gibt in veröffentlichten Anwendungen einen Wert vom Typ <i>String</i> zurück, in dem die Gruppen aufgelistet werden, zu denen der Benutzer gehört, der momentan angemeldet ist. der den Benutzer angibt, der momentan angemeldet ist. Gibt in Nur-Client-Anwendungen als Ergebnis NULL zurück.
Auswerten	Auswerten(Zeichenfolge) Wertet einen Ausdruck aus, dessen Wert eine Textzeichenfolge oder ein numerischer Wert ist.
Bereich	Bereich(Zahl;Startwert;Stopwert;Intervall) Gibt einen Wert vom Typ <i>String</i> zurück, der angibt, wo in einer berechneten Reihe von Bereichen eine bestimmte Zahl enthalten ist.
IMEStatus	IMEStatus() Gibt einen Wert vom Typ <i>Integer</i> zurück, der den aktuellen Modus für den Eingabemethoden-Editor (Input Method Editor, kurz: IME) von Microsoft Windows angibt. Ist nur in ostasiatischen Versionen verfügbar.

Funktion	Beschreibung
IstAktuellerWebbenutzerInGruppe	IstAktuellerWebbenutzerInGruppe(Gruppe) Gibt in veröffentlichten Anwendungen einen Wert vom Typ <i>Boolean</i> zurück, der angibt, ob der Benutzer, der momentan angemeldet ist, zu der angegebenen Gruppe gehört. Gibt in Nur-Client-Anwendungen <i>Falsch</i> (False) zurück.
LinkPart	LinkPart(Link;Teil) Gibt einen Wert vom Typ <i>String</i> zurück, der dem URL des angegebenen Link-Felds entspricht.
Spaltenverlauf	Spaltenverlauf(Tabellenname;Spaltenname; Abfragezeichenfolge) Gibt einen Wert vom Typ <i>String</i> zurück, der den Verlauf von Werten enthält, die in einem Memo-Feld gespeichert sind.

Datum/Uhrzeit

Funktion	Beschreibung
DatAdd	DatAdd(Intervall;Zahl;Datum) Gibt einen Wert vom Typ <i>Datum</i> zurück, der ein Datum enthält, zu dem ein angegebenes Zeitintervall addiert wurde.
DatDiff	DatDiff(Intervall;Datum1;Datum2;[ErsterWochentag]; [ErsteWocheDesJahres]) Gibt einen Wert vom Typ <i>Long Integer</i> zurück, der angibt, wie viele Zeitintervalle zwischen zwei Datumsangaben liegen.
DatSeriell	DatSeriell(Jahr;Monat;Tag) Gibt einen Wert vom Typ <i>Datum</i> für eine Tagesangabe (Jahr, Monat und Tag) zurück.
DatTeil	DatTeil(Intervall;Datum;[ErsterWochentag]; [ErsteWocheDesJahres]) Gibt einen Wert vom Typ <i>Integer</i> zurück, der den angegebenen Teil eines bestimmten Datums angibt.
Datum	Datum() Gibt einen Wert vom Typ <i>Datum</i> zurück, der das aktuelle Systemdatum enthält.
Datum\$	Datum\$() Gibt einen Wert vom Typ <i>String</i> zurück, der das aktuelle Systemdatum enthält.
DatWert	DatWert(Datum) Gibt einen Wert vom Typ <i>Datum</i> zurück.

Funktion	Beschreibung
IstDatum	IstDatum(Ausdruck) Gibt einen Wert vom Typ <i>Boolean</i> zurück, der angibt, ob ein Ausdruck in ein Datum umgewandelt werden kann.
Jahr	Jahr(Datum) Gibt einen Wert vom Typ <i>Integer</i> zurück, der die ganze Zahl angibt, die dem Jahr entspricht.
Jetzt	Jetzt() Gibt einen Wert vom Typ <i>Datum</i> zurück, der das aktuelle Systemdatum und die aktuelle Systemzeit enthält.
Minute	Minute(Zeit) Gibt einen Wert vom Typ <i>Integer</i> zurück, der eine ganze Zahl von 1 bis einschließlich 59 angibt, die einer Minute einer Stunde entspricht.
Monat	Monat(Datum) Gibt einen Wert vom Typ <i>Integer</i> zurück, der eine ganze Zahl von 1 bis einschließlich 12 angibt, die einem Monat eines Jahres entspricht.
Monatsname	Monatsname(Monat;[Kurzform]) Gibt einen Wert vom Typ <i>String</i> zurück, der den angegebenen Monat enthält.
Sekunde	Sekunde(Zeit) Gibt einen Wert vom Typ <i>Integer</i> zurück, der eine ganze Zahl von 1 bis einschließlich 59 angibt, die einer Sekunde einer Minute entspricht.
Stunde	Stunde(Zeit) Gibt einen Wert vom Typ <i>Integer</i> zurück, der eine ganze Zahl von 1 bis einschließlich 23 angibt, die einer Stunde eines Tages entspricht.
Tag	Tag(Datum) Gibt einen Wert vom Typ <i>Integer</i> zurück, der eine ganze Zahl von 1 bis einschließlich 31 angibt, die einem Tag eines Monats entspricht.
Wochentag	Wochentag(Datum;[ErsterWochentag]) Gibt einen Wert vom Typ <i>Integer</i> zurück, der die ganze Zahl angibt, die dem Wochentag entspricht.
Wochentagsname	Wochentagsname(Wochentag;[Kurzform];[ErsterWochentag]) Gibt einen Wert vom Typ <i>String</i> zurück, der den angegebenen Wochentag enthält.
ZDate	ZDate(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>Datum</i> um.

