

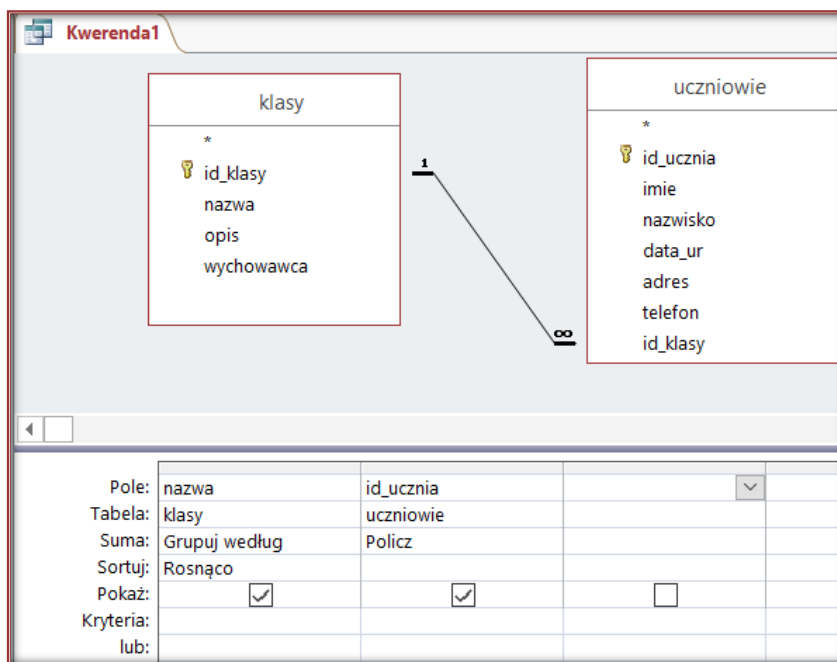
**Spis treści:**

T29: Kwerendy – wiadomości ogólne .....	1
T30: Rodzaje kwerend wykorzystywanych do przetwarzania danych.....	3
T31: Tworzenie nowych kwerend za pomocą kreatora.....	5
T32: Wybór tabel i pól do kwerendy. Określanie kryteriów kwerendy.....	8
T33: Konstruktor wyrażeń.....	14
T34: Kwerendy wybierające.....	16
T35: Funkcje grupujące.....	17
T36: Kwerendy funkcjonalne.....	19
T37: Budowanie kwerend krzyżowych.....	24
T38: Wprowadzenie do SQL.....	27
T39: Język SQL w programie Access.....	27
T40: Kwerenda wybierająca w języku SQL.....	31

**T29: Kwerendy – wiadomości ogólne**

**Kwerenda** jest zapytaniem skierowanym do tabeli lub innej kwerendy. W wyniku jej wykonania wyświetlone zostaną rekordy tabeli w liczbie i kolejności spełniających warunki zapytania. Kwerendy pozwalają na różne sposoby oglądać, zmieniać i analizować dane. Można ich również używać jako źródeł rekordów dla następnych obiektów (formularzy, raportów i stron dostępu do danych). Najczęściej używanym typem kwerendy jest kwerenda wybierająca. Kwerenda inaczej to: **query**, zapytanie.

Projektując kwerendę, wybieramy tabele oraz pola, które będą brały udział w kwerendzie, określamy kryteria wyboru rekordów oraz sposób ich wyświetlania. Okno projektowania kwerendy zostało podzielone na dwie części. Górna zawiera obiekty, które biorą udział w kwerendzie (tabele i kwerendy), oraz zdefiniowane relacje, dolna stanowi szablon projektowania kwerendy. W kolejnych kolumnach szablonu umieszczamy pola, które będą dostępne w kwerendzie. Nazwy pól wybieramy z listy w wierszu Pole lub przenosimy z obiektów znajdujących się w górnej części okna, przeciągając je myszą do szablonu lub dwukrotnie klikając wybrane pole.

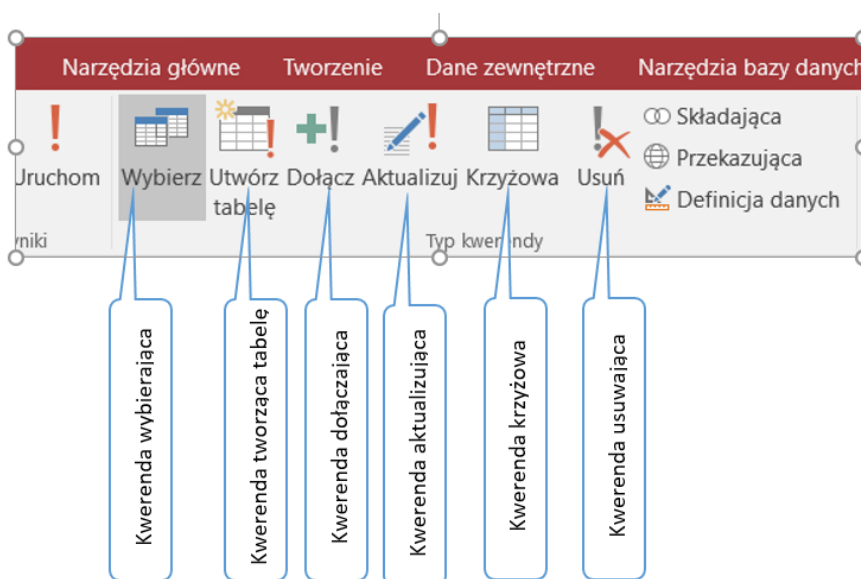


Rysunek 1. Okno projektowania kwerendy.

- Wiersz **tabeta** w szablonie określa, z jakiego obiektu pochodzi pole. Wartość ta jest zwykle ustawiana automatycznie.
- Wiersz **Sortuj** ustala sposób porządkowania danych w kwerendzie.
- Wiersz **Pokaż** umożliwia ukrywanie pól, które służą do określania kryterium, ale nie muszą być widoczne
- Wiersz **Kryteria** określa kryteria wyboru rekordu.

Kwerendy dzielimy na dwie grupy:

- **kwerendy wybierające,**
- **kwerendy funkcjonalne.**



Rysunek 2. Typy kwerend widoczne na pasu zadań w Access

## T30: Rodzaje kwerend wykorzystywanych do przetwarzania danych.

### Rodzaje kwerend:

1. Kwerenda wybierająca:
  - 1.1. podsumowująca
  - 1.2. z polem wyliczeniowym
  - 1.3. z parametrem –inaczej parametryczna
  - 1.4. krzyżowa
2. Kwerenda funkcjonalna
  - 2.1. Kwerenda aktualizująca
  - 2.2. Kwerenda dołączająca
  - 2.3. Kwerenda usuwająca
  - 2.4. Kwerenda tworząca tabelę

Dodatkowo można wyróżnić kwerendy z kreatora kwerend:

- Kwerenda wyszukująca duplikaty
- Kwerenda wyszukująca niepasujące dane

Wszystkie rodzaje kwerend można zapisać w języku SQL – czyli kwerenda SQL

### Kwerendy wybierające

Kwerendy wybierające umożliwiają wybieranie z bazy danych rekordów spełniających określone kryteria. Do grupy tej należą również kwerendy, w których definiujemy pola obliczeniowe czy wykonujemy podsumowania w grupach, oraz kwerendy z parametrem.

### Kwerenda podsumowująca

Działanie kwerend podsumowujących polega na pogrupowaniu danych z tabeli (lub tabel) według wybranego pola, a następnie wykonaniu obliczeń na danych w każdej grupie. Obliczenia odbywają się z wykorzystaniem funkcji agregujących. Funkcje te działają na wartościach wybranego pola w grupie wierszy. Mogą zostać użyte do obliczenia:

- sumy wartości (**Suma**),
- zliczania wierszy o danej wartości (**Policz**),
- wykonania obliczeń statystycznych (**Średnia, OdchStd, Wariancja**),
- wybrania wiersza z wartością najmniejszą lub największą (**Minimum, Maksimum**)
- wybrania wiersza, w którym dana wartość występuje jako pierwsza lub ostatnia (**Pierwszy, Ostatni**).

Funkcje agregujące mogą być używane tylko w kwerendach podsumowujących i są dostępne w oknie projektu kwerendy po rozwinięciu listy w wierszu **Podsumowanie**.

### Kwerenda z polem wyliczeniowym

Prawidłowo zaprojektowana baza danych nie przechowuje w tabelach wartości obliczonych. Jeżeli na podstawie danych zapisanych w tabelach należy obliczyć nowe wartości, można zaprojektować kwerendę z **polem wyliczeniowym**, w której obliczymy i wyświetlimy wymagane wyniki.

### Kwerendy parametryczne

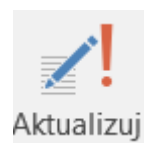
Wykonanie kwerendy parametrycznej powoduje wyświetlenie okna dialogowego, w którym należy podać informacje, takie jak kryteria pobierania rekordów czy wartości, które mają być wstawione w polach. Przy projektowaniu takiej kwerendy należy wartość wpisywaną w polu Kryteria zastąpić tekstem wpisanym w nawiasach kwadratowych [**Parametr**]. Treść jest dowolna, ważne, aby była ujęta w nawiasy. Ta zmiana spowoduje, że przy uruchamianiu kwerendy pojawi się okno dialogowe z wpisanym tekstem i polem tekstowym, do którego trzeba wprowadzić wartość parametru.

**Kwerendy krzyżowe** stosowane są w celu obliczenia i zmiany struktury danych, co ułatwia analizowanie tych danych. Kwerendy krzyżowe obliczają sumę, średnią, zliczają dane lub przeprowadzają inny rodzaj podsumowania dla danych zgrupowanych według dwóch typów informacji — jeden wzdłuż lewej, a drugi wzdłuż górnej krawędzi arkusza danych.

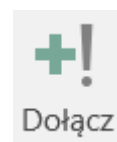
**Kwerenda funkcjonalna** pozwala na wprowadzenie zmian w wielu rekordach lub przeniesienie wielu rekordów przy użyciu pojedynczej operacji. Istnieją cztery rodzaje kwerend funkcjonalnych:



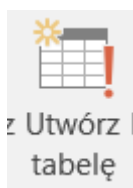
▪ **Kwerendy usuwające**- kwerenda usuwająca usuwa grupę rekordów z jednej lub wielu tabel. Kwerenda usuwająca może być użyta na przykład do usunięcia produktów, które zostały wycofane z oferty i na które nie ma zamówień. Użycie kwerendy usuwającej powoduje usunięcie całych rekordów, nie zaś wybranych pól w rekordach.



▪ **Kwerendy aktualizujące**- kwerenda aktualizująca dokonuje globalnych zmian w grupie rekordów w jednej lub wielu tabelach. Na przykład: można podnieść o 10% ceny wszystkich produktów mlecznych lub dać pięcioprocentową podwyżkę wszystkim pracownikom należącym do określonej kategorii zawodowej. Za pomocą kwerend aktualizujących można zmieniać dane w istniejących tabelach.



