

Spis treści

Opis wyrażeń.....	2
Przykłady wyrażeń używanych w formularzach i raportach	2
Tworzenie kontrolki obliczeniowej	2
Wyrażenia łączące lub zmieniające tekst.....	2
Wyrażenia w nagłówkach i stopkach	3
Wyrażenia wykonujące operacje arytmetyczne	3
Wyrażenia liczące, sumujące i uśredniające wartości	5
Wyrażenia liczące, sumujące i wyszukujące wartości selektywnie przy użyciu funkcji agregujących dziedziny	5
Wyrażenia zmieniające i obliczające daty	6
Wyrażenia warunkowe zwracające jedną z dwóch możliwych wartości	6
Przykłady wyrażeń używanych w zapytaniach i filtrach.....	6
Wyrażenia zmieniające tekst w zapytaniu lub filtrze	7
Wyrażenia wykonujące operacje arytmetyczne w polach obliczeniowych	7
Wyrażenia zmieniające daty i wykonujące obliczenia na datach w polach obliczeniowych	8
Wyrażenia liczące, sumujące i uśredniające wartości przy użyciu funkcji agregujących SQL lub funkcji agregujących dziedziny	8
Wyrażenia działające z polami bez informacji (pola o wartości null)	9
Wyrażenie używające podzapytania w celu utworzenia pola obliczeniowego	9
Wyrażenia definiujące kryteria i ograniczające liczbę rekordów w zestawie wyników.....	9
Wyrażenia dopasowujące cały tekst lub jego część.....	10
Wyrażenia używające dat w kryteriach dopasowywania	10
Wyrażenia dopasowujące brakującą wartość (null) lub ciąg o zerowej długości	11
Wyrażenia używające wzorców w celu dopasowania rekordów	11
Wyrażenia dopasowujące wiersze na podstawie wyników funkcji agregującej dziedziny	12
Wyrażenia dopasowujące na podstawie wyników podzapytań	12
Wyrażenia używane w zapytaniach aktualizujących.....	12
Wyrażenia używane w instrukcjach SQL	13
Przykłady wyrażeń wartości domyślnych.....	13
Dodawanie wartości domyślnej pola tabeli	13
Przykłady wyrażeń reguł sprawdzania poprawności dla pola.....	14
Dodawanie reguły sprawdzania poprawności dla pola.....	14
Przykłady wyrażeń w warunkach makr	15
Dodawanie warunku dla akcji makra	15

W tym artykule przedstawiono przykłady wyrażeń w programie Access. Wyrażenie to kombinacja operatorów matematycznych i logicznych, stałych, funkcji, pól tabel, kontrolki i właściwości, których wynikiem jest pojedyncza wartość. Wyrażenia w programie Access mogą służyć do obliczania wartości, sprawdzania poprawności danych i ustawiania wartości domyślnej dla pola lub kontrolki.

Uwaga : Mimo że ten artykuł zawiera podstawowe wskazówki dotyczące tworzenia wyrażeń, nie jest wyczerpującym przewodnikiem dotyczącym korzystania z narzędzi do tworzenia wyrażeń dostępnych w programie Access. Aby uzyskać więcej informacji o tworzeniu wyrażeń, zobacz artykuł Tworzenie wyrażeń.

Opis wyrażeń

W programie Access termin wyrażenie jest równoznaczny z terminem formuła. Wyrażenie składa się z pewnej liczby możliwych elementów, których można używać pojedynczo lub łącznie w celu uzyskaniażądanego wyniku. Do tych elementów należą:

Identyfikatory — nazwy pól tabeli, kontrolki w formularzach lub raportach albo właściwości tych pól lub kontrollek.

Operatory — na przykład + (plus) lub - (minus).

Funkcje — takie jak SUM lub AVG.

Stałe — wartości, które się nie zmieniają — na przykład ciągi tekstu lub liczby, które nie są obliczane przez wyrażenie.

Wyrażeń można używać na kilka sposobów, na przykład, aby wykonać obliczenie, pobrać wartość z kontrolki lub dostarczyć kryterium do zapytania.


Aby uzyskać więcej informacji na temat sposobów używania wyrażeń i sytuacji, w których są one używane, zobacz artykuł Tworzenie wyrażeń.

Przykłady wyrażeń używanych w formularzach i raportach

Tabele w tej sekcji zawierają przykłady wyrażeń służących do obliczania wartości w kontrolce umieszczonej w formularzu lub raporcie. Aby utworzyć kontrolkę obliczeniową, należy wprowadzić wyrażenie we właściwości kontrolki ŹródłoKontrolki zamiast w polu tabeli lub zapytaniu.

W kolejnych krokach omówiono sposób wprowadzania wyrażenia w kontrolce pola tekstowego w istniejącym formularzu lub raporcie.

Tworzenie kontrolki obliczeniowej

W okienku nawigacji kliknij prawym przyciskiem myszy formularz lub raport, który chcesz zmienić, a następnie w menu skrótów kliknij pozycję Widok projektu .

W formularzu lub raporcie kliknij prawym przyciskiem myszy kontrolkę pola tekstowego, którą chcesz zmienić (nie etykietę skojarzoną z tym polem tekstowym), a następnie kliknij w menu skrótów polecenie Właściwości.

W razie potrzeby kliknij kartę Wszystkie lub Dane. Obie karty zawierają właściwość ŹródłoKontrolki.

Kliknij pole obok właściwości ŹródłoKontrolki i wpisz wyrażenie. Możesz na przykład skopiować i wkleić wyrażenie z kolumny Wyrażenie tabeli znajdującej się w poniższej sekcji.

Zamknij arkusz właściwości.

Wyrażenia łączące lub zmieniające tekst

W wyrażeniach w poniższej tabeli użyto operatorów & i + w celu połączenia ciągów tekstowych, wbudowanych funkcji do zmieniania ciągu tekstowego lub innego przetworzenia tekstu, aby utworzyć kontrolkę obliczeniową.

Wyrażenie	Wynik
= "NIE DOTYCZY"	Wyświetla wyrazy „nie dotyczy”.
= [Imię] & " " & [Nazwisko]	Wyświetla wartości znajdujące się w polach tabeli o nazwach Imię i Nazwisko. W tym przykładzie operator & służy do połączenia pola Imię, znaku spacji (ujętego w cudzysłowy) i pola Nazwisko.
= Left([NazwaProduktu], 1)	Używa funkcji Left w celu wyświetlenia pierwszego znaku wartości pola lub kontrolki o nazwie NazwaProduktu.
= Right([KodSprzętu], 2)	Używa funkcji Right w celu wyświetlenia dwóch ostatnich znaków wartości pola lub kontrolki o nazwie KodSprzętu.