Funktion	Beschreibung
Zeit	Zeit() Gibt einen Wert vom Typ <i>Datum</i> zurück, der die aktuelle Systemzeit angibt.
Zeit\$	Zeit\$() Gibt einen Wert vom Typ <i>String</i> zurück, der die aktuelle Systemzeit angibt.
Zeitgeber	Zeitgeber() Gibt einen Wert vom Typ <i>Single</i> zurück, der die Anzahl von Sekunden angibt, die seit Mitternacht vergangen sind.
ZeitSeriell	ZeitSeriell(Stunde;Minute;Sekunde) Gibt einen Wert vom Typ <i>Datum</i> zurück, der die Uhrzeit für eine bestimmte Stunde, Minute und Sekunde enthält.
ZeitSeriellStr	ZeitSeriellStr(Zeit) Gibt einen Wert vom Typ <i>Datum</i> zurück, der die Uhrzeit enthält.
ZVarDat	ZVarDat(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>Datum</i> um.

Domänenaggregat

Funktion	Beschreibung
DomAnzahl	DomAnzahl(Ausdruck;Domäne;[Kriterien]) Ermittelt die Anzahl von Datensätzen, die zu einer angegebenen Datensatzgruppe (eine Domäne) gehören.
DomErsterWert	DomErsterWert(Ausdruck;Domäne;[Kriterien]) Gibt einen zufälligen Datensatz aus einem bestimmten Datenfeld einer Tabelle oder Abfrage zurück, wenn Sie lediglich irgendeinen Wert aus diesem Datenfeld benötigen.
DomLetzterWert	DomLetzterWert(Ausdruck;Domäne;[Kriterien]) Gibt einen zufälligen Datensatz aus einem bestimmten Datenfeld einer Tabelle oder Abfrage zurück, wenn Sie lediglich irgendeinen Wert aus diesem Datenfeld benötigen.
DomMax	DomMax(Ausdruck;Domäne;[Kriterien]) Ermittelt den größten Wert in einer angegebenen Datensatzgruppe (eine Domäne).
DomMin	DomMin(Ausdruck;Domäne;[Kriterien]) Ermittelt den kleinsten Wert in einer angegebenen Datensatzgruppe (eine Domäne).

Funktion	Beschreibung
DomMittelwert	DomMittelwert(Ausdruck;Domäne;[Kriterien]) Berechnet den Durchschnitt einer Wertemenge in einer angegebenen Datensatzgruppe (eine Domäne).
DomStAbw	DomStAbw(Ausdruck;Domäne;[Kriterien]) Berechnet die Standardabweichung für einer Wertemenge in einer angegebenen Datensatzgruppe (eine Domäne).
DomStAbwn	DomStAbwn(Ausdruck;Domäne;[Kriterien]) Wertet eine Grundgesamtheit und die Funktion DomStAbw aus, um eine Stichprobe auszuwerten.
DomSumme	DomSumme(Ausdruck;Domäne;[Kriterien]) Berechnet die Summe einer Wertemenge in einer angegebenen Datensatzgruppe (eine Domäne).
DomVarianz	DomVarianz(Ausdruck;Domäne;[Kriterien]) Berechnet die Varianz einer Wertemenge in einer angegebenen Datensatzgruppe (eine Domäne).
DomVarianzen	DomVarianzen(Ausdruck;Domäne;[Kriterien]) Wertet eine Grundgesamtheit und die Funktion DomStAbw aus, um eine Stichprobe auszuwerten.
DomWert	DomWert(Ausdruck;Domäne;[Kriterien]) Ruft den Wert eines bestimmten Datenfelds aus einer angegebenen Datensatzgruppe (eine Domäne) ab.

Fehlerbehandlung

Funktion	Beschreibung
CVerr	CVerr(Fehlernummer) Gibt einen Wert vom Typ <i>Error</i> zurück, der die angegebene Fehlernummer enthält.
FCode	FCode() Gibt das Objekt <i>FCode</i> zurück, das die Informationen zum letzten Systemfehler enthält.
FMeldung	FMeldung([Fehlernummer]) Gibt die Fehlermeldung zurück, die der angegebenen Fehlernummer entspricht.
FMeldung\$	FMeldung\$([Fehlernummer]) Gibt die Fehlermeldung zurück, die der angegebenen Fehlernummer entspricht.
IstError	IstError(Ausdruck) Gibt einen Wert vom Typ <i>Boolean</i> zurück, der angibt, ob ein Ausdruck ein Fehlerwert ist.

Funktion	Beschreibung
Zugriffsfehler	Zugriffsfehler(Fehlernummer) Gibt ein Wert vom Typ <i>String</i> zurück, der die Beschreibung des angegebenen Fehlers enthält.