▪ **Kwerendy dołączające** - kwerenda dołączająca dodaje na końcu jednej lub wielu tabel grupę rekordów z jednej lub wielu tabel. Przypuśćmy na przykład, że firma pozyskała nowych klientów i istnieje baza danych zawierająca tabelę z informacjami o tych klientach. Aby uniknąć wpisywania wszystkich tych informacji w bazie danych, można dane te dołączyć do tabeli Klienci.



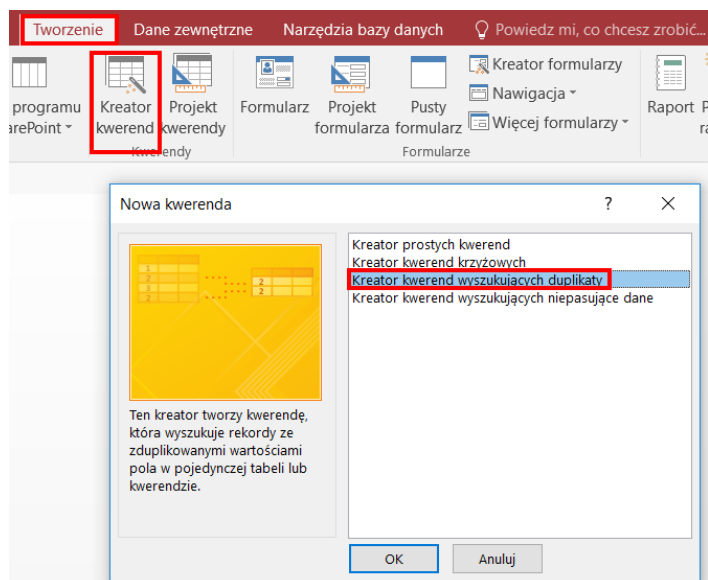
▪ **Kwerendy tworzące table** -kwerenda tworząca table tworzy nową tabelę na podstawie wszystkich lub części danych przechowywanych w jednej lub wielu tabelach. Kwerendy tworzące table są pomocne przy tworzeniu tabel, które mają zostać wyeksportowane do innych baz danych programu Microsoft Access lub tabel historii zawierających stare rekordy.

### Kwerenda wyszukująca duplikaty

Kwerenda pozwala wyszukać powtarzające się rekordy lub wartości pól w tabeli.

Najprostszą metodą utworzenia takiej kwerendy jest wykorzystanie kreatora. W tym celu należy:

1. W oknie bazy danych klikamy kartę tworzenie a następnie kreator kwerend
2. W otwartym oknie dialogowym NOWA KWERENDA wybieramy pozycję Kreator kwerend wyszukujących duplikaty
3. Dalej postępujemy zgodnie z wyświetlanymi instrukcjami.



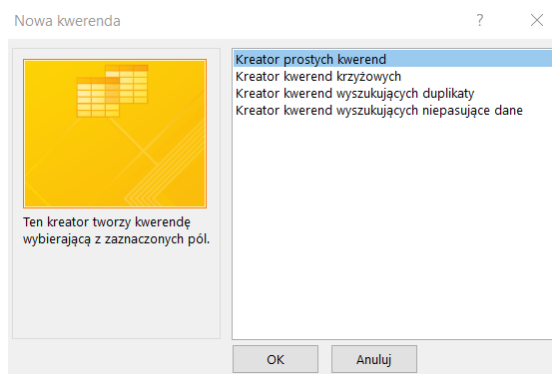
### Kwerenda wyszukująca niedopasowane dane

Za pomocą kwerendy wyszukującej niepasujące dane można wyszukiwać w jednej tabeli rekordy, które nie mają swoich odpowiedników w tej tabeli. Podobnie jak w poprzednim przypadku najprostszą metodą utworzenia takiej kwerendy jest wykorzystanie kreatora. Tak jak wyżej. Oczywiście kwerendę taką możemy wykonać samodzielnie gdyż jest to zwykła kwerenda wybierająca z odpowiednim kryterium (LEFT, RIGHT JOIN).

**Kwerendy SQL** są tworzone na podstawie instrukcji SQL, (Structured Query Language). Gdy w widoku projektu kwerendy tworzona jest kwerenda, program Access konstruuje w tle odpowiednie instrukcje SQL. W rzeczywistości, większość właściwości kwerendy w arkuszu właściwości w widoku Projekt kwerendy ma swoje odpowiedniki wśród klauzul i opcji dostępnych w widoku SQL. Jeśli jest to konieczne, można wyświetlić lub edytować instrukcje SQL w widoku SQL. Jednakże po dokonaniu zmian kwerendy w widoku SQL kwerenda ta może nie być wyświetlana w taki sposób, w jaki była wyświetlana wcześniej w widoku Projekt.

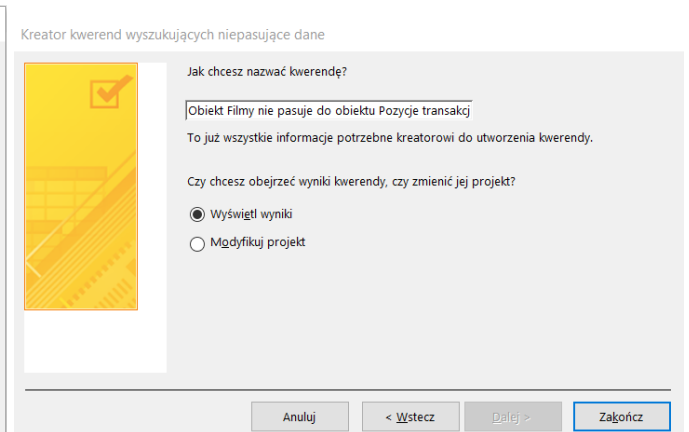
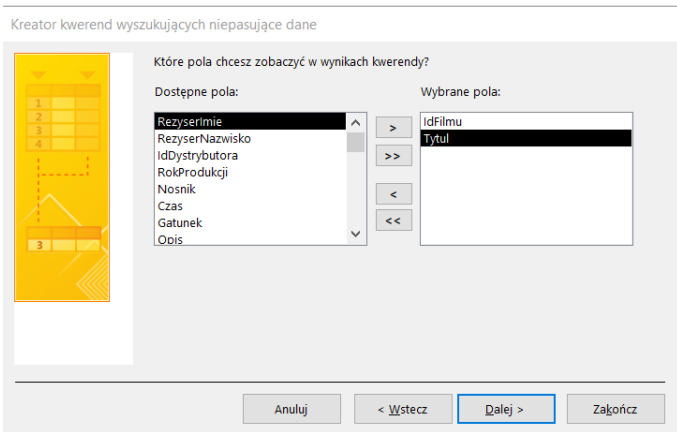
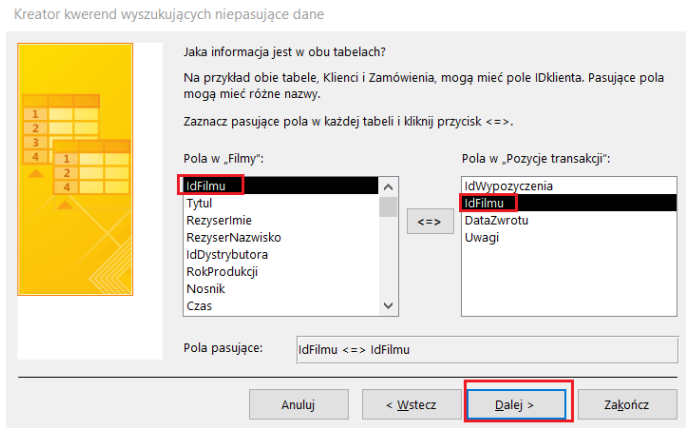
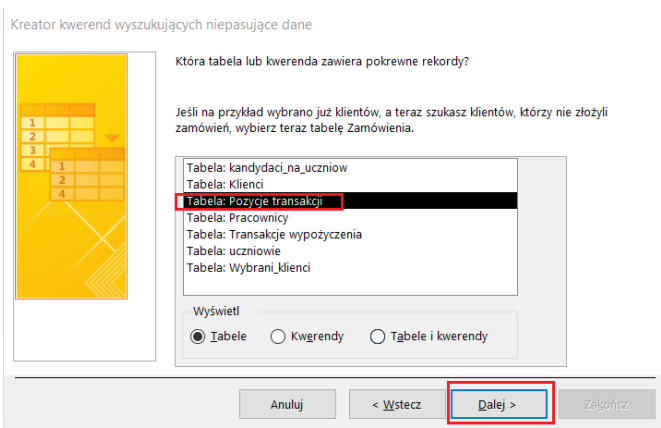
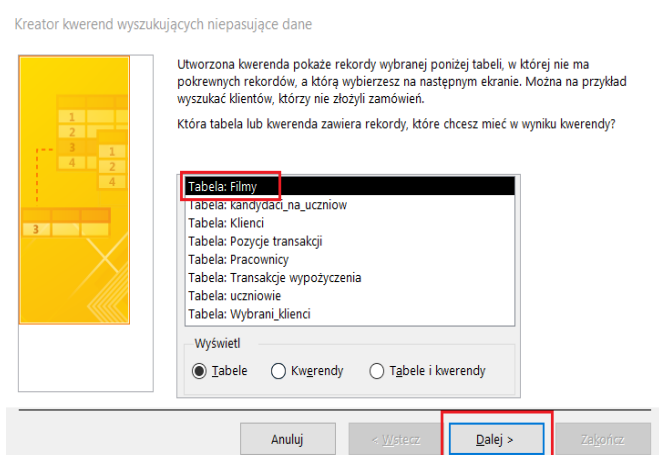
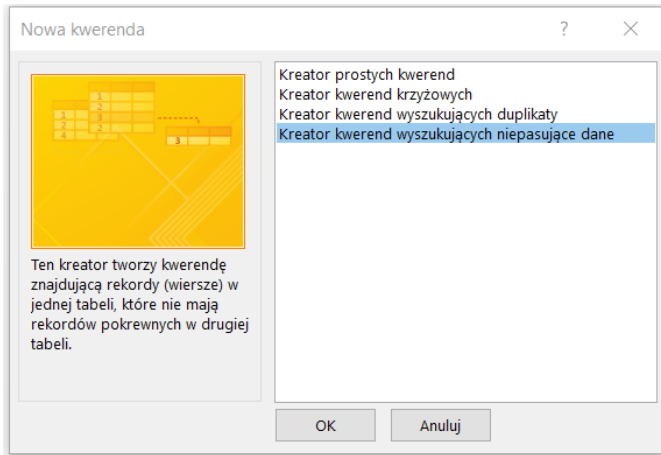
### T31: Tworzenie nowych kwerend za pomocą kreatora.

Tworząc proste kwerendy wybierające lub funkcjonalne, możemy posłużyć się specjalnie do tego celu udostępnionym narzędziem korzystającym z języka QBE(ang. Query ByExample). Jest to kreator kwerend, który w oparciu o graficzny interfejs prowadzi użytkownika przez proces tworzenia zapytania.