Wyrażenie	Wynik
=Trim([Adres])	Używa funkcji Trim w celu wyświetlenia wartości kontrolki Adres i usunięcia wszelkich spacji wiodących i końcowych.
=If(IsNull([Region]),[Miasto]&" "&[KodPocztowy],[Miasto]&" "&[Region]&" "&[KodPocztowy])	Używa funkcji If w celu wyświetlenia wartości kontrolki Miasto i KodPocztowy, jeśli wartość kontrolki Region jest równa null. W przeciwnym razie wyświetlane są wartości kontrolki Miasto, Region i KodPocztowy oddzielone spacjami.
=[Miasto] & (" " + [Region]) & " " & [KodPocztowy]	Używa operatora + i funkcji propagacji wartości pustych w celu wyświetlenia wartości kontrolki Miasto i KodPocztowy, jeśli wartość pola lub kontrolki Region jest równa null. W przeciwnym razie wyświetla wartości pól lub kontrolki Miasto, Region i KodPocztowy oddzielone spacjami. Funkcja propagacji wartości pustych oznacza, że jeśli dowolny składnik wyrażenia jest pusty, całe wyrażenie również jest puste. Operator + obsługuje funkcję propagacji wartości pustych, natomiast operator & nie obsługuje jej.

Wyrażenia w nagłówkach i stopkach

Do wyświetlania lub drukowania numerów stron w formularzach lub raportach służą właściwości Page i Pages. Właściwości Page i Pages są dostępne tylko podczas drukowania lub wyświetlania podglądu, więc nie są wyświetlane w arkuszu właściwości formularza ani raportu. Zazwyczaj te właściwości są umieszczane w polu tekstowym w sekcji nagłówek lub stopka formularza lub raportu i używane w formie wyrażenia, takiego jak te przedstawione w poniższej tabeli.

Aby uzyskać więcej informacji na temat używania nagłówków i stopek w formularzach oraz raportach, zobacz artykuł Wstawianie numerów stron do formularza lub raportu.

Wyrażenie	Przykład wyniku
=[Page]	1
"Strona " & [Page]	Strona 1
"Strona " & [Page] & " z " & [Pages]	Strona 1 z 3
[Page] & " z " & [Pages] & " stron"	1 z 3 stron
[Page] & "/" & [Pages] & " stron"	1/3 stron
[Kraj/region] & " - " & [Page]	UK - 1
=Format([Page], "000")	001
"Wydrukowano w dniu:" & Date()	Wydrukowano w dniu: 31-12-2007

Wyrażenia wykonujące operacje arytmetyczne

Za pomocą wyrażeń można dodawać, odejmować, mnożyć, dzielić wartości z dwóch lub większej liczby pól bądź kontroltek. Można również przy ich użyciu wykonywać operacje arytmetyczne na datach. Załóżmy na przykład, że istnieje pole tabeli typu Data/godzina o nazwie DataWymagana. W polu lub kontrolce powiązanej z polem wyrażenie =[DataWymagana] - 2 zwraca wartość typu Data/godzina oznaczającą datę o dwa dni wcześniej niż wartość pola DataWymagana.

Wyrażenie	Wynik
= [SumaCzęściowa] + [Fracht]	Suma wartości pól lub kontroltek SumaCzęściowa i Fracht.
=[Data_dostawy]-[Data_wysyłki]	Różnica między wartościami daty pól lub kontroltek DataDostawy i DataWysyłki.
=[Cena] * 1,06	Iloczyn wartości pola lub kontrolki Cena i liczby 1,06 (dodaje 6 procent do wartości Cena).

Wyrażenie	Wynik
=[Ilość] * [Cena]	Iloczyn wartości pól lub kontroltek Ilość i Cena.
=[SumaDlaPracownika]/[SumaDlaKrajuRegionu]	Iloraz wartości pól lub kontroltek SumaDlaPracownika i SumaDlaKrajuRegionu.

Uwaga : Jeśli w wyrażeniu używany jest operator arytmetyczny (+, -, * lub /) a jedna z kontroltek w wyrażeniu ma wartość null, to wynikiem całego wyrażenia również będzie wartość null — tak działa funkcja propagacji wartości pustych. Jeśli dowolny rekord w kontrolkach używanych w wyrażeniu może mieć wartość null, propagacji wartości pustych można uniknąć dzięki przekonwertowaniu wartości null na zero, korzystając z funkcji Nz — na przykład =Nz([SumaCzęściowa])+Nz([Fracht]).

Aby uzyskać więcej informacji na temat tej funkcji, zobacz artykuł Funkcja Nz.

Wyrażenia odwołujące się do wartości z innych pól lub kontroltek

Czasami potrzebna jest wartość istniejąca w innym miejscu, na przykład w polu lub kontrolce w innym formularzu albo raporcie. Aby zwrócić wartość z innego pola lub innej kontrolki, można użyć wyrażenia.

W tabeli podano przykłady wyrażen, które można stosować w kontrolkach obliczeniowych formularzy.

Wyrażenie	Wynik
=Forms![Zamówienia]![IdentyfikatorZamówienia]	Wartość kontrolki IdentyfikatorZamówienia w formularzu Zamówienia.
=Forms![Zamówienia]![Zamówienia1].Form![SumaCzęściowaZamówienia].	Wartość kontrolki SumaCzęściowaZamówienia w podformularzu o nazwie Zamówienia1 formularza Zamówienia.
=Forms![Zamówienia]![Zamówienia1]![IdentyfikatorProduktu].Column(2)	Wartość trzeciej kolumny w kontrolce IdentyfikatorProduktu, wielokolumnowym polu listy w podformularzu o nazwie Zamówienia1, zawartym w formularzu Zamówienia. (Należy zwrócić uwagę, że 0 odnosi się do pierwszej kolumny, 1 do drugiej kolumny itd.).
=Forms![Zamówienia]![Zamówienia1]![Cena] * 1,06	Iloczyn liczby 1,06 i wartości kontrolki Cena w podformularzu o nazwie Zamówienia1 formularza Zamówienia (wartość kontrolki Cena jest zwiększana o 6 procent).
=Parent![IdentyfikatorZamówienia]	Wartość kontrolki IdentyfikatorZamówienia w formularzu głównym lub nadrzędnym bieżącego podformularza.

Wyrażenia w poniższej tabeli przedstawiają niektóre sposoby używania kontroltek obliczeniowych w raportach. Wyrażenia odwołują się do właściwości Report.

Aby uzyskać więcej informacji na temat tej właściwości, zobacz artykuł Właściwość Report.

Wyrażenie	Wynik
=Report![Faktura]![IdentyfikatorZamówienia]	Wartość kontrolki o nazwie „IdentyfikatorZamówienia” w raporcie o nazwie „Faktura”.
=Report![Podsumowanie]![RaportPodsumowujący]![PodsumowanieSprzedaży]	Wartość kontrolki PodsumowanieSprzedaży w podraporcie o nazwie RaportPodsumowujący w raporcie Podsumowanie.

Wyrażenie	Wynik
=Parent![IdentyfikatorZamówienia]	Wartość kontrolki IdentyfikatorZamówienia w raporcie głównym lub nadrzędnym bieżącego podraportu.

Wyrażenia liczące, sumujące i uśredniające wartości

Aby obliczyć wartości pól lub kontrollek można użyć funkcji agregującej. Można na przykład obliczyć sumę wartości grupy dla stopki grupy w raporcie lub sumę częściową zamówienia dla pozycji formularza. Można również policzyć pozycje w polu lub polach albo obliczyć wartość średnią.

Wyrażenia w poniższej tabeli pokazują niektóre sposoby użycia funkcji takich jak Avg, Count i Sum.