Finanzmathematisch

Funktion	Beschreibung
BW	BW(Zins;AnzPerioden;Zahlung;[ZukünftigerWert];[Typ]) Gibt einen Wert vom Typ <i>Double</i> zurück, der den Barwert einer Annuität angibt (den heutigen Gesamtwert einer Reihe zukünftiger Zahlungen).
DIA	DIA(Anschaffungswert;Restwert;Nutzungsdauer;Periode) Gibt einen Wert vom Typ <i>Double</i> zurück, der die arithmetisch-degressive Abschreibung eines Wirtschaftsguts für eine bestimmte Periode angibt.
GDA	GDA(Anschaffungswert;Restwert;Nutzungsdauer;Periode; [Faktor]) Gibt einen Wert vom Typ <i>Double</i> zurück, der die Abschreibung eines Wirtschaftsguts für eine bestimmte Zeitperiode angibt, wobei die Abschreibung nach der Methode der degressiven Doppelraten-Abschreibung oder nach einer anderen von Ihnen angegebenen Methode berechnet wird.
IRR	IRR(Werte;[Schätzwert]) Gibt einen Wert vom Typ <i>Double</i> zurück, der den internen Zinsfuß für eine Reihe periodischer Cashflows (Zahlungen und Eingänge) angibt.
KAPZ	KAPZ(Zins;Periode;AnzPerioden;Barwert;[ZukünftigerWert]; [Typ]) Gibt einen Wert vom Typ <i>Double</i> zurück, der die Tilgungszahlung für eine bestimmte Periode einer Annuität angibt, für die es regelmäßige konstante Zahlungen und einen festen Zinssatz gibt.
LIA	LIA(Anschaffungswert;Restwert;Nutzungsdauer) Gibt einen Wert vom Typ <i>Double</i> zurück, der die lineare Abschreibung eines Wirtschaftsguts für eine Periode angibt.
MIRR	MIRR(Werte;Investionssatz;Reinvestionssatz) Gibt einen Wert vom Typ <i>Double</i> zurück, der den geänderten internen Zinsfuß für eine Reihe periodischer Cashflows (Zahlungen und Eingänge) angibt.

Funktion	Beschreibung
NPV	NPV(Zins;Werte) Gibt einen Wert vom Typ <i>Double</i> zurück, der den Nettobarwert einer Investition basierend auf einer Reihe von periodischen Cashflows (Zahlungen und Eingänge) und einem Rabattsatz angibt.
RMZ	RMZ(Zins;AnzPerioden;Barwert;[ZukünftigerWert];[Typ]) Gibt einen Wert vom Typ <i>Double</i> zurück, der die Zahlung für eine Annuität angibt, für die es regelmäßige konstante Zahlungen und einen festen Zinssatz gibt.
ZINS	ZINS(AnzPerioden;Zahlung;Barwert;[ZukünftigerWert];[Typ]; [Schätzwert]) Gibt einen Wert vom Typ <i>Double</i> zurück, der den Zinssatz pro Periode einer Annuität angibt.
ZINSZ	ZINSZ(Zins;Periode;AnzPerioden;Barwert;[ZukünftigerWert]; [Typ]) Gibt einen Wert vom Typ <i>Double</i> zurück, der die Zinszahlungen für eine bestimmte Periode einer Annuität angibt, für die es regelmäßige konstante Zahlungen und einen festen Zinssatz gibt.
ZW	ZW(Zins;AnzPerioden;[Barwert];[Typ]) Gibt einen Wert vom Typ <i>Double</i> zurück, der den zukünftigen Wert einer Annuität angibt, für die es regelmäßige konstante Zahlungen und einen festen Zinssatz gibt.
ZZR	ZZR(Zins;Zahlung;Barwert;[ZukünftigerWert];[Typ]) Gibt einen Wert vom Typ <i>Double</i> zurück, der die Anzahl von Perioden für eine Annuität angibt, für die es regelmäßige konstante Zahlungen und einen festen Zinssatz gibt.

Inspektion

Funktion	Beschreibung
isArray	isArray(VarName) Gibt einen Wert vom Typ <i>Boolean</i> zurück, der angibt, ob eine Variable ein Array ist.
IsError	IsError(Ausdruck) Gibt einen Wert vom Typ <i>Boolean</i> zurück, der angibt, ob ein Ausdruck ein Fehlerwert ist.
IsMissing	IsMissing(Argumentname) Gibt einen Wert vom Typ <i>Boolean</i> zurück, der angibt, ob ein optionales Argument vom Typ <i>Variant</i> an eine Prozedur übergeben wurde.

Funktion	Beschreibung
IsObject	IsObject(Bezeichner) Gibt einen Wert vom Typ <i>Boolean</i> zurück, der angibt, ob ein Bezeichner (Visual Basic) einer Objektvariablen entspricht.
IstDatum	IstDatum(Ausdruck) Gibt einen Wert vom Typ <i>Boolean</i> zurück, der angibt, ob ein Ausdruck in ein Datum umgewandelt werden kann.
IstLeer	IstLeer(Ausdruck) Gibt einen Wert vom Typ <i>Boolean</i> zurück, der angibt, ob eine Variable initialisiert wurde.
IstNull	IstNull(Ausdruck) Gibt einen Wert vom Typ <i>Boolean</i> zurück, der angibt, ob ein Ausdruck eine Wert ohne gültigen Daten hat (NULL).
IstNumerisch	IstNumerisch(Ausdruck) Gibt einen Wert vom Typ <i>Boolean</i> zurück, der angibt, ob ein Ausdruck als eine Zahl ausgewertet werden kann.
TypeName	TypeName(VarName) Gibt einen Wert vom Typ <i>String</i> zurück, der Informationen zu einer Variablen bereitstellt.
VarTyp	VarTyp(VarName) Gibt einen Wert vom Typ <i>Integer</i> zurück, der den Untertyp einer Variablen angibt.