Rysunek 3. Okno kreatora kwerend.

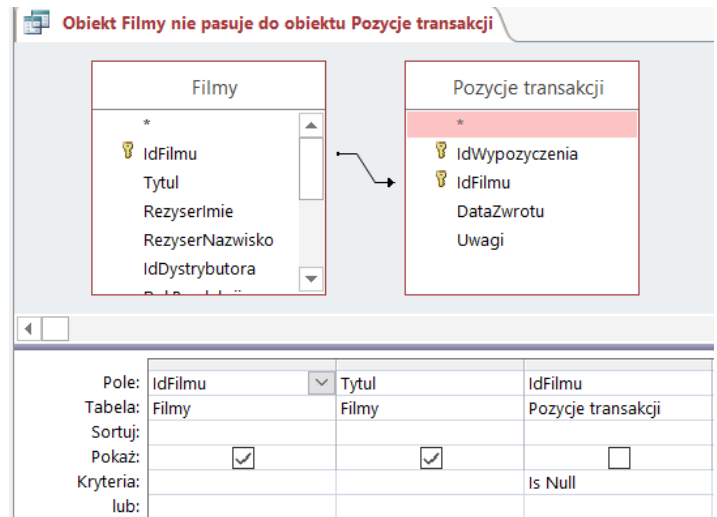
**Przykład zastosowania:** Wyświetl tytuły filmów, które nie były wypożyczone:



## Access -kwerendy

Obiekt Filmy nie pasuje do obiektu Pozycje transakcji

IdFilmu	Tytuł
3	BROKEN FLOWERS
6	101 REYKJAVIK
16	TRAFFIC
17	OBSEŁUGIWAŁEM ANGIELSKIEG
18	INFILTRACJA
19	CZEKOLADA
22	MIASTO ŚLEPCÓW
31	WIERNY OGRODNIK
33	CAST AWAY POZA ŚWIATEM
36	MOTYL I SKAFANDER
37	DZIEŃ ŚWIRA
49	MICHAEL CLAYTON
51	LABIRYNT FAUNA
55	PIĘKNY UMYŚŁ
57	MAŁA MISS
74	CZŁOWIEK BEZ PRZESZŁOŚCI
91	PIANISTKA
95	WĄTLIWOŚĆ



Rysunek 4. Widok projektu

Rysunek 5. Wynik kwerendy- tytuły filmów bez wypożyczeń

### Widok sql -z przykładu:

```
SELECT Filmy.IdFilmu, Filmy.Tytuł
FROM Filmy LEFT JOIN [Pozycje transakcji] ON Filmy.[IdFilmu] = [Pozycje
transakcji].[IdFilmu]
WHERE ((([Pozycje transakcji].IdFilmu) Is Null));
```

**Przykład 2-** Wyświetl z transakcji wypożyczeń, tych pracowników i tych klientów, którzy zostali obsłużeni co najmniej 1 raz –wyszukująca duplikaty.

Kreator kwerend wyszukiwujących duplikaty

Nowa kwerenda

Ten kreator tworzy kwerendę, która wyszukuje rekordy ze zduplikowanymi wartościami pola w pojedynczej tabeli lub kwerendzie.

Kreator prostych kwerend  
 Kreator kwerend krzyżowych  
**Kreator kwerend wyszukiwujących duplikaty**  
 Kreator kwerend wyszukiwujących niepasujące dane

OK Anuluj

Które tabele lub kwerendy chcesz przeszukać pod względem duplikujących się wartości?  
 Aby na przykład znaleźć miasta, w których masz więcej niż jednego klienta, wybierz poniżej tabelę Klienci.

Tabela: Filmy  
 Tabela: kandydaci\_na\_uczniow  
 Tabela: Klienci  
 Tabela: Pozycje transakcji  
 Tabela: Pracownicy  
**Tabela: Transakcje wypożyczenia**  
 Tabela: uczniowie  
 Tabela: Wybrani\_klienci

Wyświetl  
 Tabele  Kwerendy  Tabele i kwerendy

Anuluj < Wstecz **Dalej >** Zakończ

**Kreator kwerend wyszukiujących duplikaty**

Które pola mogą zawierać duplikujące się informacje?  
Jeśli na przykład szukasz miast, w których masz więcej niż jednego klienta, wybierz tu pola Miasto i Region.

Dostępne pola: IdWypozyczenia, DataWypozyczenia, Uwagi  
Pola ze zduplikowaną wartością: IdKlienta, IdPracownika

**Kreator kwerend wyszukiujących duplikaty**

Czy chcesz, aby kwerenda pokazała jakieś pola, oprócz tych z duplikującymi się wartościami?  
Jeśli na przykład wybierzesz szukanie duplikujących się wartości w polu Miasto, możesz tu wybrać pola NazwaKlienta i Adres.

Dostępne pola: IdWypozyczenia, DataWypozyczenia, Uwagi  
Dodatkowe pole kwerendy: zostawiamy puste

**Kreator kwerend wyszukiujących duplikaty**

Jaka ma być nazwa tworzonej kwerendy?  
Znajdź duplikaty dla: Transakcje wypożyczenia

Czy chcesz obejrzeć wyniki kwerendy, czy zmienić jej projekt?

Wyświetl wyniki  
 Modyfikuj projekt

Znajdź duplikaty dla: Transakcje wypożyczenia			
IdKlienta	Pole	IdPracownika	LiczbaPowtó
9	RZ01		2
12	RZ02		2
20	RZ01		2
25	RZ02		2
48	RZ02		2
49	RZ02		2
51	RZ03		2
57	RZ03		2

Przykład z kreatora wyszukiujących duplikaty pojawił się kiedyś na maturze- należało wyświetlić pary bliźniąt, miały one wspólną datę urodzenia i wspólny id\_matki.

### T32: Wybór tabel i pól do kwerendy. Określanie kryteriów kwerendy.<sup>1</sup>

Kryteria zapytania pomagają skoncentrować się na określonych elementach w bazie danych programu Access. Jeśli element pasuje do wszystkich wprowadzonych kryteriów, jest wyświetlany w wynikach zapytania.

Aby dodać kryteria do zapytania programu Access, otwórz je w widoku projektu i określ pola (kolumny), których mają dotyczyć kryteria. Jeśli pole nie znajduje się w siatce projektu, kliknij dwukrotnie dane pole, aby dodać je do siatki projektu, a następnie wprowadź kryterium w wierszu **Kryteria** dla tego pola.

Kryteria zapytania to wyrażenie, które program Access porównuje z wartościami w polu zapytania, aby określić, czy mają zostać uwzględnione rekordy zawierające poszczególne wartości. Na przykład = "Chicago" to wyrażenie, które program Access może porównać z wartościami w polu tekstowym w zapytaniu. Jeśli to pole w danym rekordzie ma wartość "Chicago", program Access uwzględni ten rekord w wynikach zapytania.

Poniżej przedstawiono kilka przykładów często używanych kryteriów, których można użyć jako punktu wyjściowego podczas tworzenia własnych kryteriów.

<sup>1</sup> <https://support.microsoft.com/pl-pl/office/przyk%C5%82ady-kryteri%C3%B3w-kwerend-3197228c-8684-4552-ac03-aba746fb29d8>

Kryteria	Opis
>25 and <50	To kryterium dotyczy pól typu Liczba, <b>wartość większą niż 25 i mniejszą niż 50.</b>
DateDiff ("rrrr"; [DataUrodzenia]; Date()) > 30	To kryterium dotyczy pól typu Data/Godzina, na przykład pola DataUrodzenia, <b>liczba lat między datą urodzenia danej osoby a datą bieżącą jest większa niż 30.</b>
Is Null	To kryterium można zastosować do pola dowolnego typu w celu pokazania rekordów, w których <b>pole ma wartość null.</b>

### Kryteria dla pól Tekst, Nota i Hiperłącze:

Uwzględniane rekordy	Kryterium	Wynik zapytania
Rekordy, które dokładnie pasują do określonej wartości, takiej jak Chiny.	"Chiny"	Zwraca rekordy, w których pole KrajRegion ma wartość Chiny.
Rekordy, które nie pasują do określonej wartości, takiej jak Meksyk.	Not "Meksyk"	Zwraca rekordy, w których pole KrajRegion ma wartość inną niż Meksyk.
Rekordy, które zaczynają się od określonego ciągu, takiego jak P.	Like P*	Zwraca rekordy wszystkich krajów/regionów, których nazwy zaczynają się od znaku „P”, na przykład Polska. <b>Uwaga:</b> Znak gwiazki (*) użyty w wyrażeniu reprezentuje dowolny ciąg znaków
Rekordy, które nie zaczynają się od określonego ciągu znaków, takiego jak P.	Not Like P*	Zwraca rekordy wszystkich krajów/regionów, których nazwy zaczynają się od znaku innego niż „P”.
Rekordy, które zawierają określony ciąg znaków, taki jak Korea.	Like "*Korea*"	Zwraca rekordy wszystkich krajów/regionów, które zawierają ciąg znaków „Korea”.
Rekordy, które nie zawierają określonego ciągu znaków, takiego jak Korea.	Not Like "*Korea*"	Zwraca rekordy wszystkich krajów/regionów, które nie zawierają ciągu znaków „Korea”.
Rekordy, które kończą się określonym ciągiem znaków, takim jak „yna”.	Like "*yna"	Zwraca rekordy wszystkich krajów/regionów, których nazwy kończą się na „yna”, na przykład Argentyna.
Rekordy, które nie kończą się określonym ciągiem znaków, takim jak „yna”.	Not Like "*yna"	Zwraca rekordy wszystkich krajów/regionów, których nazwy nie kończą się na „yna”, na przykład Argentyna.
Rekordy, które zawierają wartość null (lub nie zawierają żadnej wartości).	Is Null	Zwraca rekordy, które nie zawierają w polu żadnej wartości.
Rekordy, które nie zawierają wartości null.	Is Not Null	Zwraca rekordy, które zawierają w polu dowolną wartość.
Rekordy, które zawierają ciąg znaków o zerowej długości.	"" (para cudzysłówów)	Zwraca rekordy, w których pole jest puste (ale nie zawiera wartości null). Na przykład rekordy