Wyrażenie	Opis
=Avg([Fracht])	Używa funkcji Avg w celu wyświetlenia wartości średniej pola tabeli lub kontrolki o nazwie „Fracht”.
=Count([IdentyfikatorZamówienia])	Używa funkcji Count w celu wyświetlenia liczby rekordów w kontrolce IdentyfikatorZamówienia.
=Sum([Sprzedaż])	Używa funkcji Sum w celu wyświetlenia sumy wartości w kontrolce Sprzedaż.
=Sum([Ilość] * [Cena])	Używa funkcji Sum w celu wyświetlenia sumy iloczynów wartości kontrollek Ilość i Cena.
=([Sprzedaż]/Sum([Sprzedaż])*100	Wyświetla obrót procentowy wyrażony stosunkiem wartości kontrolki Sprzedaż do sumy wszystkich wartości kontrolki Sprzedaż. Uwaga : Jeśli właściwość Format kontrolki jest ustawiona na Procent, w wyrażeniu nie należy umieszczać elementu *100.

Aby uzyskać więcej informacji na temat używania funkcji agregujących i sumowania wartości pól lub kolumn, zobacz artykuły Sumowanie danych przy użyciu zapytania, Zliczanie danych przy użyciu zapytania, Zliczanie wierszy w arkuszu danych i Wyświetlanie sum kolumn w arkuszu danych.

Wyrażenia liczące, sumujące i wyszukujące wartości selektywnie przy użyciu funkcji agregujących dziedziny

Aby selektywnie zsumować lub policzyć wartości, należy skorzystać z funkcji nazywanej funkcją agregującą dziedziny. „Dziedzina” składa się z co najmniej jednego pola w co najmniej jednej tabeli albo co najmniej jednej kontrolki w co najmniej jednym formularzu lub raporcie. Można na przykład dopasować wartości z pola tabeli do wartości w kontrolce formularza.

Wyrażenie	Opis
=DLookup("[Przedstawiciel]", "[Dostawcy]", "[IdentyfikatorDostawcy] = " & Forms("Dostawcy")("[IdentyfikatorDostawcy"]))	Używa funkcji DLookup w celu zwrócenia wartości pola Przedstawiciel w tabeli Dostawcy dla rekordu o wartości pola IdentyfikatorDostawcy równej wartości kontrolki IdentyfikatorDostawcy w formularzu Dostawcy.
=DLookup("[Przedstawiciel]", "[Dostawcy]", "[IdentyfikatorDostawcy] = " & Forms([Nowi dostawcy]![IdentyfikatorDostawcy])	Używa funkcji DLookup w celu zwrócenia wartości pola Przedstawiciel w tabeli Dostawcy dla rekordu o wartości pola IdentyfikatorDostawcy równej wartości kontrolki IdentyfikatorDostawcy w formularzu Nowi dostawcy.
=DSum("[WartośćZamówienia]", "[Zamówienia]", "[IdentyfikatorKlienta] = 'RATTC'")	Używa funkcji DSum w celu zwrócenia sumy wartości w polu WartośćZamówienia w tabeli Zamówienia dla rekordów, w których pole IdentyfikatorKlienta ma wartość RATTC.
=DCount("[Wycofany]", "[Majątek]", "[Wycofany]=Tak")	Używa funkcji DCount w celu zwrócenia liczby wystąpień wartości Tak w polu Wycofany (pole typu Tak/Nie) w tabeli Majątek.

Wyrażenia zmieniające i obliczające daty

Śledzenie dat i godzin jest podstawowym działaniem w bazie danych. Można na przykład obliczyć, ile dni upłynęło od czasu wystawienia faktury w celu wickowania należności. Daty i godziny można formatować na wiele sposobów, jak pokazano w poniższej tabeli

Wyrażenie	Opis
=Date()	Używa funkcji Date w celu wyświetlenia bieżącej daty w formacie mm-dd-rr, gdzie mm to miesiąc (1-12), dd to dzień (1-31), a rr to dwie ostatnie cyfry roku (1980-2099).
=Format(Now(), "tt")	Używa funkcji Format w celu wyświetlenia numeru tygodnia roku dla bieżącej daty, gdzie tt należy do przedziału 1–53.
=DatePart("rrrr", [DataZamówienia])	Używa funkcji DatePart w celu wyświetlenia roku w formacie czterocyfrowym dla wartości kontrolki DataZamówienia.
=DateAdd("r", -10, [OstatecznaDataDostawy])	Używa funkcji DateAdd w celu wyświetlenia daty wcześniejszej o 10 dni od wartości kontrolki OstatecznaDataDostawy.
=DateDiff("d", [DataZamówienia], [DataWysyłki])	Używa funkcji DateDiff w celu wyświetlenia liczby dni między wartościami kontrolki DataZamówienia i DataWysyłki.
=[DataFaktury] + 30	Używa operacji arytmetycznych na datach w celu obliczenia daty przypadającej 30 dni po dacie w polu lub kontrolce DataFaktury.

Wyrażenia warunkowe zwracające jedną z dwóch możliwych wartości

Przykładowe wyrażenia w poniższej tabeli używają funkcji Iif w celu zwrócenia jednej z dwóch możliwych wartości. Do funkcji Iif zostają przekazane trzy argumenty: Pierwszym argumentem jest wyrażenie zwracające wartość Prawda lub Fałsz. Drugim argumentem jest wartość zwracana w przypadku spełnienia warunku wyrażenia. Trzecim — wartość zwracana w przypadku niespełnienia warunku wyrażenia.

Wyrażenie	Opis
=Iif([Potwierdzono] = "Tak", "Zamówienie potwierdzono", "Zamówienia nie potwierdzono")	Używa funkcji Iif (Immediate If) w celu wyświetlenia komunikatu „Zamówienie potwierdzono”, jeśli wartość kontrolki Potwierdzono ma wartość Tak. W przeciwnym razie wyświetla komunikat „Zamówienia nie potwierdzono”.
=Iif(IsNull([Kraj/region]), " ", [Kraj])	Używa funkcji Iif i IsNull w celu wyświetlenia pustego ciągu, jeśli wartość kontrolki Kraj/region ma wartość null. W przeciwnym razie wyświetla wartość kontrolki Kraj/region.
=Iif(IsNull([Region]),[Miasto]&" "&[KodPocztowy],[Miasto]&" "&[Region]&" "&[KodPocztowy])	Używa funkcji Iif i IsNull w celu wyświetlenia wartości kontrolki Miasto i KodPocztowy, jeśli wartość kontrolki Region jest równa null. W przeciwnym razie wyświetla wartości pól lub kontrolki Miasto, Region i KodPocztowy.
=Iif(IsNull([DataDostawy] Or IsNull([DataWysyłki])), "Uwaga: brak daty!", [DataDostawy] - [DataWysyłki])	Używa funkcji Iif i IsNull w celu wyświetlenia komunikatu „Uwaga: brak daty!”, jeśli wynik odejmowania wartości kontrolki DataWysyłki od wartości kontrolki DataDostawy jest równy null. W przeciwnym razie wyświetla interwał między wartościami dat w kontrolkach DataDostawy i DataWysyłki.