Konvertierung

Funktion	Beschreibung
Asc	Asc(Zeichenfolge) Gibt einen Wert vom Typ <i>Integer</i> zurück, der dem Zeichen-code entspricht, den der erste Buchstabe einer Zeichenfolge hat.
AscW	AscW(Zeichenfolge) Gibt einen Wert vom Typ <i>Integer</i> zurück, der dem Zeichen-code entspricht, den der erste Buchstabe einer Zeichenfolge hat.
ChrW	ChrW(Zeichencode) Gibt einen Wert vom Typ <i>String</i> zurück, der das Zeichen enthält, das dem angegebenen Zeichencode zugeordnet ist.
ChrW\$	ChrW(Zeichencode) Gibt einen Wert vom Typ <i>String</i> zurück, der das Zeichen enthält, das dem angegebenen Zeichencode zugeordnet ist.

Funktion	Beschreibung
CLngPtr	CLngPtr(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>LongPtr</i> um.
DatSeriell	DatSeriell(Jahr;Monat;Tag) Gibt einen Wert vom Typ <i>Datum</i> für eine Tagesangabe (Jahr, Monat und Tag) zurück.
DatWert	DatWert(Datum) Gibt einen Wert vom Typ <i>Datum</i> zurück.
EuroKonvertieren	EuroKonvertieren(Zahl;Quellwährung;Zielwährung; [VolleGenauigkeit;Triangulationsgenauigkeit]) Konvertiert eine Zahl in Euro oder aus Euro in eine teilnehmende Währung.
FormatDatumZeit	FormatDatumZeit(Datum;[BenanntesFormat]) Gibt einen Ausdruck zurück, der als Datum oder Uhrzeit formatiert ist.
FormatProzent	FormatProzent(Ausdruck;[Genauigkeit];[FührendeZiffer]; [KlammernFürNegativzahl];[ZiffernGruppieren]) Gibt einen Ausdruck zurück, der als Prozentsatz (multipliziert mit 100) mit angehängtem %-Zeichen formatiert ist.
FormatWährung	FormatWährung(Ausdruck;[Genauigkeit];[FührendeZiffer]; [KlammernFürNegativzahl];[ZiffernGruppieren]) Gibt einen Ausdruck zurück, der als Währungswert mit dem Währungssymbol formatiert ist, das in der Systemsteuerung von Windows definiert ist.
FormatZahl	FormatZahl(Ausdruck;[Genauigkeit];[FührendeZiffer]; [KlammernFürNegativzahl];[ZiffernGruppieren]) Gibt einen Ausdruck zurück, der als Zahl formatiert ist.
GUIDFromString	GUIDFromString(Zeichenfolge) Wandelt eine Zeichenfolge in eine GUID um, die ein Array vom Typ <i>Byte</i> ist.
Hex	Hex(Zahl) Gibt einen Wert vom Typ <i>String</i> zurück, der dem hexadezimalen Wert einer Zahl entspricht.
Hex\$	Hex\$(Zahl) Gibt einen Wert vom Typ <i>String</i> zurück, der dem hexadezimalen Wert einer Zahl entspricht.
HtmlEncode	HtmlEncode(NurText;[Länge]) Gibt einen HTML-codierten Wert vom Typ <i>String</i> zurück, der die angegebene Gesamtlänge hat.

Funktion	Beschreibung
Jahr	Jahr(Datum) Gibt einen Wert vom Typ <i>Integer</i> zurück, der die ganze Zahl angibt, die dem Jahr entspricht.
Minute	Minute(Zeit) Gibt einen Wert vom Typ <i>Integer</i> zurück, der eine ganze Zahl von 1 bis einschließlich 59 angibt, die einer Minute einer Stunde entspricht.
Monat	Monat(Datum) Gibt einen Wert vom Typ <i>Integer</i> zurück, der eine ganze Zahl von 1 bis einschließlich 12 angibt, die einem Monat eines Jahres entspricht.
NurText	NurText(RichText;[Länge]) Gibt einen Wert vom Typ <i>String</i> zurück, der als Nur-Text-Zeichenfolge formatiert ist, deren Gesamtlänge gleich der angegebenen Anzahl von Zeichen ist.
Nz	Nz(Variante;[WertWennNull]) Gibt 0, eine Zeichenfolge mit der Länge 0 (""), oder einen anderen angegebenen Wert zurück, wenn eine Variable vom Typ <i>VARIANT</i> den Wert NULL hat.
Oktal	Oktal(Zahl) Gibt einen Wert vom Typ <i>String</i> zurück, der dem Oktalwert einer Zahl entspricht.
Oktal\$	Oktal\$(Zahl) Gibt einen Wert vom Typ <i>String</i> zurück, der dem Oktalwert einer Zahl entspricht.
Sekunde	Sekunde(Zeit) Gibt einen Wert vom Typ <i>Integer</i> zurück, der eine ganze Zahl von 1 bis einschließlich 59 angibt, die einer Sekunde einer Minute entspricht.
Str	Str(Zahl) Gibt einen Wert vom Typ <i>String</i> zurück, der der bereitgestellten Zahl entspricht.
Str\$	Str\$(Zahl) Gibt einen Wert vom Typ <i>String</i> zurück, der der bereitgestellten Zahl entspricht.
StringFromGUID	StringFromGUID(Guid) Konvertiert eine GUID, die ein Array mit Werten vom Typ <i>Byte</i> ist, in eine Zeichenfolge.
StrKonv	StrKonv(Zeichenfolge;Konvertierung;[LCID]) Gibt einen Wert vom Typ <i>String</i> zurück, der wie angegeben konvertiert wurde.