		dotyczące sprzedaży do innego oddziału mogą zawierać puste pole KrajRegion.
Rekordy, które nie zawierają ciągu znaków o zerowej długości.	Not ""	Zwraca rekordy, w których pole KrajRegion nie jest puste.
Rekordy, które zawierają wartości null lub ciągu znaków o zerowej długości.	"" Or Is Null	Zwraca rekordy, w których w polu nie ma wartości lub ustawiono wartość oznaczającą pole puste.
Rekordy, które nie zawierają wartości oznaczającej pole puste ani nie są puste.	Is Not Null And Not ""	Zwraca rekordy, w których pole KrajRegion nie jest puste i nie zawiera wartości null.
Rekordy, które następują po określonej wartości (takiej jak Meksyk), gdy są posortowane w kolejności alfabetycznej.	>= "Meksyk"	Zwraca rekordy wszystkich krajów/regionów, poczynając od Meksyku, a kończąc na ostatniej literze alfabetu.
Rekordy, które zawierają się w określonym zakresie, na przykład od A do D.	Like "[A-D]*"	Zwraca rekordy krajów/regionów, których nazwy zaczynają się od liter od „A” do „D”.
Rekordy, które pasują do jednej z dwóch wartości, na przykład Polska lub USA.	"Polska" Or "USA"	Zwraca rekordy zawierające wartości Polska i USA.
Rekordy, które zawierają jedną spośród wymienionych wartości.	In("Francja"; "Chiny"; "Niemcy"; "Japonia")	Zwraca rekordy, które pasują do wszystkich krajów/regionów znajdujących się na liście.
Rekordy, które zawierają określone znaki na określonej pozycji w wartości pola.	Right([KrajRegion]; 1) = "y"	Zwraca rekordy wszystkich krajów/regionów, których nazwa kończy się literą „y”.
Rekordy, które spełniają wymagania dotyczące długości.	Len([KrajRegion]) > 10	Zwraca rekordy krajów/regionów, których długość nazwy przekracza 10 znaków.
Rekordy, które pasują do określonego wzoru.	Like "Chi??"	Zwraca rekordy pasujące do krajów/regionów, takich jak Chiny i Chile, których nazwy mają długość pięciu znaków, a pierwsze trzy z nich to „Chi”.  <b>Uwaga:</b> Znak zapytania (?) lub podkreślenia ( ) użyty w wyrażeniu reprezentuje pojedynczy znak— te znaki są również określane jako symbole wieloznaczne. Znaku _ nie można użyć w jednym wyrażeniu razem ze znakiem ?, nie można go też użyć w jednym wyrażeniu z symbolem wieloznacznym *. Symbolu wieloznacznego _ można użyć w wyrażeniu, w którym znajduje się symbol wieloznacznym %.

**Kryteria dla pól Data/godzina:**

<b>Uwzględniane rekordy</b>	<b>Kryterium</b>	<b>Wynik zapytania</b>
Rekordy, które dokładnie pasują do określonej wartości, takiej jak 2006-02-02.	#2006-02-02#	Zwraca rekordy transakcji, które miały miejsce 2 lutego 2006. Należy pamiętać, żeby otoczyć wartość daty znakami numeru #.
Rekordy, które nie pasują do określonej wartości, takiej jak 2006-02-02.	Not #2006-02-02#	Zwraca rekordy transakcji, które miały miejsce dnia innego niż 2 lutego 2006.
Rekordy, które zawierają wartości dat poprzedzających określoną datę, taką jak 2006-02-02.	< #2006-02-02#	Zwraca rekordy transakcji, które miały miejsce przed 2 lutego 2006.  Aby wyświetlić transakcje, które miały miejsce określonego dnia lub przed nim, należy skorzystać z operatora <= zamiast operatora <.
Rekordy, które zawierają wartości dat przypadających po określonej dacie, takiej jak 2006-02-02.	> #2006-02-02#	Zwraca rekordy transakcji, które miały miejsce po 2 lutym 2006.  Aby wyświetlić transakcje, które miały miejsce określonego dnia lub po nim, należy skorzystać z operatora >= zamiast operatora >.
Rekordy, które zawierają wartości z określonego zakresu dat	>#2006-02-02# and <#2006-02-04#	Zwraca rekordy transakcji, które miały miejsce między 2 lutym 2006 a 4 lutym 2006.  Do filtrowania zakresów dat (z datami krańcowymi włącznie) można również użyć operatora <b>Between</b> . Na przykład kryterium Between #2006-02-02# and #2006-02-04# ma takie samo znaczenie jak kryterium >=#2006-02-02# and <=#2006-02-04#.
Rekordy, które zawierają wartości spoza zakresu	<#2006-02-02# or >#2006-02-04#	Zwraca rekordy transakcji, które miały miejsce przed 2 lutym 2006 lub po 4 lutym 2006.
Rekordy, które zawierają jedną z dwóch wartości, takich jak 2006-02-02 lub 2006-02-03.	#2006-02-02# or #2006-02-03#	Zwraca rekordy transakcji, które miały miejsce 2 lutego 2006 lub 3 lutego 2006.
Elementy zawierające jedną z wielu wymienionych wartości.	In (#2006-02-01#, #2006-03-01#, #2006-04-01#)	Zwraca rekordy transakcji, które miały miejsce 1 lutego 2006, 1 marca 2006 lub 1 kwietnia 2006.
Rekordy, które zawierają datę	DatePart("m";[DataSprzedaż y]) = 12	Zwraca rekordy transakcji, które miały miejsce w grudniu dowolnego roku.

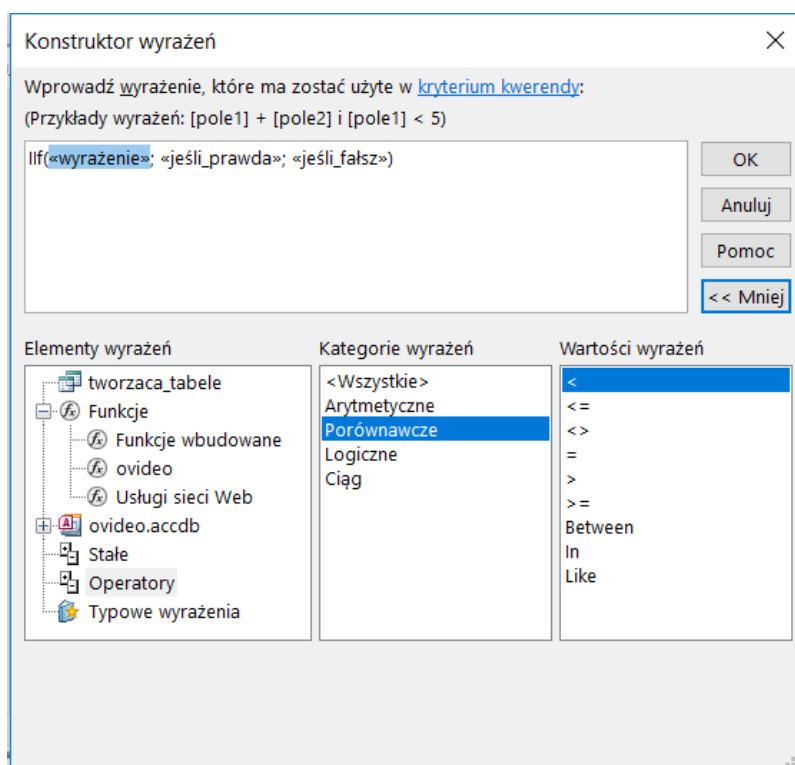
przypadającą w określonym miesiącu (niezależnie od roku), na przykład w grudniu.		
Elementy zawierające dzisiejszą datę.	Date()	Zwraca rekordy transakcji, które miały miejsce w dniu bieżącym. Jeśli dzisiejszą datą jest 2006-02-02, zostaną wyświetlone rekordy, w których pole DataZamówienia zawiera datę 2 lutego 2006.
Elementy zawierające wczorajszą datę.	Date()-1	Zwraca rekordy transakcji, które miały miejsce dzień przed dniem bieżącym. Jeśli dzisiejszą datą jest 2006-02-02, zostaną wyświetlone rekordy z dnia 1 lutego 2006.
Elementy zawierające jutrzejszą datę.	Date() + 1	Zwraca rekordy transakcji, które miały miejsce dzień po dniu bieżącym. Jeśli dzisiejszą datą jest 2006-02-02, zostaną wyświetlone rekordy z dnia 3 lutego 2006.
Rekordy, które zawierają datę przypadającą w bieżącym tygodniu.	DatePart("ww";[DataSprzedaży]) = DatePart("ww";Date()) and Year([DataSprzedaży]) = Year(Date())	Zwraca rekordy transakcji, które miały miejsce w bieżącym tygodniu. Tydzień rozpoczyna się w niedzielę i kończy w sobotę.
Rekordy, które zawierają datę przypadającą w zeszłym tygodniu.	Year([DataSprzedaży])* 53 + DatePart("ww";[DataSprzedaży]) = Year(Date())* 53 + DatePart("ww";Date()) - 1	Zwraca rekordy transakcji, które miały miejsce w zeszłym tygodniu. Tydzień rozpoczyna się w niedzielę i kończy w sobotę.
Rekordy, które zawierają datę przypadającą w przyszłym tygodniu.	Year([DataSprzedaży])* 53+DatePart("ww"; [DataSprzedaży]) = Year(Date())* 53+DatePart("ww"; Date()) + 1	Zwraca rekordy transakcji, które będą miały miejsce w przyszłym tygodniu. Tydzień rozpoczyna się w niedzielę i kończy w sobotę.
Rekordy, które zawierają datę przypadającą w ciągu ostatnich 7 dni.	Between Date() and Date()-6	Zwraca rekordy transakcji, które miały miejsce w ciągu ostatnich 7 dni. Jeśli dzisiejszą datą jest 2006-02-02, zostaną wyświetlone rekordy, których data należy do zakresu od 24 stycznia 2006 do 2 lutego 2006.
Rekordy, które zawierają datę przypadającą w bieżącym miesiącu.	Year([DataSprzedaży]) = Year(Now()) And Month([DataSprzedaży]) = Month(Now())	Zwraca rekordy z bieżącego miesiąca. Jeśli dzisiejszą datą jest 2006-02-02, zostaną wyświetlone rekordy z lutego 2006.
Rekordy, które zawierają datę przypadającą w zeszłym miesiącu.	Year([DataSprzedaży])* 12 + DatePart("m";[DataSprzedaży]) = Year(Date())* 12 + DatePart("m";Date()) - 1	Zwraca rekordy z zeszłego miesiąca. Jeśli dzisiejszą datą jest 2006-02-02, zostaną wyświetlone rekordy ze stycznia 2006.