Przykłady wyrażeń używanych w zapytaniach i filtrach

Ta sekcja zawiera przykłady wyrażeń używanych w celu utworzenia pola obliczeniowego w zapytaniu lub wprowadzenia kryteriów do zapytania. Pole obliczeniowe jest kolumną w zapytaniu zawierającą wynik wyrażenia.. Przy jego użyciu można na przykład obliczyć wartości, połączyć wartości tekstowe, takie jak imię i nazwisko, lub sformatować część daty.

Używając kryteriów w zapytaniu, można ograniczyć liczbę rekordów, na których są wykonywane operacje. Na przykład w celu określenia daty początkowej i końcowej oraz ograniczenia wyników zapytania do zamówień wysłanych między tymi datami można użyć operatora Between.

W poniższych sekcjach omówiono sposoby dodawania pola obliczeniowego do zapytania i przedstawiono przykłady wyrażeń używanych w zapytaniach.

Dodawanie pola obliczeniowego w widoku projektu zapytania

W okienku nawigacji kliknij prawym przyciskiem myszy kwerendę, którą chcesz zmienić, a następnie w menu skrótów kliknij polecenie Widok projektu.

Kliknij komórkę Pole w kolumnie, w której chcesz utworzyć pole obliczeniowe. Możesz wprowadzić nazwę pola i dwukropek lub wpisać wyrażenie. W przypadku niewprowadzenia nazwy program Access doda wartość Exprn:, gdzie n to numer porządkowy.

Wpisz wyrażenie.

— lub —

Na karcie Projektowanie w grupie Konfiguracja zapytania kliknij przycisk Konstruktor, aby uruchomić Konstruktor wyrażeń.

Aby uzyskać więcej informacji na temat korzystania z Konstruktor wyrażeń, zobacz artykuł Tworzenie wyrażenia.

Wyrażenia zmieniające tekst w zapytaniu lub filtrze

Wyrażenia w poniższej tabeli używają operatorów & i + w celu połączenia ciągów tekstowych, używają wbudowanych funkcji do wykonywania działań na ciągu tekstowym lub operują na tekście w inny sposób, aby utworzyć pole obliczeniowe.

Wyrażenie	Opis
ImięNazwisko: [Imię] & " " & [Nazwisko]	Tworzy pole o nazwie ImięNazwisko wyświetlające wartości pól Imię i Nazwisko oddzielone spacją.
Adres2: [Miasto] & " " & [Region] & " " & [KodPocztowy]	Tworzy pole o nazwie Adres2 wyświetlające wartości pól Miasto, Region i KodPocztowy oddzielone spacjami.
InicjałProduktu: Left([NazwaProduktu], 1)	Tworzy pole o nazwie InicjałProduktu, a następnie używa funkcji Left w celu wyświetlenia w polu InicjałProduktu pierwszego znaku wartości pola NazwaProduktu.
KodTypu: Right([KodSprzętu], 2)	Tworzy pole o nazwie KodTypu, a następnie używa funkcji Right w celu wyświetlenia ostatnich dwóch znaków wartości pola KodSprzętu.
Kierunkowy: Mid([Telefon], 2, 3)	Tworzy pole o nazwie Kierunkowy, a następnie używa funkcji Mid w celu wyświetlenia trzech znaków, licząc od drugiego znaku wartości pola Telefon.

Wyrażenia wykonujące operacje arytmetyczne w polach obliczeniowych

Za pomocą wyrażeń można dodawać, odejmować, mnożyć, dzielić wartości z dwóch lub większej liczby pól bądź kontroltek. Można również wykonywać operacje arytmetyczne na datach. Załóżmy na przykład, że istnieje pole typu Data/godzina o nazwie DataWymagana. Wyrażenie =[DataWymagana] - 2 zwraca wartość typu Data/godzina oznaczającą datę o dwa dni wcześniejszą niż wartość pola DataWymagana.

Wyrażenie	Opis
FrachtPriorytetowy: [Fracht] * 1,1	Tworzy pole o nazwie FrachtPriorytetowy, a następnie wyświetla w tym polu wartość opłat frachtowych powiększoną o 10%.
WielkośćZamówienia: [Ilość] * [CenaJednostkowa]	Tworzy pole o nazwie WielkośćZamówienia, a następnie wyświetla iloczyn wartości pól Ilość i CenaJednostkowa.
Wyprzedzenie: [DataDostawy] - [DataWysyłki]	Tworzy pole o nazwie Wyprzedzenie, a następnie wyświetla różnicę między wartościami pól DataDostawy i DataWysyłki.
ZapasCałkowity: [StanMagazynu] + [JednostkiZamówione]	Tworzy pole o nazwie ZapasCałkowity, a następnie wyświetla sumę wartości pól StanMagazynu i JednostkiZamówione.
FrachtProcentowo: Sum([Fracht]) / Sum([SumaCzęściowa]) * 100	Tworzy pole o nazwie FrachtProcentowo, a następnie wyświetla procent opłaty frachtowej w każdej sumie częściowej. Wyrażenie używa funkcji Sum w celu podsumowania wartości

Wyrażenie	Opis
	<p>w polu Fracht, a następnie podzielenia tych sum przez sumę wartości w polu SumaCzęściowa.</p> <p>Aby użyć tego wyrażenia, należy przekonwertować zapytanie wybierające na zapytanie podsumowujące, ponieważ konieczne jest użycie wiersza Podsumowanie w siatce projektu i ustawienie wartości komórki Podsumowanie tego pola na Wyrażenie.</p> <p>Aby uzyskać więcej informacji na temat tworzenia zapytań podsumowujących, zobacz artykuł Sumowanie danych przy użyciu zapytania.</p> <p>Jeśli właściwość Format pola jest ustawiona na Procent, w wyrażeniu nie należy umieszczać elementu *100.</p>

Aby uzyskać więcej informacji na temat używania funkcji agregujących i sumowania wartości pól lub kolumn, zobacz artykuły Sumowanie danych przy użyciu zapytania, Zliczanie danych przy użyciu zapytania, Zliczanie wierszy w arkuszu danych i Wyświetlanie sum kolumn w arkuszu danych.

Wyrażenia zmieniające daty i wykonujące obliczenia na datach w polach obliczeniowych

W niemal wszystkich bazach danych są przechowywane i śledzone daty oraz godziny. Praca z datami i godzinami w programie Access jest możliwa po ustawieniu w polach tabeli zawierających daty i godziny typu danych Data/godzina. W programie Access możliwe jest wykonanie obliczeń arytmetycznych na datach. Można na przykład obliczyć, ile dni upłynęło od czasu wystawienia faktury w celu wickowania należności.

Wyrażenie	Opis
CzasZwłoki: DateDiff("d", [DataZamówienia], [DataWysyłki])	Tworzy pole o nazwie CzasZwłoki, a następnie używa funkcji DateDiff w celu wyświetlenia liczby dni między datą zamówienia a datą wysyłki.
RokZatrudnienia: DatePart("rrrr", [RokZatrudnienia])	Tworzy pole o nazwie RokZatrudnienia, a następnie używa funkcji DatePart w celu wyświetlenia roku zatrudnienia każdego pracownika.
Minus30: Date() - 30	Tworzy pole o nazwie Minus30, a następnie używa funkcji Date w celu wyświetlenia daty wcześniejszej o 30 dni od bieżącej daty.