Funktion	Beschreibung
Stunde	Stunde(Zeit) Gibt einen Wert vom Typ <i>Integer</i> zurück, der eine ganze Zahl von 1 bis einschließlich 23 angibt, die einer Stunde eines Tages entspricht.
Tag	Tag(Datum) Gibt einen Wert vom Typ <i>Integer</i> zurück, der eine ganze Zahl von 1 bis einschließlich 31 angibt, die einem Tag eines Monats entspricht.
Wert	Wert(Zeichenfolge) Gibt die in einer Zeichenfolge enthaltenen Zahlen als numerischen Wert mit geeignetem Typ zurück.
Wochentag	Wochentag(Datum;[ErsterWochentag]) Gibt einen Wert vom Typ <i>Integer</i> zurück, der die ganze Zahl angibt, die dem Wochentag entspricht.
ZBool	ZBool(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>Boolean</i> um.
ZByte	ZByte(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>Byte</i> um.
Zchn	Zchn(Zeichencode) Gibt einen Wert vom Typ <i>String</i> zurück, der das Zeichen enthält, das dem angegebenen Zeichencode zugeordnet ist.
Zchn\$	Zchn\$(Zeichencode) Gibt einen Wert vom Typ <i>String</i> zurück, der das Zeichen enthält, das dem angegebenen Zeichencode zugeordnet ist.
ZCurrency	ZCurrency(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>Währung</i> um.
ZDate	ZDate(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>Datum</i> um.
ZDouble	ZDouble(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>Double</i> um.
ZeitSeriell	ZeitSeriell(Stunde;Minute;Sekunde) Gibt einen Wert vom Typ <i>Datum</i> zurück, der die Uhrzeit für eine bestimmte Stunde, Minute und Sekunde enthält.
ZeitSeriellStr	ZeitSeriellStr(Zeit) Gibt einen Wert vom Typ <i>Datum</i> zurück, der die Uhrzeit enthält.

Funktion	Beschreibung
ZInteger	ZInteger(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>Integer</i> um.
ZLong	ZInteger(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>Long Integer</i> um.
ZSingle	ZSingle(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>Single</i> um.
ZString	ZString(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>String</i> um.
ZVarDat	ZString(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>Datum</i> um.
ZVariant	ZString(Ausdruck) Wandelt den Wert eines Ausdrucks in einen Wert vom Typ <i>Variant</i> um.

Mathematisch

Funktion	Beschreibung
Abs	Abs(Zahl) Gibt einen Wert zurück, der denselben Typ hat wie der als Eingang übergebene Wert und der den absoluten Wert einer Zahl angibt.
ArcTan	ArcTan(Zahl) Gibt einen Wert vom Typ <i>Double</i> zurück, der den Arkustangens einer Zahl angibt.
Cos	Cos(Zahl) Gibt einen Wert vom Typ <i>Double</i> zurück, der den Kosinus eines Winkels angibt.
Exponential	Exponential(Zahl) Gibt einen Wert vom Typ <i>Double</i> zurück, der e (die Basis natürlicher Logarithmen) hoch Exponent angibt.
Fix	Fix(Zahl) Gibt den ganzzahligen Teil einer Zahl zurück.
Int	Int(Zahl) Gibt den ganzzahligen Teil einer Zahl zurück.

Funktion	Beschreibung
Logarithmus	Logarithmus(Zahl) Gibt einen Wert vom Typ <i>Double</i> zurück, der den natürlichen Logarithmus einer Zahl angibt.
QWurzel	QWurzel(Zahl) Gibt einen Wert vom Typ <i>Double</i> zurück, der die Quadratwurzel einer positiven Zahl angibt.
Runden	Runden(Zahl;[Genauigkeit]) Gibt eine Zahl zurück, die auf eine bestimmte Anzahl von Dezimalstellen gerundet wurde.
Sin	Sin(Zahl) Gibt einen Wert vom Typ <i>Double</i> zurück, der den Sinus eines Winkels angibt.
Tan	Tan(Zahl) Gibt einen Wert vom Typ <i>Double</i> zurück, der den Tangens eines Winkels angibt.
Vorzchn	Vorzchn(Zahl) Gibt einen Wert vom Typ <i>Integer</i> zurück, der das Vorzeichen einer Zahl angibt.
ZZG	ZZG(Zahl) Gibt einen Wert vom Typ <i>Single</i> zurück, der eine Zufallszahl enthält.

Meldungen

Funktion	Beschreibung
Eingabefeld	Eingabefeld(Eingabeaufforderung;[Titel];[Standard]; [XPosition];[YPosition];[Hilfdatei];[Kontext]) Zeigt eine Eingabeaufforderung in einem Dialogfeld an, wartet darauf, dass der Benutzer Text eingibt oder auf eine Schaltfläche klickt, und gibt einen Wert vom Typ <i>String</i> zurück, der den Inhalt des Textfelds enthält.
Eingabefeld\$	Eingabefeld\$(Eingabeaufforderung;[Titel];[Standard]; [XPosition];[YPosition];[Hilfdatei];[Kontext]) Zeigt eine Eingabeaufforderung in einem Dialogfeld an, wartet darauf, dass der Benutzer Text eingibt oder auf eine Schaltfläche klickt, und gibt einen Wert vom Typ <i>String</i> zurück, der den Inhalt des Textfelds enthält.