Rekordy, które zawierają datę przypadającą w przyszłym miesiącu.	$\text{Year}([\text{DataSprzedaży}]) * 12 + \text{DatePart}("m"; [\text{DataSprzedaży}]) = \text{Year}(\text{Date}()) * 12 + \text{DatePart}("m"; \text{Date}()) + 1$	Zwraca rekordy z przyszłego miesiąca. Jeśli dzisiejszą datą jest 2006-02-02, zostaną wyświetlone rekordy z marca 2006.
Rekordy, które zawierają datę przypadającą w ciągu ostatnich 30 lub 31 dni.	$\text{Between Date}() \text{ And } \text{DateAdd}("M"; -1; \text{Date}())$	Rekordy zawierające wartości miesięcznej sprzedaży. Jeśli dzisiejszą datą jest 2006-02-02, zostaną wyświetlone rekordy z okresu od 2 stycznia 2006. do 2 lutego 2006
Rekordy, które zawierają datę przypadającą w bieżącym kwartale.	$\text{Year}([\text{DataSprzedaży}]) = \text{Year}(\text{Now}()) \text{ And } \text{DatePart}("q"; \text{Date}()) = \text{DatePart}("q"; \text{Now}())$	Zwraca rekordy transakcji z bieżącego kwartału. Jeśli dzisiejszą datą jest 2006-02-02, zostaną wyświetlone rekordy z pierwszego kwartału 2006.
Rekordy, które zawierają datę przypadającą w zeszłym kwartale.	$\text{Year}([\text{DataSprzedaży}] * 4 + \text{DatePart}("q"; [\text{DataSprzedaży}])) = \text{Year}(\text{Date}()) * 4 + \text{DatePart}("q"; \text{Date}()) - 1$	Zwraca rekordy z zeszłego kwartału. Jeśli dzisiejszą datą jest 2006-02-02, zostaną wyświetlone rekordy z ostatniego kwartału 2005.
Rekordy, które zawierają datę przypadającą w przyszłym kwartale.	$\text{Year}([\text{DataSprzedaży}] * 4 + \text{DatePart}("q"; [\text{DataSprzedaży}])) = \text{Year}(\text{Date}()) * 4 + \text{DatePart}("q"; \text{Date}()) + 1$	Zwraca rekordy z przyszłego kwartału. Jeśli dzisiejszą datą jest 2006-02-02, zostaną wyświetlone rekordy z drugiego kwartału 2006.
Rekordy, które zawierają datę przypadającą w bieżącym roku.	$\text{Year}([\text{DataSprzedaży}]) = \text{Year}(\text{Date}())$	Zwraca rekordy z bieżącego roku. Jeśli dzisiejszą datą jest 2006-02-02, zostaną wyświetlone rekordy z roku 2006.
Rekordy, które zawierają datę przypadającą w zeszłym roku.	$\text{Year}([\text{DataSprzedaży}]) = \text{Year}(\text{Date}()) - 1$	Zwraca rekordy transakcji, które miały miejsce w zeszłym roku. Jeśli dzisiejsza datą jest 2006-02-02, zostaną wyświetlone rekordy z roku 2005.
Rekordy, które zawierają datę przypadającą w przyszłym roku.	$\text{Year}([\text{DataSprzedaży}]) = \text{Year}(\text{Date}()) + 1$	Zwraca rekordy transakcji, które zawierają datę przypadającą w przyszłym roku. Jeśli dzisiejsza datą jest 2006-02-02, zostaną wyświetlone rekordy z roku 2007.
Rekordy, które zawierają datę należącą do zakresu od 1 stycznia do dnia dzisiejszego	$\text{Year}([\text{DataSprzedaży}]) = \text{Year}(\text{Date}()) \text{ and } \text{Month}([\text{DataSprzedaży}]) \leq \text{Month}(\text{Date}()) \text{ and } \text{Day}([\text{DataSprzedaży}]) \leq \text{Day}(\text{Date}())$	Zwraca rekordy transakcji, których daty należą do zakresu od 1 stycznia bieżącego roku do dnia dzisiejszego. Jeśli dzisiejszą datą jest 2006-02-02, zostaną wyświetlone rekordy z okresu od 1 stycznia 2006 do 2 lutego 2006.
Elementy, które zawierają minioną datę.	$< \text{Date}()$	Zwraca rekordy transakcji, które miały miejsce przed dniem dzisiejszym.
Rekordy, które zawierają przyszłą datę.	$> \text{Date}()$	Zwraca rekordy transakcji, które będą miały miejsce po dniu dzisiejszym.
Elementy zawierające wartość null	$\text{Is Null}$	Zwraca rekordy, w których brakuje daty transakcji.

Elementy zawierające wartości inne niż null.	Is Not Null	Zwraca rekordy, w których data transakcji jest podana.
--	-------------	--

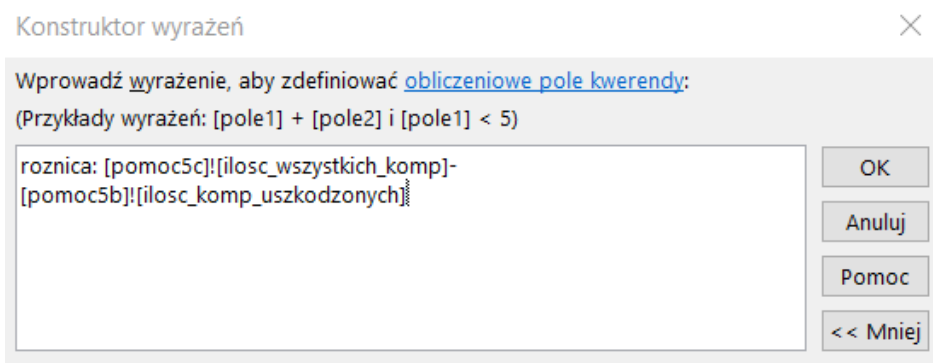
### T33: Konstruktor wyrażeń.

Niekiedy napisanie wyrażenia może być dużym wyzwaniem. Jednak Konstruktor wyrażeń znacznie ułatwia pracę. Wyrażenia składają się z wielu elementów lub „ruchomych części”: funkcji, operatorów, stałych, identyfikatorów i wartości. Za pomocą Konstruktora wyrażeń możesz szybko wyszukać te składniki i wstawić je w odpowiedni sposób. Z Konstruktora wyrażeń możesz korzystać na dwa sposoby. Być może wystarczy, że skorzystasz z pola Konstruktora wyrażeń, ale gdy wyrażenie jest bardziej złożone, możesz użyć rozwiniętego Konstruktora wyrażeń. Dzięki polu Konstruktora wyrażeń możesz szybciej i dokładniej tworzyć wyrażenia przy użyciu inteligentnych narzędzi i informacji kontekstowych.



**Porada** Aby korzystać z gotowych wyrażeń, takich jak wyświetlanie numerów stron, bieżącej daty oraz bieżącej daty i godziny, wybierz pozycję **Typowe wyrażenia**.

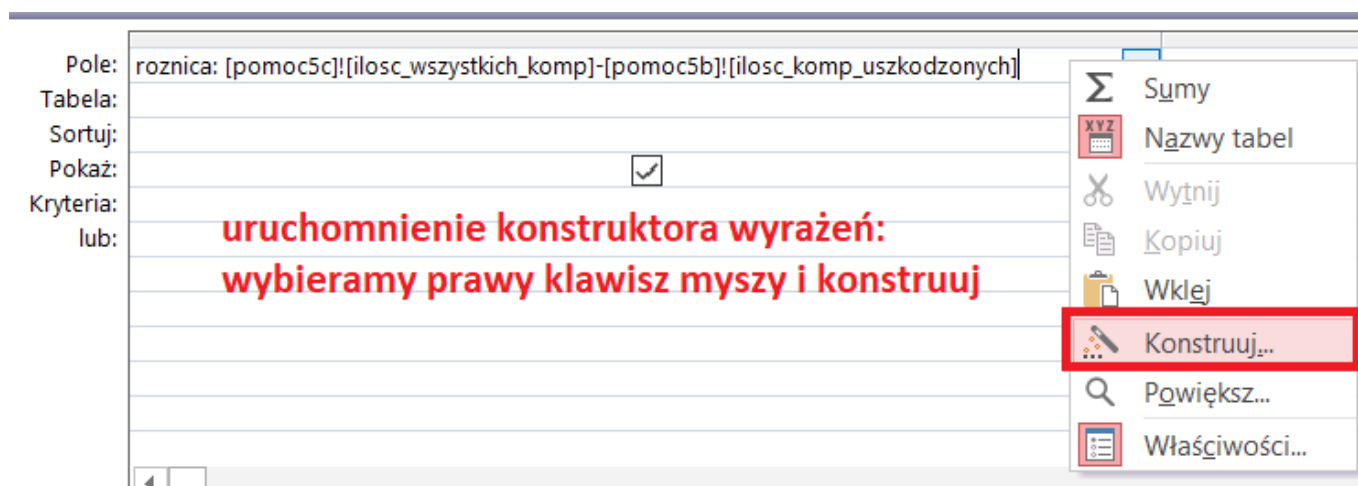
Za pomocą konstruktora można również dokonywać obliczeń i pisać formuły, które nas interesują-  
wyklikując- a nie pisząc- a to nam gwarantuje, że nie zrobimy literówki, np. jak na poniższym zdjęciu.



Istnieje wiele różnych wyrażeń, np. widoczne w poniższej tabeli, które można wpisać samemu ręcznie lub skorzystać z konstruktora wyrażeń.

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 Sales.
=Sum([Ilość] * [Cena])	Używa funkcji Sum w celu wyświetlenia sumy iloczynów wartości kontrolki Quantity i Price.
=([Sprzedaż]/Sum([Sprzedaż])*100	Wyświetla obrót procentowy wyrażony stosunkiem wartości kontrolki Sprzedaż do sumy wszystkich wartości kontrolki Sprzedaż.  Jeśli właściwość Format kontrolki jest ustawiona na Procent, w wyrażeniu nie należy umieszczać elementu *100.

Uruchomienie konstruktora podczas pisania kwerend w widoku projektu:



**Zadanie:** Na podstawie bazy ovideo i konstruktora wyświetl:

1. Imię i nazwisko pracownika wraz z pensją (czas pracy \* stawka)
2. Imię i nazwisko pracownika wraz z wiekiem (policzyć różnicę lat)

### T34: Kwerendy wybierające.

**Kwerendy wybierające** - kwerenda wybierająca jest najczęściej używanym rodzajem kwerendy. Służy do otrzymywania danych z tabeli lub tabel i wyświetla wyniki w arkuszu danych, w którym można je następnie aktualizować (z pewnymi ograniczeniami). Kwerendy wybierające mogą być również używane do grupowania rekordów i obliczania sum, zliczeń, wyliczania średnich i przeprowadzania innych obliczeń.

**Ćwiczenie 1:** Utwórz kwerendę wybierającą, która wyświetli oceny z przedmiotu j. polski zgodnie z poniższym schematem: imie, nazwisko, nazwa przedmiotu i ocena (na nazwij Oceny z polskiego). Zacznij wyświetlanie od najlepszych do najgorszych ocen.

Pole:	imie	nazwisko	nazwa	ocena
Tabela:	uczniowie	uczniowie	przedmioty	oceny
Sortuj:				Malejąco
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kryteria:			"j.polski"	
lub:				

**Ćwiczenie 2 :** Wyświetl imie i nazwisko ucznia, nazwę przedmiotu z którego otrzymał ocenę niedostateczną. Posortuj po nazwisku. Nazwij tę kwerendę oceny niedostateczne.

Pole:	imie	nazwisko	nazwa	ocena
Tabela:	uczniowie	uczniowie	przedmioty	oceny
Sortuj:		Rosnąco		
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kryteria:				"1"
lub:				

**Ćwiczenie 3 :** Utwórz zestawienie w którym podasz ile jest ocen niedostatecznych z poszczególnych przedmiotów. Nazwij tą kwerendę zestawienie ocen niedostatecznych. Wykorzystaj opcję grupowania.