Wyrażenia liczące, sumujące i uśredniające wartości przy użyciu funkcji agregujących SQL lub funkcji agregujących dziedziny

Wyrażenia w poniższej tabeli używają funkcji języka SQL (Structured Query Language) w celu zagregowania lub podsumowania danych. Te funkcje (na przykład Sum, Count i Avg) często są nazywane funkcjami agregującymi.

Oprócz funkcji agregujących program Access udostępnia również funkcje agregujące dziedziny służące do selektywnego sumowania lub liczenia wartości. Dzięki nim możliwe jest na przykład policzenie wartości tylko z określonego zakresu lub wyszukanie wartości w innej tabeli. Zbiór funkcji agregujących dziedziny zawiera funkcję DSum, funkcję DCount i funkcję DAvg.

Aby obliczyć sumy, często jest konieczne utworzenie zapytania podsumowującego. Zapytania podsumowującego należy na przykład użyć do podsumowania danych w grupach. Aby przejść do zapytania podsumowującego z widoku siatki projektu, kliknij pozycję Sumy w menu Widok.

Wyrażenie	Opis
LicznikWierszy: Count(*)	Tworzy pole o nazwie LicznikWierszy, a następnie używa funkcji Count w celu policzenia liczby rekordów w zapytaniu, w tym również rekordów z polami o wartości null (pustymi).
FrachtProcentowo: Sum([Fracht]) / Sum([SumaCzęściowa]) * 100	Tworzy pole o nazwie FrachtProcentowo, a następnie oblicza procent, jaki stanowią opłaty frachtowe w każdej sumie częściowej, przez podzielenie sumy wartości pola Fracht przez sumę wartości pola SumaCzęściowa. (W tym przykładzie użyto funkcji Sum).

Wyrażenie	Opis
	<p>Uwaga : Tego wyrażenia należy użyć z zapytaniem podsumowującym. Jeśli właściwość Format pola jest ustawiona na Procent, w wyrażeniu nie należy umieszczać elementu *100.</p> <p>Aby uzyskać więcej informacji na temat tworzenia zapytań podsumowujących, zobacz artykuł Sumowanie danych przy użyciu zapytania.</p>
ŚredniFracht: DAvg("[Fracht]", "[Zamówienia]")	Tworzy pole o nazwie ŚredniFracht, a następnie używa funkcji DAvg w celu obliczenia średniej opłaty frachtowej dla wszystkich zamówień połączonych w zapytaniu podsumowującym.

Wyrażenia działające z polami bez informacji (pola o wartości null)

Przedstawione poniżej wyrażenia działają z polami, w których może brakować informacji, czyli na przykład z polami mającymi wartość null (nieznaną lub niezdefiniowaną). Często spotyka się wartości null, takie jak nieznaną cenę nowego produktu lub wartość, którą współpracownik zapomniał dodać do zamówienia. Zdolność bazy danych do wyszukiwania i przetwarzania wartości null może być krytyczna dla jej działania, a wyrażenia w poniższej tabeli stanowią przykłady typowych sposobów przetwarzania wartości null.

Wyrażenie	Opis
BieżącyKrajRegion: IIf(IsNull([KrajRegion]), " ", [KrajRegion])	Tworzy pole o nazwie BieżącyKrajRegion, a następnie używa funkcji IIf i IsNull w celu wyświetlenia pustego ciągu w polu, jeśli pole KrajRegion ma wartość null. W przeciwnym razie wyświetla zawartość pola KrajRegion.
Wyprzedzenie: IIf(IsNull([DataWymagana] - [DataWysyłki]), "Uwaga: brak daty!", [DataWymagana] - [DataWysyłki])	Tworzy pole o nazwie Wyprzedzenie, a następnie używa funkcji IIf i IsNull w celu wyświetlenia komunikatu „Uwaga: brak daty!”, jeśli jedno z pól DataWymagana lub DataWysyłki ma wartość null. W przeciwnym razie wyświetla różnicę w tych datach.
SprzedażPółroczna: SprzedażPółroczna: Nz([SprzedażKw1]) + Nz([SprzedażKw2])	Tworzy pole o nazwie SprzedażPółroczna, a następnie wyświetla sumę wartości pól SprzedażKw1 i SprzedażKw2, używając wcześniej funkcji Nz w celu przekonwertowania wszystkich wartości null na zero.

Wyrażenie używające podzapytania w celu utworzenia pola obliczeniowego


Pole obliczeniowe można utworzyć także przy użyciu zapytania zagnieżdżonego, nazywanego również podzapytaniem. Wyrażenie w poniższej tabeli jest przykładem pola obliczeniowego pobierającego wyniki z podzapytania.

Wyrażenie	Opis
Kategoria: (SELECT [NazwaKategorii] FROM [Kategorie] WHERE [Produkty].[IDKategorii] = [Kategorie].[IdentyfikatorKategorii])	Tworzy pole o nazwie Kategoria, a następnie wyświetla wartość pola NazwaKategorii, jeśli wartość IdentyfikatorKategorii z tabeli Kategorie jest taka sama jak wartość IdentyfikatorKategorii z tabeli Produkty.

Wyrażenia definiujące kryteria i ograniczające liczbę rekordów w zestawie wyników

Za pomocą wyrażenia można definiować kryteria dla zapytania. Program Access zwraca wówczas tylko wiersze spełniające kryteria. W tej sekcji przedstawiono podstawowe informacje dotyczące dodawania kryteriów do zapytania, a w tabelach ujęto przykłady kryteriów dopasowujących wartości typu tekst i data.


Dodawanie kryterium do zapytania

W okienku nawigacji kliknij prawym przyciskiem myszy zapytanie, które chcesz zmienić, a następnie w menu skrótów kliknij polecenie Widok projektu .

W wierszu Kryteria siatki projektu kliknij komórkę w kolumnie, której chcesz użyć, a następnie wpisz kryterium.

Jeśli potrzebujesz większego obszaru do wpisania wyrażenia, naciśnij klawisze SHIFT+F2, aby wyświetlić pole Powiększenie.

— lub —

Kliknij kartę Projektowanie, a następnie w grupie Konfiguracja zapytania kliknij przycisk Konstruktor , aby utworzyć wyrażenie za pomocą Konstruktora wyrażień.

Uwaga : Podczas tworzenia wyrażenia definiującego kryteria nie należy go poprzedzać operatorem =.

Aby uzyskać więcej informacji na temat korzystania z Konstruktora wyrażień, zobacz artykuł Tworzenie wyrażenia.

Wyrażenia dopasowujące cały tekst lub jego część

Przykładowe wyrażenia w poniższej tabeli przedstawiają kryteria dopasowujące cały tekst lub jego część.