Funktion	Beschreibung
Meldung	<p>Meldung(Eingabeaufforderung;[Schaltflächen];[Titel];[Hilfdatei];[Kontext])</p> <p>Zeigt eine Meldung in einem Dialogfeld an, wartet, bis der Benutzer auf eine Schaltfläche geklickt hat, und gibt einen Wert vom Typ <i>Integer</i> zurück, der angibt, auf welche Schaltfläche der Benutzer geklickt hat.</p>

Programmablauf

Funktion	Beschreibung
Schalter	<p>Schalter(Ausdruck-1;Wert-1;[Ausdruck-2];[Wert-2];...;[Ausdruck-n];[Wert-n])</p> <p>Wertet eine Liste mit Ausdrücken aus und gibt einen Wert vom Typ <i>Variant</i> oder einen Ausdruck zurück, der dem ersten Ausdruck in der Liste zugeordnet ist, der wahr ist.</p>
Wahl	<p>Wahl(Index;Auswahl-1;[Auswahl-2];...;[Auswahl-n])</p> <p>Wählt in einer Argumentenliste einen Wert aus und gibt diesen zurück.</p>
Wenn	<p>Wenn(Ausdruck;True-Teil;False-Teil)</p> <p>Gibt abhängig von der Auswertung eines Ausdrucks einen von zwei Teilen zurück.</p>

SQL-Aggregat

Funktion	Beschreibung
Anzahl	<p>Anzahl(Ausdruck)</p> <p>Berechnet die Anzahl von Datensätzen, die eine Abfrage zurückgegeben hat.</p>
Max	<p>Max(Ausdruck)</p> <p>Gibt den größten Wert aus einer Menge von Werten zurück, die in einem angegebenen Feld für eine Abfrage enthalten sind.</p>
Min	<p>Min(Ausdruck)</p> <p>Gibt den kleinsten Wert aus einer Menge von Werten zurück, die in einem angegebenen Feld für eine Abfrage enthalten sind.</p>
Mittelwert	<p>Mittelwert(Ausdruck)</p> <p>Berechnet das arithmetische Mittel mehrerer Werte, die im angegebenen Datenfeld einer Abfrage enthalten sind.</p>

Funktion	Beschreibung
StAbw	StAbw(Ausdruck) Gibt die Standardabweichung einer Stichprobe zurück, die aus einer Menge von Werten besteht, die in einem angegebenen Datenfeld für eine Abfrage enthalten sind.
StAbwn	StAbwn(Ausdruck) Gibt die Standardabweichung einer Grundgesamtheit zurück, die aus einer Menge von Werten besteht, die in einem angegebenen Datenfeld für eine Abfrage enthalten sind.
Summe	Summe(Ausdruck) Gibt die Summe von einer Menge von Werten zurück, die in einem angegebenen Datenfeld für eine Abfrage enthalten sind.
Varianz	Varianz(Ausdruck) Gibt die Varianz einer Stichprobe zurück, die aus einer Menge von Werten besteht, die in einem angegebenen Datenfeld für eine Abfrage enthalten sind.
Varianzen	Varianzen(Ausdruck) Gibt die Varianz einer Stichprobe zurück, die aus einer Menge von Werten besteht, die in einem angegebenen Datenfeld für eine Abfrage enthalten sind.

Text

Funktion	Beschreibung
Asc	Asc(Zeichenfolge) Gibt einen Wert vom Typ <i>Integer</i> zurück, der dem Zeichencode entspricht, den der erste Buchstabe einer Zeichenfolge hat.
AscW	AscW(Zeichenfolge) Gibt einen Wert vom Typ <i>Integer</i> zurück, der dem Zeichencode entspricht, den der erste Buchstabe einer Zeichenfolge hat.
ChrW	ChrW(Zeichencode) Gibt einen Wert vom Typ <i>String</i> zurück, der das Zeichen enthält, das dem angegebenen Zeichencode zugeordnet ist.
ChrW\$	ChrW(Zeichencode) Gibt einen Wert vom Typ <i>String</i> zurück, der das Zeichen enthält, das dem angegebenen Zeichencode zugeordnet ist.

Funktion	Beschreibung
Ersetzen	Ersetzen(Zeichenfolge;Suchen;ErsetzenDurch;[Startwert]; [Anzahl];[Zeichenvergleich]) Gibt einen Wert vom Typ <i>String</i> zurück, in dem eine bestimmte Teilzeichenfolge so oft wie angegeben durch eine andere Teilzeichenfolge ersetzt wurde.
ErstellenKriterien	ErstellenKriterien(Feld;Feldtyp;Ausdruck) Gibt einen Wert vom Typ <i>String</i> zurück, der den Filter angibt, der in einer Abfrage für das angegebene Datenfeld und die angegebenen Kriterien verwendet wird.
Format	Format(Ausdruck;[Format];[ErsterWochentag]; [ErsteWocheDesJahres]) Gibt einen Wert vom Typ <i>String</i> zurück, der einen Ausdruck enthält, der entsprechend den Anweisungen formatiert ist, die in einem Formatausdruck stehen.
Format\$	Format\$(Ausdruck;[Format];[ErsterWochentag]; [ErsteWocheDesJahres]) Gibt einen Wert vom Typ <i>String</i> zurück, der einen Ausdruck enthält, der entsprechend den Anweisungen formatiert ist, die in einem Formatausdruck stehen.
Glätten	Glätten(Zeichenfolge) Gibt einen Wert vom Typ <i>String</i> zurück, der eine Kopie einer angegebenen Zeichenfolge ohne führende und nachfolgende Leerzeichen enthält.
Glätten\$	Glätten\$(Zeichenfolge) Gibt einen Wert vom Typ <i>String</i> zurück, der eine Kopie einer angegebenen Zeichenfolge ohne führende und nachfolgende Leerzeichen enthält.
Großbst	Großbst(Zeichenfolge) Gibt einen Wert vom Typ <i>String</i> zurück, der die in Großbuchstaben konvertierte angegebene Zeichenfolge enthält.
Großbst\$	Großbst\$(Zeichenfolge) Gibt einen Wert vom Typ <i>String</i> zurück, der die in Großbuchstaben konvertierte angegebene Zeichenfolge enthält.
GUIDFromString	GUIDFromString(Zeichenfolge) Wandelt eine Zeichenfolge in eine GUID um, die ein Array vom Typ <i>Byte</i> ist.
HtmlEncode	HtmlEncode(NurText;[Länge]) Gibt einen HTML-codierten Wert vom Typ <i>String</i> zurück, der die angegebene Gesamtlänge hat.