Pole:	nazwa	ocena	ocena
Tabela:	przedmioty	oceny	oceny
Suma:	Grupuj według	Policz	Gdzie
Sortuj:		Malejąco	
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Kryteria:			"1"
lub:			

### Kwerendy parametryczne

Wykonanie kwerendy parametrycznej powoduje wyświetlenie okna dialogowego, w którym należy podać informacje, takie jak kryteria pobierania rekordów czy wartości, które mają być wstawione w polach. Kwerendę można tak zaprojektować, aby użytkownik był pytany o kilka różnych informacji, na przykład o dwie daty.

**Ćwiczenie 4** Utwórz kwerendę parametryczną, która „zapyta” o nazwę przedmiotu, a po wpisaniu nazwy wyświetli oceny z tego przedmiotu zgodnie z poniższym schematem (nazwij ją Oceny z danego przedmiotu). Imie, nazwisko, nazwa przedmiotu, ocena. Aby utworzyć kwerendę parametryczną należy w kwerendzie wybierającej wywołać okno dialogowe z pytaniem - robimy to nawiasem [ ] który umieszczamy w wierszu KRYTERIA bądź LUB kwerendy. W nawiasie tym możemy wpisać treść komunikatu, który będzie wyświetlony w oknie.

Pole:	imie	nazwisko	nazwa	ocena
Tabela:	uczniowie	uczniowie	przedmioty	oceny
Sortuj:				Malejąco
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kryteria:			[podaj nazwę przedmiotu]	
lub:				

Uwaga! Instrukcja LIKE [Wpisz nazwę przedmiotu] & \* wprowadzona w polu Przedmiot umożliwi wyszukiwać przedmioty podając część początku ich nazwy.

Pole:	imie	nazwisko	nazwa	ocena
Tabela:	uczniowie	uczniowie	przedmioty	oceny
Sortuj:				Malejąco
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kryteria:			Like [podaj nazwę przedmiotu] & "**	
lub:				

**Ćwiczenie 5.** Utwórz kwerendę parametryczną, która wyświetli dwa okna dialogowe w których podamy dwie oceny (ocenaA i ocenaB) a następnie wyświetli oceny z przedziału (ocenaA do ocenaB) zgodnie z poniższym schematem (nazwij ją Przedział ocen).

Pole:	imie	nazwisko	nazwa	ocena
Tabela:	uczniowie	uczniowie	przedmioty	oceny
Sortuj:				
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kryteria:				<=[podaj ocenę maksymalną] And >=[podaj ocenę minimalną]
lub:				

**Zadanie: 1.** Utwórz zapytanie wypisujące imię i nazwisko ucznia, których wychowawcą jest pan Krystian Salwacki

**Zadanie: 2.** Utwórz kwerendę parametryczną, która prosi o padanie nazwiska oraz imienia wychowawcy a następnie wypisze imię i nazwisko wszystkich uczniów, którymi opiekuje się podana osoba.

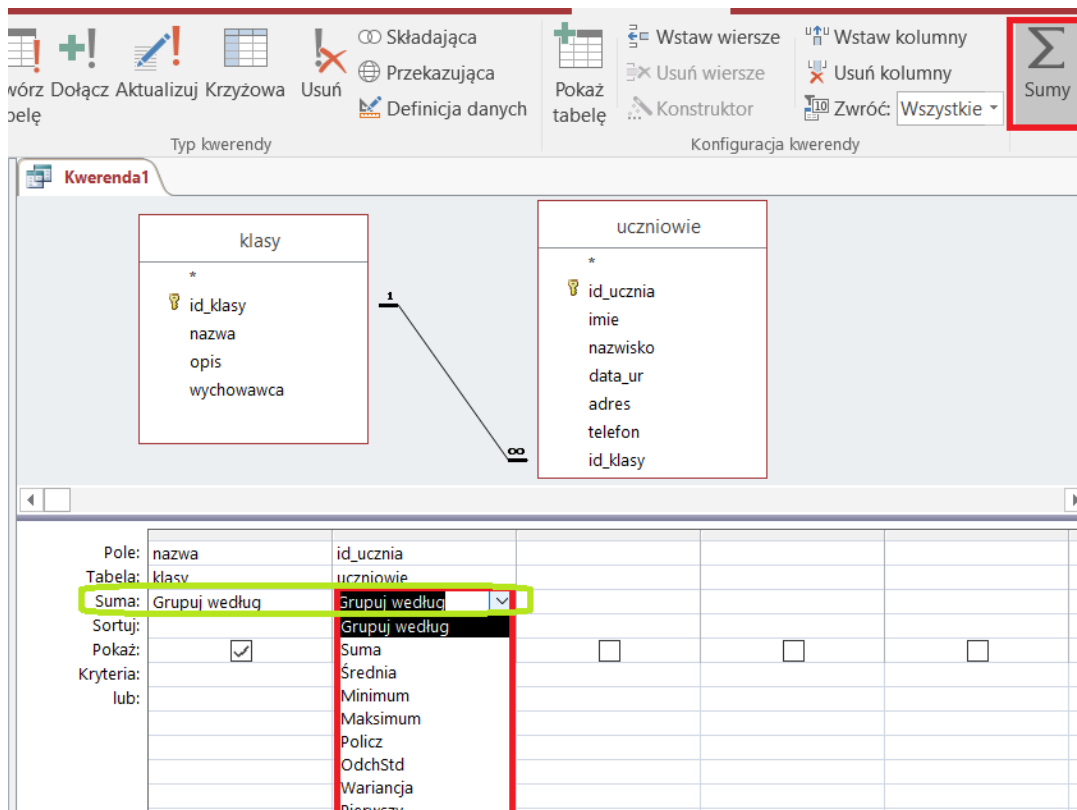
## T35: Funkcje grupujące.

### Opis sposobów zliczania danych

Liczbę elementów w polu (kolumnie wartości) można zliczyć, używając funkcji **Zlicz**. Funkcja **Zlicz** należy do zestawu funkcji nazywanych funkcjami agregującymi. Funkcje agregujące służą do wykonywania obliczeń na kolumnie danych i zwracania jednej wartości. Oprócz funkcji **Zlicz** program Access udostępnia szereg funkcji agregujących, takich jak:

- **Suma** umożliwiająca sumowanie kolumny liczb.
- **Średnia** umożliwiająca obliczanie średniej kolumny liczb.
- **Maksimum** umożliwiająca znalezienie największej wartości w polu.
- **Minimum** umożliwiająca znalezienie najmniejszej wartości w polu.
- **Odchylenie standardowe** umożliwiająca mierzenie szerokości rozproszenia wartości od wartości średniej.
- **Wariancja** umożliwiająca mierzenie statystycznej wariancji wszystkich wartości w kolumnie.

Otwórz zapytanie w widoku arkusza danych, a następnie dodaj wiersz sumy. Wiersz Suma umożliwia użycie funkcji agregującej w jednej lub większej liczbie kolumn zestawu wyników zapytania bez konieczności zmieniania projektu zapytania.



**Ćwiczenie:** Ile jest uczniów w poszczególnych klasach? Grupujemy po nazwie klasy, a po id\_ucznia zliczamy rekordy.

Pole:	nazwa	id_ucznia
Tabela:	klasy	uczniowie
Suma:	Grupuj według	Policz
Sortuj:	Rosnąco	
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kryteria:		

**Ćwiczenie:** Baza ovideo. Wyświetl: minimalną, maksymalną i średnią stawkę pracownika w danym oddziale. Oddział staje się grupą a stawkę wyciągamy 3 razy i odpowiednio ustawiamy: min,max i avg.

The screenshot shows the Microsoft Access interface. The 'Kwerendy' (Queries) ribbon is active, with the 'Sumy' (Totals) group highlighted. Below the ribbon, the 'Kwerenda1' query design grid is shown. The grid has the following structure:

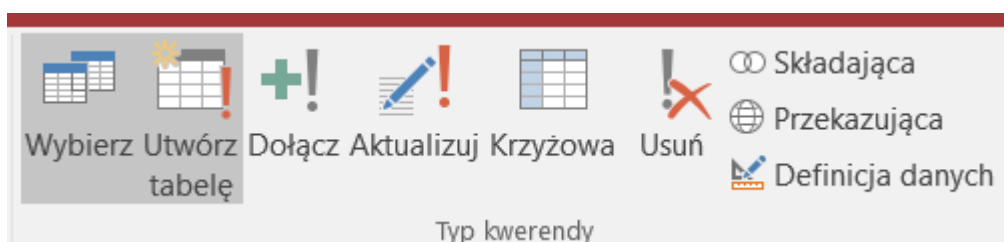
Pole:	Stawka	Stawka	Stawka	NazwaOddziału
Tabela:	Pracownicy	Pracownicy	Pracownicy	Pracownicy
Suma:	Minimum	Maksimum	Srednia	Grupuj według
Sortuj:				
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kryteria:				
lub:				

Below the design grid, the resulting query results are displayed in a table:

MinimumOfs	MaksimumO	ŚredniaOfSta	NazwaOddziału
12,50 zł	24,38 zł	17,03 zł	DĘBICA
12,50 zł	20,00 zł	16,46 zł	KOLBUSZOWA
16,88 zł	40,63 zł	23,55 zł	MIELEC
15,63 zł	27,50 zł	18,88 zł	ROPCZYCE
12,50 zł	30,00 zł	20,36 zł	RZESZÓW
19,38 zł	26,88 zł	21,88 zł	ST. WOLA
12,50 zł	33,13 zł	22,58 zł	TARNÓW

### T36: Kwerendy funkcjonalne.


Oprócz kwerend wybierających można tworzyć kwerendy funkcjonalne.



Przy ikonach czterech typów kwerend widoczny jest wykrzyknik, co oznacza, że wykonanie tych kwerend skutkuje trwałymi zmianami w bazie danych. Każdorazowe ich uruchomienie może powodować przeliczanie wartości lub usuwanie danych, jeśli spełnione zostaną określone kryteria. Trzeba więc zachować szczególną ostrożność. Należy również czytać komunikaty pojawiające się na ekranie przy próbie uruchomienia kwerend funkcjonalnych. Warto też wykonać kopię bazy lub przynajmniej tabel, aby nie stracić cennych danych.

**Kwerenda funkcjonalna** pozwala na wprowadzenie zmian w wielu rekordach lub przeniesienie wielu rekordów przy użyciu pojedynczej operacji. Istnieją cztery rodzaje kwerend funkcjonalnych:

Przykłady będą oparte o bazę danych ovideo.

- Kwerendy aktualizujące**- kwerenda aktualizująca dokonuje globalnych zmian w grupie rekordów w jednej lub wielu tabelach. Za pomocą kwerend aktualizujących można zmieniać dane w istniejących tabelach. Ponieważ każde uruchomienie kwerendy aktualizującej oznacza przeliczenie wartości, trzeba być bardzo ostrożnym przy jej uruchamianiu. To właśnie dlatego kwerendy tego typu mają ikonę opatrzoną wykrzyknikiem symbolizującym niebezpieczeństwo.
 