Pole	Wyrażenie	Opis
MiastoOdbiorcy	Kraków	Wyświetla zamówienia wysłane do Krakowa.
MiastoOdbiorcy	"Kraków" Or "Warszawa"	Używa operatora Or w celu wyświetlenia zamówień wysłanych do Krakowa lub Warszawy.
KrajRegionWysyłki	In("Kanada", "Wielka Brytania")	Używa operatora In w celu wyświetlenia zamówień wysłanych do Kanady lub Wielkiej Brytanii.
KrajRegionWysyłki	Not "USA"	Używa operatora Not w celu wyświetlenia zamówień wysłanych do krajów/regionów innych niż USA.
ProductName	Not Like "C*"	Używa operatora Not Like i symbolu wieloznacznego * w celu wyświetlenia produktów, których nazwy nie zaczynają się od litery C.
NazwaFirmy	>="N"	Wyświetla zamówienia wysłane do firm, których nazwy zaczynają się na litery od N do Z.
KodProduktu	Right([KodProduktu], 2)="99"	Używa funkcji Right w celu wyświetlenia zamówień, w których wartość pola KodProduktu kończy się na 99.
NazwaOdbiorcy	Like "S*"	Wyświetla zamówienia wysłane do klientów, których nazwa zaczyna się od litery S.

Wyrażenia używające dat w kryteriach dopasowywania

Wyrażenia w poniższej tabeli przedstawiają sposób użycia dat i związanych z nimi funkcji w wyrażeniach kryteriów.

Aby uzyskać więcej informacji na temat wprowadzania i używania wartości dat, zobacz artykuł Wprowadzanie wartości daty lub godziny. Aby uzyskać informacje dotyczące używania funkcji zawartych w poniższych przykładowych wyrażeniach, kliknij linki do tematów dotyczących konkretnych funkcji.

Pole	Wyrażenie	Opis
DataWysyłki	#02.02.07#	Wyświetla zamówienia wysłane w dniu 2 lutego 2007 r..
DataWysyłki	Data	Wyświetla zamówienia wysłane danego dnia.
DataWymagana	Between Date() And DateAdd(";", 3; Date())	Używa operatora Between...And oraz funkcji DateAdd i Date w celu wyświetlenia zamówień wymaganych między bieżącym dniem a datą o trzy miesiące później.
DataZamówienia	< Date() - 30	Używa funkcji Date w celu wyświetlenia zamówień złożonych przed datą 30 dni wcześniej.
DataZamówienia	Year([DataZamówienia])=2007	Używa funkcji Year w celu wyświetlenia zamówień złożonych w roku 2007.

Pole	Wyrażenie	Opis
DataZamówienia	DatePart("q", [DataZamówienia])=4	Używa funkcji DatePart w celu wyświetlenia zamówień z czwartego kwartału.
DataZamówienia	DateSerial(Year([DataZamówienia]), Month([DataZamówienia])+1, 1)-1	Używa funkcji DateSerial, Year i Month w celu wyświetlenia zamówień z ostatniego dnia każdego miesiąca.
DataZamówienia	Year([DataZamówienia]) = Year(Now()) And Month([DataZamówienia]) = Month(Now())	Używa funkcji Year i Month oraz operatora And w celu wyświetlenia zamówień dla bieżącego roku i miesiąca.
DataWysyłki	Between #05.01.07# And #10.01.07#	Używa operatora Between...And w celu wyświetlenia zamówień wysłanych nie wcześniej niż w dniu 05.01.2007 i nie później niż w dniu 10.01.2007.
DataWymagana	Between Date() And DateAdd("M", 3, Date())	Używa operatora Between...And w celu wyświetlenia zamówień wymaganych między danym dniem i datą o trzy miesiące później.
DataUrodzenia	Month([DataUrodzenia])=Month(Date())	Używa funkcji Month i Date w celu wyświetlenia danych pracowników, których urodziny przypadają w danym miesiącu.

Wyrażenia dopasowujące brakującą wartość (null) lub ciąg o zerowej długości

Wyrażenia w poniższej tabeli działają z polami, w których potencjalnie brakuje informacji — czyli takimi, które mogą zawierać wartość null lub ciąg o zerowej długości. Wartość null oznacza brak informacji. Nie oznacza ona wartości zerowej ani żadnej innej wartości. Program Access umożliwia obsługę pól z brakującymi informacjami, ponieważ jest ona niezbędna dla zapewnienia integralności bazy danych. W rzeczywistym świecie często brakuje informacji, nawet jeśli to tylko stan tymczasowy (na przykład gdy nie ustalono jeszcze ceny nowego produktu). Dlatego baza danych odzwierciedlająca element świata rzeczywistego, na przykład firmę, musi mieć możliwość rejestrowania braku informacji. Do rozpoznawania wartości null w polu lub kontrolce służy funkcja IsNull, natomiast funkcja Nz służy do konwertowania wartości null na zero.

Pole	Wyrażenie	Opis
RegionOdbiorcy	Is Null	Wyświetla zamówienia klientów, których pole RegionOdbiorcy zawiera wartość null (brak wartości).
RegionOdbiorcy	Is Not Null	Wyświetla zamówienia klientów, których pole RegionOdbiorcy nie jest puste.
Faks	""	Wyświetla zamówienia klientów, którzy nie mają urządzeń faksujących, na co wskazuje ciąg zerowej długości w polu Faks, zamiast wartości null (brak wartości).

Wyrażenia używające wzorców w celu dopasowania rekordów

Operator Like zapewnia znaczną elastyczność podczas dopasowywania wierszy zgodnych ze wzorcem, ponieważ definiując wzorce dla programu Access, można używać operatora Like z symbolami wieloznacznymi. Na przykład symbol wieloznacznym * (gwiazdka) zapewnia zgodność z dowolnym ciągiem znaków i ułatwia wyszukanie wszystkich nazw rozpoczynających się od danej litery. Na przykład wyrażenie Like "S*" umożliwia wyszukanie wszystkich nazw rozpoczynających się od litery S.

Aby uzyskać więcej informacji na ten temat, zobacz artykuł Operator Like.

Pole	Wyrażenie	Opis
NazwaOdbiorcy	Like "S*"	Znajduje wszystkie rekordy, w których wartość pola NazwaOdbiorcy rozpoczyna się od litery S.
NazwaOdbiorcy	Like "*Contoso"	Znajduje wszystkie rekordy, w których wartość pola NazwaOdbiorcy kończy się wyrazem „Contoso”.

Pole	Wyrażenie	Opis
NazwaOdbiorcy	Like "[A-D]*"	Znajduje wszystkie rekordy, w których wartość pola NazwaOdbiorcy rozpoczyna się od litery A, B, C lub D.
NazwaOdbiorcy	Like "*ar*"	Znajduje wszystkie rekordy, w których wartość pola NazwaOdbiorcy zawiera sekwencję liter „ar”.
NazwaOdbiorcy	Like "Maison Dewe?"	Znajduje wszystkie rekordy, w których wartość pola NazwaOdbiorcy zawiera wyraz „Maison” w pierwszej części wartości i pięcioliterowy ciąg, w którym cztery pierwsze litery to „Dewe”, a ostatnia litera jest nieznana.
NazwaOdbiorcy	Not Like "A*"	Znajduje wszystkie rekordy, w których wartość pola NazwaOdbiorcy nie rozpoczyna się od litery A.