Funktion	Beschreibung
InStr	InStr(Startwert;Zeichenfolge;SuchenNach;[Zeichenvergleich]) Gibt einen Wert vom Typ <i>String</i> zurück, der die Position angibt, an der eine Zeichenfolge das erste Mal in einer anderen Zeichenfolge vorkommt.
InStrRev	InStrRev(Zeichenfolge;SuchenNach;[Startwert]; [Zeichenvergleich]) Gibt die Position zurück, an der eine Zeichenfolge in einer anderen Zeichenfolge, bezogen auf deren Ende, enthalten ist.
Kleinbst	Kleinbst(Zeichenfolge) Gibt einen Wert vom Typ <i>String</i> zurück, der die in Kleinbuchstaben konvertierte angegebene Zeichenfolge enthält.
Kleinbst\$	Kleinbst\$(Zeichenfolge) Gibt einen Wert vom Typ <i>String</i> zurück, der die in Kleinbuchstaben konvertierte angegebene Zeichenfolge enthält.
Länge	Länge(Zeichenfolge) Gibt einen Wert vom Typ <i>Long Integer</i> zurück, der angibt, wie viele Zeichen eine Zeichenfolge enthält oder wie viele Bytes erforderlich sind, eine Variable zu speichern.
Leerchn	Leerchn(Zahl) Gibt einen Wert vom Typ <i>String</i> zurück, der aus der angegebenen Anzahl von Leerzeichen besteht.
Leerchn\$	Leerchn\$(Zahl) Gibt einen Wert vom Typ <i>String</i> zurück, der aus der angegebenen Anzahl von Leerzeichen besteht.
LGlätten	LGlätten(Zeichenfolge) Gibt einen Wert vom Typ <i>String</i> zurück, der eine Kopie einer angegebenen Zeichenfolge ohne führende Leerzeichen enthält.
LGlätten\$	LGlätten\$(Zeichenfolge) Gibt einen Wert vom Typ <i>String</i> zurück, der eine Kopie einer angegebenen Zeichenfolge ohne führende Leerzeichen enthält.
Links	Links(Zeichenfolge;Länge) Gibt einen Wert vom Typ <i>String</i> zurück, der eine angegebene Anzahl von Zeichen von der linken Seite einer Zeichenfolge enthält.
Links\$	Links\$(Zeichenfolge;Länge) Gibt einen Wert vom Typ <i>String</i> zurück, der eine angegebene Anzahl von Zeichen von der linken Seite einer Zeichenfolge enthält.

Funktion	Beschreibung
NurText	NurText(RichText;Länge) Gibt einen Wert vom Typ <i>String</i> zurück, der als Nur-Text-Zeichenfolge formatiert ist, deren Gesamtlänge gleich der angegebenen Anzahl von Zeichen ist.
Rechts	Rechts(Zeichenfolge;Länge) Gibt einen Wert vom Typ <i>String</i> zurück, der eine angegebene Anzahl von Zeichen von der rechten Seite einer Zeichenfolge enthält.
Rechts\$	Rechts\$(Zeichenfolge;Länge) Gibt einen Wert vom Typ <i>String</i> zurück, der eine angegebene Anzahl von Zeichen von der rechten Seite einer Zeichenfolge enthält.
RGlätten	RGlätten(Zeichenfolge) Gibt einen Wert vom Typ <i>String</i> zurück, der eine Kopie einer angegebenen Zeichenfolge ohne nachfolgende Leerzeichen enthält.
RGlätten\$	RGlätten\$(Zeichenfolge) Gibt einen Wert vom Typ <i>String</i> zurück, der eine Kopie einer angegebenen Zeichenfolge ohne nachfolgende Leerzeichen enthält.
String	String(Anzahl;Zeichen) Gibt einen Wert vom Typ <i>String</i> zurück, der eine wiederholte Zeichenfolge der angegebenen Länge hat.
String\$	String\$(Anzahl;Zeichen) Gibt einen Wert vom Typ <i>String</i> zurück, der eine wiederholte Zeichenfolge der angegebenen Länge hat.
StringFromGUID	StringFromGUID(Guid) Konvertiert eine GUID, die ein Array mit Werten vom Typ <i>Byte</i> ist, in eine Zeichenfolge.
StrKonv	StrKonv(Zeichenfolge;Konvertierung;[LCID]) Gibt einen Wert vom Typ <i>String</i> zurück, der wie angegeben konvertiert wurde.
StrReverse	StrReverse(Zeichenfolge) Gibt einen Wert vom Typ <i>String</i> zurück, in dem die Zeichenreihenfolge der angegebenen Zeichenfolge umgekehrt wurde.
StrVgl	StrVgl(Zeichenfolge1;Zeichenfolge2;[Zeichenvergleich]) Gibt einen Wert vom Typ <i>Integer</i> zurück, der das Ergebnis des Vergleichs zweier Zeichenfolgen angibt.
Teil	Teil(Zeichenfolge;Startwert;[Länge]) Gibt einen Wert vom Typ <i>String</i> zurück, der eine angegebene Anzahl von Zeichen einer Zeichenfolge enthält.