Wyrażenia dopasowujące wiersze na podstawie wyników funkcji agregującej dziedziny

Aby selektywnie zsumować, policzyć lub uśrednić wartości, należy skorzystać z funkcji agregującej dziedziny. Można na przykład zliczyć tylko wartości z określonego zakresu lub prowadzące do uzyskania wartości Tak. Innym przykładem może być potrzeba wyszukania wartości w innej tabeli w celu wyświetlenia tej wartości. W przykładowych wyrażeniach w poniższej tabeli użyto funkcji agregujących dziedziny w celu wykonania obliczeń na zbiorze wartości i użycia wyników jako kryterium zapytania.

Pole	Wyrażenie	Opis
Fracht	> (DStDev("[Fracht]", "Zamówienia") + DAVg("[Fracht]", "Zamówienia"))	Używa funkcji DStDev i DAVg w celu wyświetlenia wszystkich zamówień, w których koszty frachtu przekroczyły wartość średniego frachtu powiększoną o odchylenie standardowe.
Ilość	> DAVg("[Ilość]", "[Szczegóły zamówień]")	Używa funkcji DAVg w celu wyświetlenia produktów zamówionych w ilościach przekraczających średnią wielkość zamówienia.

Wyrażenia dopasowujące na podstawie wyników podzapytań

Wartość do użycia jako kryterium można także obliczyć przy użyciu zapytania zagnieżdżonego, nazywanego również podzapytaniem. Przykładowe wyrażenia w poniższej tabeli dopasowują wiersze na podstawie wyników zwróconych przez podzapytanie.

Pole	Wyrażenie	Wyświetlana wartość
CenaJednostkowa	(SELECT [CenaJednostkowa] FROM [Produkty] WHERE [NazwaProduktu] = "Syrop anyżowy")	Produkty, których cena jest taka sama, jak cena Syropu anyżowego.
CenaJednostkowa	> (SELECT AVG([CenaJednostkowa]) FROM [Produkty])	Produkty, których cena przekracza średnią cenę.
Wynagrodzenia	> ALL (SELECT [Wynagrodzenia] FROM [Pracownicy] WHERE ([Stanowisko] LIKE "*Kierownik*") OR ([Stanowisko] LIKE "*Wiceprezes*"))	Wynagrodzenie każdego przedstawiciela handlowego zarabiającego więcej niż każdy z pracowników, których nazwa stanowiska zawiera słowo „Kierownik” lub „Wiceprezes”.
WielkośćZamówienia: [CenaJednostkowa] * [Ilość]	> (SELECT AVG([CenaJednostkowa] * [Ilość]) FROM [Szczegóły zamówień])	Zamówienia, których suma jest wyższa niż średnia wartość zamówienia.

Wyrażenia używane w zapytaniach aktualizujących

Zapytanie aktualizujące służy do modyfikowania danych w co najmniej jednym istniejącym polu bazy danych. Na przykład można przy jego użyciu zastąpić wartości lub usunąć je całkowicie. Poniższa tabela przedstawia niektóre sposoby użycia wyrażeń w zapytaniach aktualizujących. Tych wyrażeń należy użyć w wierszu Aktualizacja do w siatce projektu zapytania dla pola, które ma zostać zaktualizowane.

Aby uzyskać więcej informacji o tworzeniu zapytań aktualizujących, zobacz temat Tworzenie zapytania aktualizującego.

Pole	Wyrażenie	Wynik
Stanowisko	"Sprzedawca"	Zmienia wartość tekstową na wyraz „Sprzedawca”.
RozpoczęcieProjektu	#10.08.07#	Zmienia datę na 10 sierpnia 2007 r.
Na emeryturze	Tak	Zmienia wartość Nie w polu typu Tak/Nie na Tak.
NumerCzęści	"NC" & [NumerCzęści]	Dodaje oznaczenie „NC” na początku każdego wskazanego numeru części.
SumaPozycji	[CenaJednostkowa] * [Ilość]	Oblicza iloczyn ceny jednostkowej i ilości.
Fracht	[Fracht] * 1,5	Zwiększa opłaty przewozowe o 50 procent.
Sprzedaż	DSum("[Ilość] * [CenaJednostkowa]", "Szczegóły zamówień", "[IdentyfikatorProduktu]=" & [IdentyfikatorProduktu])	Jeśli identyfikatory produktów w bieżącej tabeli odpowiadają identyfikatorom produktów w tabeli Szczegóły zamówień, aktualizuje podsumowania sprzedaży na podstawie ilości i ceny jednostkowej produktu.
KodPocztowyOdbiorcy	Right([KodPocztowyOdbiorcy];5)	Odrzuca początkowe znaki, pozostawiając pięć pierwszych znaków od prawej.
CenaJednostkowa	Nz([CenaJednostkowa])	Zmienia wartość null (wartość nieznaną lub niezdefiniowaną) na zero (0) w polu o nazwie CenaJednostkowa.

Wyrażenia używane w instrukcjach SQL

SQL jest językiem używanym w programie Access do tworzenia zapytań. Każde zapytanie tworzone w widoku projektu można przedstawić przy użyciu języka SQL. Aby wyświetlić instrukcję SQL dla danego zapytania, kliknij polecenie Widok SQL w menu Widok. W poniższej tabeli przedstawiono przykłady instrukcji SQL zawierających wyrażenia.

Instrukcja SQL zawierająca wyrażenie	Wynik
SELECT [Imię],[Nazwisko] FROM [Pracownicy] WHERE [Nazwisko]="Kowalski"	Wyświetla wartości pól Imię i Nazwisko pracowników o nazwisku Kowalski.
SELECT [IdentyfikatorProduktu], [NazwaProduktu] FROM [Produkty] WHERE [IdentyfikatorKategorii] = Forms![NoweProdukty]![IdentyfikatorKategorii];	Wyświetla wartości pól IdentyfikatorProduktu i NazwaProduktu z tabeli Produkty tych rekordów, w których wartość IdentyfikatorKategorii zgadza się z wartością IdentyfikatorKategorii w otwartym formularzu NoweProdukty.
SELECT Avg([CenaRozszerzona]) AS [Średnia cena końcowa] FROM [RozszerzoneSzczegółyZamówień] WHERE [CenaRozszerzona] > 1000;	Oblicza średnią cenę końcową dla zamówień, których wartość w polu CenaRozszerzona przekracza wartość 1000, a następnie wyświetla wynik w polu o nazwie „Średnia cena końcowa”.
SELECT [IdentyfikatorKategorii],Count([IdentyfikatorProduktu]) AS [Liczba ID produktów] FROM [Produkty] GROUP BY [IdentyfikatorKategorii] HAVING Count([IdentyfikatorProduktu]) > 10;	W polu o nazwie „Liczba ID produktów” wyświetla łączną liczbę produktów dla kategorii, które zawierają ponad 10 produktów.


Przykłady wyrażeń wartości domyślnych

Podczas projektowania bazy danych może wystąpić potrzeba przypisania wartości domyślnej do pola lub kontrolki. Program Access wprowadza wówczas wartość domyślną podczas tworzenia nowego rekordu z danym polem lub obiektem zawierającego daną kontrolkę. Wyrażenia w poniższej tabeli przedstawiają przykłady wartości domyślnych dla pola lub kontrolki.

Dodawanie wartości domyślnej pola tabeli

W okienku nawigacji kliknij prawym przyciskiem myszy tabelę, którą chcesz zmienić, i w menu skrótów kliknij polecenie Widok projektu.

Kliknij pole, które chcesz zmienić, a następnie na karcie Ogólne kliknij pole właściwości Wartość domyślna.

Wpisz wyrażenie lub kliknij przycisk Konstruujuj  po prawej stronie pola właściwości, aby utworzyć wyrażenie za pomocą Konstruktor wyrażzeń.

Jeśli kontrolka jest powiązana z polem tabeli, które ma wartość domyślną, wartość domyślna kontrolki ma pierwszeństwo.

Pole	Wyrażenie	Domyślna wartość w polu
Ilość	1	1
Region	"MT"	MT
Region	"Szczecin"	Szczecin Szczecin (należy zauważyć, że wartość zawierająca znaki interpunkcyjne musi być ujęta w cudzysłowy).
Faks	""	Ciąg zerowej długości wskazuje, że dane pole powinno być domyślnie puste, zamiast zawierać wartość null.
Data zamówienia	Date()	Dzisiejsza data
Data wykonania	Date() +60	Data 60 dni późniejsza od daty bieżącej.

Przykłady wyrażzeń reguł sprawdzania poprawności dla pola


Przy użyciu wyrażenia można utworzyć regułę sprawdzania poprawności dla pola lub kontrolki. Program Access wymusza wówczas tę regułę podczas wprowadzania danych do pola lub kontrolki. Aby utworzyć regułę sprawdzania poprawności, należy zmodyfikować właściwość RegułaPoprawności pola lub kontrolki. Należy również rozważyć ustawienie wartości właściwości KomunikatOBłędzie przechowującej tekst komunikatu wyświetlanego przez program Access w przypadku naruszenia reguły sprawdzania poprawności. W przypadku nieustawienia właściwości KomunikatOBłędzie program Access wyświetla domyślny komunikat o błędzie.

Dodawanie reguły sprawdzania poprawności dla pola

W okienku nawigacji kliknij prawym przyciskiem myszy tabelę, którą chcesz zmienić, i w menu skrótów kliknij polecenie Widok projektu.

Kliknij pole, które chcesz zmienić.

Kliknij pole właściwości RegułaPoprawności znajdujące się w dolnej sekcji projektanta tabel.

Wpisz wyrażenie lub kliknij przycisk Konstruujuj  po prawej stronie pola właściwości, aby utworzyć wyrażenie za pomocą Konstruktor wyrażzeń.

Uwaga : Tworząc regułę poprawności, nie poprzedzaj wyrażenia operatorem =.

Przykłady w poniższej tabeli przedstawiają wyrażenia reguły sprawdzania poprawności dla właściwości RegułaPoprawności i przypisany do niej tekst właściwości KomunikatOBłędzie.

Właściwość RegułaPoprawności	Właściwość KomunikatOBłędzie
<> 0	Wprowadź wartość różną od zera.
0 Or > 100	Wartość musi być równa 0 lub większa niż 100.
Like "K???"	Wartość musi być złożona z czterech znaków i musi zaczynać ją litera K.
< #01.01.07#	Wprowadź datę wcześniejszą niż 01.01.2007.
>= #01.01.2007# And < #01.01.2008#	Data musi przypadać w 2007 roku.

Aby uzyskać więcej informacji na temat sprawdzania poprawności danych, zobacz artykuł Tworzenie reguły sprawdzania poprawności do sprawdzania poprawności danych w polu.

Przykłady wyrażeń w warunkach makr

W niektórych przypadkach może być konieczne wykonanie akcji lub serii akcji w makrze tylko w przypadku spełnienia określonego warunku. Załóżmy na przykład, że chcemy, aby pewna akcja była wykonywana tylko wówczas, gdy wartość pola tekstowego Licznik wynosi 10. Stosowny warunek makra można zdefiniować, używając odpowiedniego wyrażenia w kolumnie Warunek: [Licznik]=10.

Dodawanie warunku dla akcji makra

W okienku nawigacji kliknij prawym przyciskiem myszy makro, które chcesz zmienić, a następnie kliknij w menu skrótów polecenie Widok projektu.

Aby wyświetlić kolumnę Warunek w projektancie makr, na karcie Projektowanie w grupie Pokazywanie/ukrywanie kliknij przycisk Warunki.

Kliknij komórkę Warunek dla akcji makra, które chcesz zmienić, a następnie wpisz wyrażenie warunkowe.

Zapisz zmiany i zamknij makro.

Podobnie jak w przypadku właściwości RegułaPoprawności wyrażenie w kolumnie Warunek jest wyrażeniem warunkowym. Musi ono zwracać wartość Prawda lub Fałsz. Akcja jest wykonywana tylko wówczas, gdy warunek jest spełniony.

Wyrażenie służące do wykonania akcji	If
[Miasto]="Wrocław"	Wrocław jest wartością kontrolki Miasto w formularzu, z poziomu którego zostało uruchomione makro.
DCount("[IdentyfikatorZamówienia]", "Zamówienia") > 35	Istnieje więcej niż 35 elementów w polu IdentyfikatorZamówienia tabeli Zamówienia.
DCount("*", "[Szczegóły zamówień]", "[IdentyfikatorZamówienia]=" & Forms![Zamówienia].[IdentyfikatorZamówienia]) > 3	W tabeli Szczegóły zamówień istnieją więcej niż 3 elementy, dla których pole IdentyfikatorZamówienia jest zgodne z polem IdentyfikatorZamówienia w formularzu Zamówienia.
[DataWysyłki] Between #2-lut-2007# And #2-mar-2007#	Data w polu DataWysyłki w formularzu, z którego zostało uruchomione makro, nie jest wcześniejsza niż 2 lutego 2007 r. i nie późniejsza niż 2 marca 2007 r..
Forms![Produkty].[StanMagazynu]<5	Wartość pola StanMagazynu w formularzu Produkty jest mniejsza niż 5.
IsNull([Imię])	Wartość pola Imię w formularzu, z poziomu którego zostało uruchomione makro, jest równa null (nie ma wartości). To wyrażenie jest równoważne z wyrażeniem [Imię] Is Null.
[KrajRegion]="Wielka Brytania" And Forms![PodsumowaniaSprzedaży].[PodsumowaniaZamówień]>100	Wartość pola KrajRegion w formularzu, z poziomu którego zostało uruchomione makro, jest równa Wielka Brytania, a wartość w polu PodsumowaniaZamówień formularza PodsumowaniaSprzedaży jest większa niż 100.
[KrajRegion] In ("Francja", "Włochy", "Hiszpania") And Len([KodPocztowy])<>5	Wartość pola KrajRegion w formularzu, z poziomu którego zostało uruchomione makro, jest równa Francja, Włochy lub Hiszpania, a długość kodu pocztowego jest inna niż 5 znaków.
MsgBox("Czy potwierdzasz zmiany?",1)=1	Kliknięto przycisk OK w oknie dialogowym wyświetlonym przez funkcję MsgBox.. W przypadku kliknięcia w tym oknie dialogowym przycisku Anuluj program Access zignoruje akcję.

Uwaga : Aby wymusić tymczasowe zignorowanie akcji w programie Access wpisz wartość Fałsz jako warunek. Tymczasowe ignorowanie akcji może być pomocne podczas prób zlokalizowania błędów w makrze.