Funktion	Beschreibung
Teil\$	Teil(Zeichenfolge;Startwert;[Länge]) Gibt einen Wert vom Typ <i>String</i> zurück, der eine angegebene Anzahl von Zeichen einer Zeichenfolge enthält.
Zchn	Zchn(Zeichencode) Gibt einen Wert vom Typ <i>String</i> zurück, der das Zeichen enthält, das dem angegebenen Zeichencode zugeordnet ist.
Zchn\$	Zchn\$(Zeichencode) Gibt einen Wert vom Typ <i>String</i> zurück, der das Zeichen enthält, das dem angegebenen Zeichencode zugeordnet ist.

Funktionen mit und ohne Dollarzeichen

Wenn Sie sich die Funktionen genauer angeschaut haben, werden Ihnen sicherlich einige Funktionen aufgefallen sein, wo der Funktionsname mal ohne und mal mit einem Dollarzeichen (\$) geschrieben wird (z.B. **Glätten**, **Glätten\$**, **Links**, **Links\$**, **Teil**, **Teil\$**, usw.). Sie werden sich auch sicherlich gefragt haben, worin besteht der Unterschied? Hat es eine Bedeutung, wenn der Funktionsname mal ohne und mal mit dem Dollarzeichen geschrieben wird? Die Antwort ist: Nein, es gibt keinen wirklichen Unterschied. Das Dollarzeichen deutet lediglich darauf hin, dass das Ergebnis vom Typ *String*, also Text ist. Die Dollarangabe ist noch aus der Zeit der Programmiersprache Basic übriggeblieben. Es spielt also keine Rolle, ob Sie die Funktionen ohne oder mit Dollarzeichen verwenden.

Fachbegriffe

In den Kurzbeschreibungen der aufgelisteten Funktionen werden auch immer wieder mal Fachbegriffe verwendet, die Ihnen vermutlich nicht bekannt sind. In der nachfolgenden Tabelle bekommen Sie eine Beschreibung der im Skript verwendeten Fachbegriffe.

Fachbegriff	Beschreibung
Byte	Ganzzahlen im Bereich von 0 ... 255
Integer	Ganzzahlen im Bereich von -32.768 ... 32.767
Long Integer	Ganzzahlen im Bereich von -2.147.483.648 ... 2.147.483.647
LongPtr	Ganzzahlen, die in 32-Bit-Systemen dem Datentyp Long Integer entsprechen; in 64-Bit-Systemen liegen sie im Bereich von -9.223.372.036.854.775.808 ... 9.223.372.036.854.775.807
Single	Gleitkommazahlen im Bereich von -3,4*1038 ... 3,4*1038
Double	Gleitkommazahlen im Bereich von -1,797*10308 ... 1,797*10308
String	Zeichenfolge von beliebigen Zeichen
Boolean	Wahrheitswerte (Wahr bzw. Falsch)

Fachbegriff	Beschreibung
Variant	Datentyp für alle Variablen, die nicht explizit als anderer Datentyp deklariert werden. Eine Variable vom Datentyp Variant kann auch die Sonderwerte Empty (leer), Error (Fehler), Nothing (nichts) und NULL enthalten.
Nur-Client-Anwendung	Lokaler Einsatz von Access in einer nicht webbasierten Umgebung.
RGB	Es handelt sich um einen Farbraum, in dem die Farben aus den Grundfarben Rot , Grün und Blau zusammengemischt werden.
URL	Uniform Resource Locator ; identifiziert und lokalisiert eine Ressource, beispielsweise eine Webseite im Internet, über die zu verwendende Zugriffsmethode (z.B. das verwendete Netzwerkprotokoll) und den Ort der Ressource in Computernetzwerken. Aus: www.wikipedia.org
HTML	HyperText Markup Language ; ist eine textbasierte Auszeichnungssprache zur Strukturierung elektronischer Dokumente wie Texte mit Hyperlinks, Bildern und anderen Inhalten. HTML-Dokumente sind die Grundlage des World Wide Web und werden von Webbrowsern dargestellt. Aus: www.wikipedia.org
Memo	Felddatentyp älterer Access-Versionen. Entspricht in Access für Microsoft 365 dem Felddatentyp Langer Text .
NULL	Diese Angabe hat nichts mit der Ziffer Null zu tun, sondern damit wird der leere Inhalt von Datenfeldern bezeichnet ¹ . Der Begriff NULL kann im Grunde auch mit <i>nicht definiert</i> beschrieben werden. NULL ist streng genommen kein Wert.

¹ In Datenfeldern vom Felddatentyp **Kurzer Text** bzw. **Langer Text** gibt es noch zwischen NULL und der leeren Zeichenfolge einen Unterschied. Eine leere Zeichenfolge ist aus Sicht von Datenbanksystemen eine Zeichenfolge, auch wenn Sie keine Zeichen enthält. NULL bedeutet dagegen: nicht definiert.