Aktualizuj

**Przykład:** Zwiększenie wszystkim pracownikom stawki o 25%

Pole:	Stawka
Tabela:	Pracownicy
Aktualizacja do:	[stawka]*1,25
Kryteria:	
lub:	

Widok sql: UPDATE Pracownicy SET Pracownicy.Stawka = [stawka]\*1.25;

Fragment danych przed aktualizacją:

IdPracownika	Nazwisko	Imie	NazwaOddzialu	DataUrodzenia	CzasPracy	Stawka	Ubezpieczen
DB01	Stanisz	Klaudia	DĘBICA	20-maj-90	80	15,00 zł	<input checked="" type="checkbox"/>
DB02	Janik	Joanna	DĘBICA	07-cze-87	160	10,00 zł	<input checked="" type="checkbox"/>
DB03	Bazan	Alicja	DĘBICA	01-wrz-89	120	10,00 zł	<input checked="" type="checkbox"/>
DB04	Czekaj	Bogdan	DĘBICA	08-mar-80	160	19,50 zł	<input type="checkbox"/>
KO01	Grzesik	Jolanta	KOLBUSZOWA	01-wrz-87	120	16,00 zł	<input checked="" type="checkbox"/>

Po aktualizacji:


IdPracownika	Nazwisko	Imie	NazwaOddzialu	DataUrodzenia	CzasPracy	Stawka	Ubezpieczen
DB01	Stanisz	Klaudia	DĘBICA	20-maj-90	80	18,75 zł	<input checked="" type="checkbox"/>
DB02	Janik	Joanna	DĘBICA	07-cze-87	160	12,50 zł	<input checked="" type="checkbox"/>
DB03	Bazan	Alicja	DĘBICA	01-wrz-89	120	12,50 zł	<input checked="" type="checkbox"/>
DB04	Czekaj	Bogdan	DĘBICA	08-mar-80	160	24,38 zł	<input type="checkbox"/>
KO01	Grzesik	Jolanta	KOLBUSZOWA	01-wrz-87	120	20,00 zł	<input checked="" type="checkbox"/>

Uwaga! Każdorazowe uruchomienie tej kwerendy spowoduje kolejne przeliczenie stawki spełniających kryterium. Dlatego uruchomienie kwerendy aktualizującej jest poprzedzone ostrzeżeniem i zapytaniem, czy na pewno chcesz zaktualizować dane.


Microsoft Access

×

**Masz zamiar uruchomić kwerendę aktualizującą, która zmodyfikuje dane w tabeli.**

 Czy na pewno chcesz uruchomić tę kwerendę funkcjonalną?  
Aby uzyskać informacje, jak zapobiec dalszemu wyświetlaniu tego komunikatu przy każdym uruchamianiu kwerendy funkcjonalnej, kliknij przycisk Pomoc.

---

- Kwerendy tworzące tabele** -kwerenda tworząca tabele tworzy nową tabelę na podstawie wszystkich lub części danych przechowywanych w jednej lub wielu tabelach. Kwerendy tworzące tabele są pomocne przy tworzeniu tabel, które mają zostać wyeksportowane do innych baz danych programu Microsoft Access lub tabel historii zawierających stare rekordy.
 

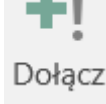
z Utwórz |  
tabelę

Rozwiązanie zadania tego typu przebiega w dwóch etapach. Etap pierwszy to utworzenie kwerendy wybierającej, a następnie przekształcenie jej w kwerendę tworzącą nową tabelę. Po podaniu nazwy powstałej tabeli w zakładce z tabelami pojawi się nowy zestaw danych- wybrani\_klienci - nowa tabela.

**Przykład:** Utworzy nową tabelę o nazwie wybrani\_klienci z Rzeszowa, Mielca lub Dębicy.

The screenshot shows the Microsoft Access interface. The 'Projektowanie' (Design) ribbon is active, with the 'Utwórz tabelę' (Create Table) button highlighted. A dialog box titled 'Tworzenie tabeli' (Create Table) is open, showing 'Nazwa tabeli: Wybrani\_klienci' and the option 'Bieżąca baza danych' (Current database) selected. Below the dialog, a table design grid is visible for the 'Wybrani\_klienci' table. The grid shows fields: Imię, Nazwisko, E-mail, Telefon, Miasto, KodPocztowy, DataPrzystapienia, and NrDowoduOsobistego. The 'Miasto' field has a filter criterion 'In ("Rzeszów";"Mielec";"Dębica")'. A warning message from Microsoft Access is displayed at the bottom, asking for confirmation to create the table with 30 records.

Po wykonaniu pojawiła się nowa tabela:

- **Kwerendy dołączające** - kwerenda dołączająca dodaje na końcu jednej lub wielu tabel grupę rekordów z jednej lub wielu tabel. Przypuśćmy na przykład, że firma pozyskała nowych klientów i istnieje baza danych zawierająca tabelę z informacjami o tych klientach. Aby uniknąć wpisywania wszystkich tych informacji w bazie danych, można dane te dołączyć do tabeli Klienci. Kwerenda dołączająca dodaje na koniec tabeli rekordy pochodzące z innej tabeli (lub kilku tabel). Jest to kwerenda z grupy kwerend funkcjonalnych. Jej ikona również jest wyróżniona znakiem wykrzyknika. Każdorazowe uruchomienie kwerendy dołączającej; będzie próbą ponownego dodania rekordów do tabeli.
 

**Przykład:** Przed wykonaniem tej kwerendy utworzymy dwie nowe tabele z następującymi rekordami (rekordy mogą być również dowolne):

id_uczen	imie	nazwisko	nazwa_klasy
1	Piotr	Nowak	1d
2	Jan	Kubica	3c
3	Piotr	Żyła	1g
*	(Nowy)		

Rysunek 6. Tabela-kandydaci na uczniów

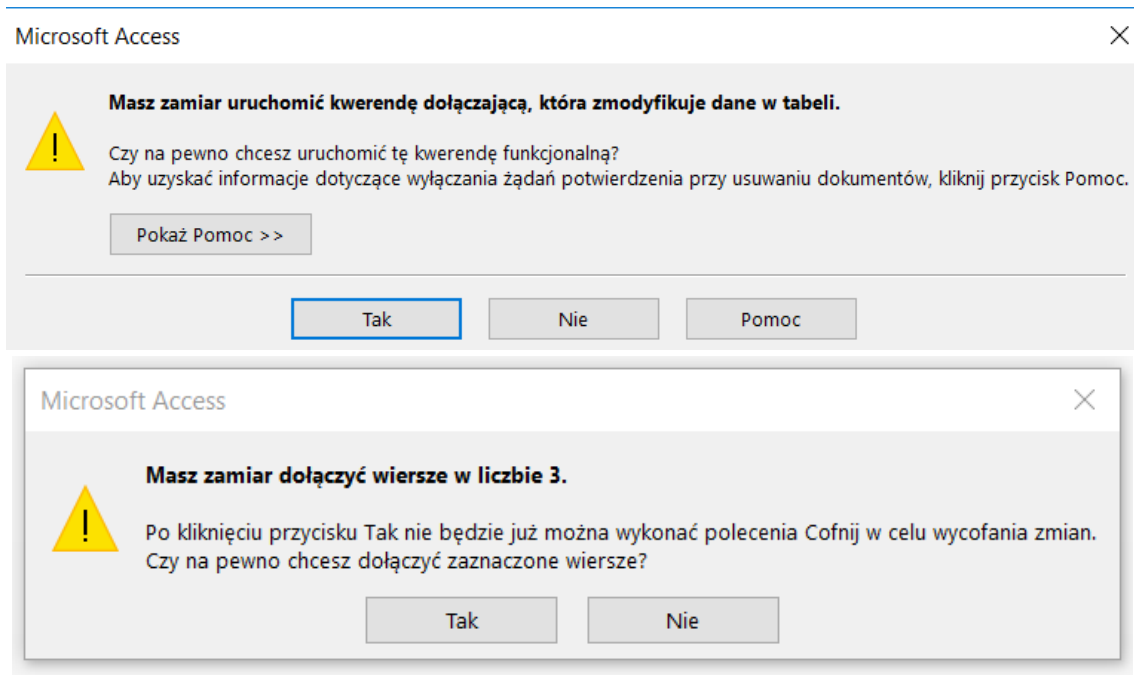
id_uczen	imie	nazwisko	nazwa_klasy
1	Anna	Kowalska	2b
2	Janina	Kowalska	3c
*	(Nowy)		

Rysunek 7. Tabela uczniowie

Do tabeli uczniów dołączę dane z tabeli kandydaci\_na\_uczniów. Pierwsze wybieramy tabelę, którą chcemy dołączyć- czyli kandydaci\_na\_uczniów. Wybierz pola, które chcesz dołączyć: imie, nazwisko, nazwa\_klasy. Teraz zmień typ kwerendy z wybierającej na dołączającą. Wybierz z listy lub wpisz nazwę tabeli, do której mają być dołączone rekordy po uruchomieniu kwerendy. Utworzoną kwerendę zapisz pod nazwą dolaczajaca\_test. Teraz uruchom kwerendę dołączającą. W odpowiedzi na komunikat o próbie dodania rekordów do tabeli potwierdź chęć ich dodania.

The screenshot shows the Microsoft Access interface. In the ribbon, the 'Dołącz' (Join) button is highlighted with a red box. Below the ribbon, the 'Dołączanie' (Join) dialog box is open and also highlighted with a red box. The dialog shows 'Dołącz do' (Join to) set to 'uczniowie'. The 'Bieżąca baza danych' (Current database) radio button is selected. Below the dialog, a table grid shows the join configuration:

Pole:	imie	nazwisko	nazwa_klasy
Tabela:	kandydaci_na_uczniow	kandydaci_na_uczniow	kandydaci_na_uczniow
Sortuj:			
Dołączanie do:	imie	nazwisko	nazwa_klasy
Kryteria:			
lub:			



Dołączone rekordy w tabeli uczniowie:

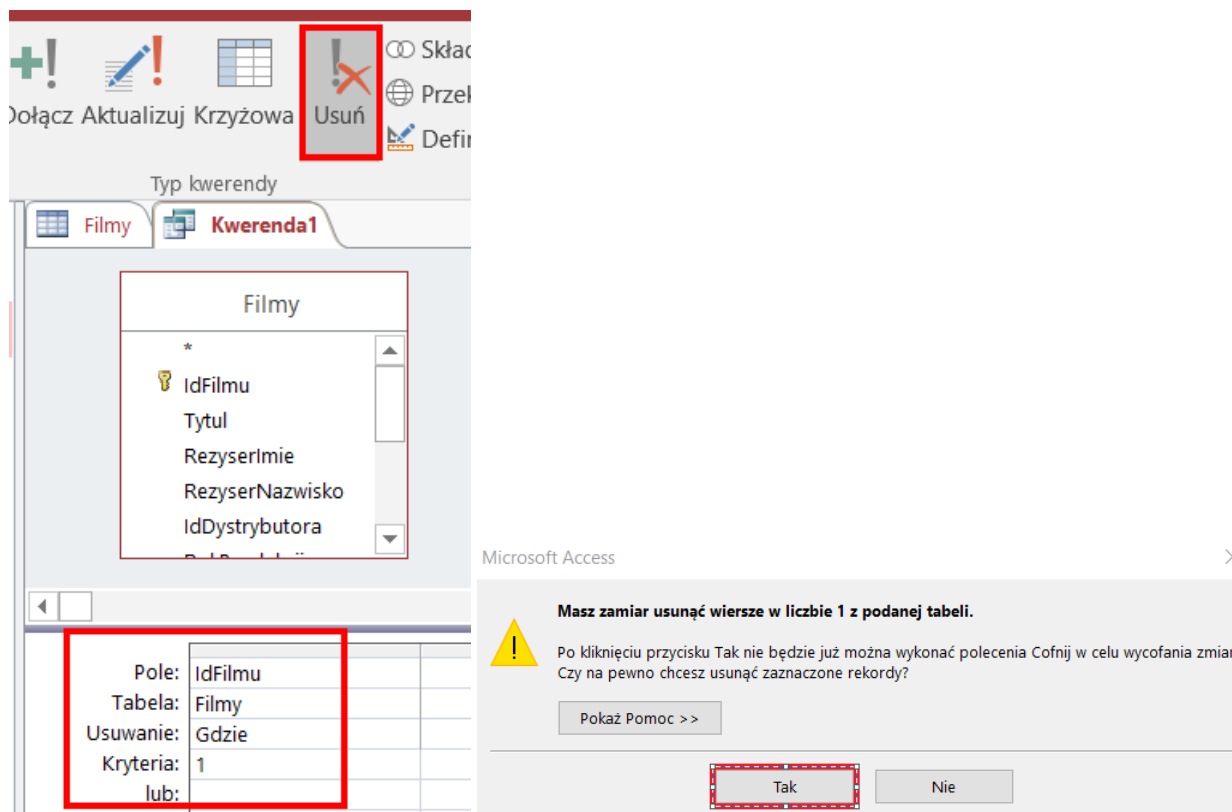
id_uczen	imie	nazwisko	nazwa_klasy
1	Anna	Kowalska	2b
2	Janina	Kowalska	3c
3	Piotr	Żyła	1g
4	Piotr	Nowak	1d
5	Jan	Kubica	3c
6	Piotr	Żyła	1g

- Kwerendy usuwające**- kwerenda usuwająca usuwa grupę rekordów z jednej lub wielu tabel. Użycie kwerendy usuwającej powoduje usunięcie całych rekordów, nie zaś wybranych pól w rekordach. Jeśli rekordy zostaną usunięte za pomocą kwerendy usuwającej, która usuwa z jednej tabeli lub z kilku tabel wiersze spełniające określone kryteria, czynności tej nie można cofnąć.

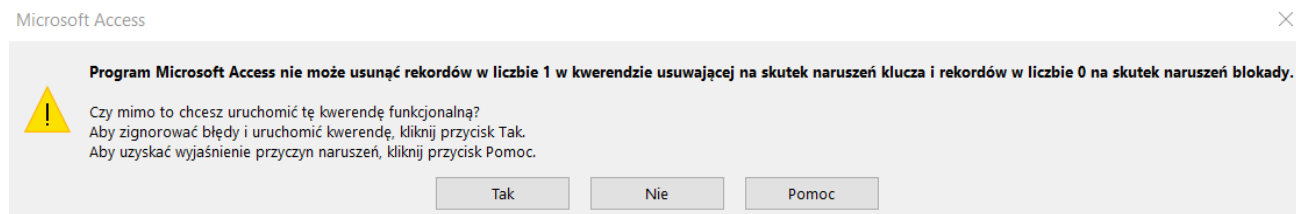


W niektórych przypadkach uruchomienie kwerendy usuwającej może spowodować usunięcie rekordów z tabel związanych relacją, nawet jeśli nie zostały one uwzględnione w kwerendzie. Może się tak stać, jeśli tabela znajdująca się w kwerendzie jest stroną „jeden” relacji jeden-do-wielu i dla tej relacji została włączona opcja usuwania kaskadowego. Wówczas, usuwając rekordy w tabeli po stronie „jeden” relacji, usuwa się również rekordy z tabeli po stronie „wiele”.

**Przykład:** Usunięcie filmu o id=1;



Poniższy komunikat mówi o tym, że nie można usunąć tego rekordu, ponieważ istnieje powiązanie w innej tabeli, czyli taki film znajduje się w tabeli pozycje\_transakcji.



Można usunąć taki rekord, jeśli ustawimy kaskadowo usuń rekord pokrewny przy edycji relacji. Jeśli to ustawimy usunie się film o id=1 i w tabeli pozycje transakcji również zostanie usunięta pozycja z filmu o id=1.

Jeśli nie chcemy przestawiać kaskadowo usuń rekord pokrewny można wyświetlić np. tytuły filmów, które nie były wypożyczane i dopiero usunąć przykładowy film.

### T37: Budowanie kwerend krzyżowych.

**Kwerendy krzyżowe** stosowane są w celu obliczenia i zmiany struktury danych, co ułatwia analizowanie tych danych. Kwerendy krzyżowe obliczają sumę, średnią, zliczają dane lub przeprowadzają inny rodzaj podsumowania dla danych zgrupowanych według dwóch typów informacji—jeden wzdłuż lewej, a drugi wzdłuż górnej krawędzi arkusza danych.

**Przykład** na bazie ovideo: Wyświetl ile było wypożyczeń danego gatunku w danym mieście klienta. Na przecięciu musimy policzyć wartość: Policz, Gatunek i Miasto –ustalamy wierszem i kolumną, nie ma znaczenia co jest wierszem a co kolumną, chyba, że jest to sprecyzowane w pytaniu.

## Access -kwerendy

Typ kwerendy: **Krzyżowa**

Konfiguracja kwerendy: Wstaw wiersze, Wstaw kolumny, Usuń wiersze, Usuń kolumny, Konstruktor, Zwróć: [ ]

Pokazywanie/ukrywanie: Sumy, Parametry, Arkusz właściwości, Nazwy tabel

Typ kwerendy: **Krzyżowa**

Typ kwerendy: **Krzyżowa**

Pole: Gatunek, Miasto, IdWypożyczenia  
Tabela: Filmy, Klienci, Transakcje wypożyczenia  
Suma: Grupuj według, Grupuj według, Policz  
Krzyżowe: Nagłówek wiersza, Nagłówek kolumny, Wartość  
Sortuj:

Gatunek	Albigowa	Boguchw	Bratkov	Dębica	Głogów Młp	Jasło	Kolbuszow	Krosno	Łańcut	Mielec	Nisko	Ropczy	Rzeszów
Biograficzny	1						1	2	2	2			2
Dramat	2	1		6	4	3		4	5	4	2	2	1
Fantasy					1			1					
Horror													
Komedia	1			1	4			2	2		1	1	2
Kostiumowy				1	1	1				1			1
Kryminał									3	1	4		1
Melodramat				2	1			1					1

Rysunek 8. Fragment odpowiedzi na powyższy przykład.

**Ćwiczenie 6.** Utwórz kwerendę krzyżową, która policzy ile jest uczniów w poszczególnych klasach. Kwerendę nazwij Ilość uczniów.

## Access -kwerendy

Typ kwerendy

klasy

- \* id\_klasy
- nazwa
- opis
- wychowawca

uczniowie

- \* id\_ucznia
- imie
- nazwisko
- data\_ur
- adres
- telefon
- id\_klasy

Pole:	nazwa	id_ucznia
Tabela:	klasy	uczniowie
Suma:	Grupuj według	Policz
Krzyżowe:	Nagłówek wiersza	Wartość
Sortuj:		

nazwa	1a1	1b1	1c1	1d1	1e1	1f1	1g1
1a1	26						
1b1		29					
1c1			27				
1d1				27			
1e1					30		
1f1						27	
1g1							29

Rysunek 9. Fragment odpowiedzi na ćw.6.

**Ćwiczenie 7.** Utwórz kwerendę krzyżową, która policzy ile jest poszczególnych ocen z wszystkich przedmiotów. Kwerendę nazwij Ilość ocen.

oceny

- \* id\_oceny
- ocena
- id\_ucznia
- id\_przedmiotu

przedmioty

- \* id\_przedmiotu
- nazwa
- opis

Pole:	nazwa	ocena	id_oceny
Tabela:	przedmioty	oceny	oceny
Suma:	Grupuj według	Grupuj według	Policz
Krzyżowe:	Nagłówek wiersza	Nagłówek kolumny	Wartość
Sortuj:			

nazwa	-6	-5	-4	-3	-2	1	2
biologia	35	104	186	131	55	67	222
chemia	45	129	180	142	60	63	226
fizyka	39	121	176	142	70	68	209
geografia	43	111	199	141	64	81	180
historia	56	106	182	144	77	68	205
j.angielski	62	123	193	145	78	77	224
j.francuski	48	111	181	129	64	76	230

Rysunek 10. Fragment odpowiedzi na ćw.7

Rysunek 11. Fragment odpowiedzi

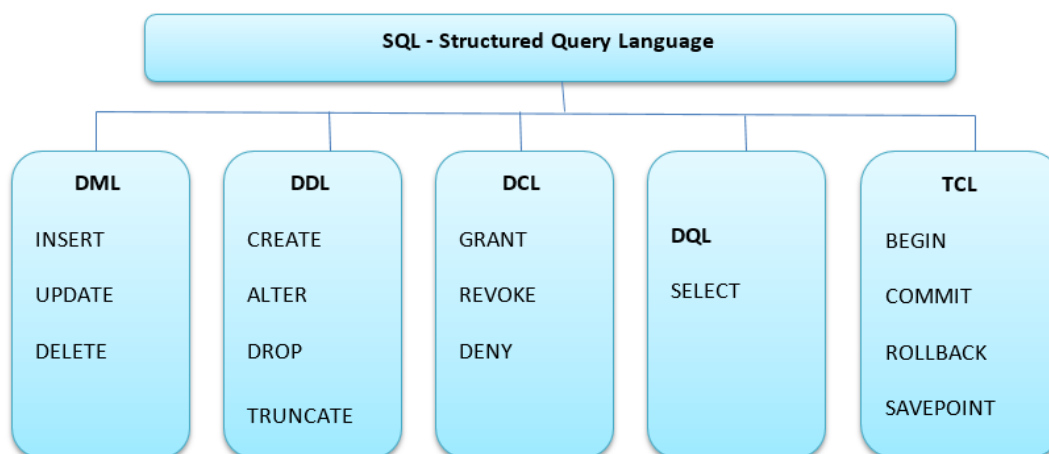
## T38: Wprowadzenie do SQL.

## T39: Język SQL w programie Access.

### SQL (Structured Query Language, strukturalny język zapytań).

Użycie SQL, zgodnie z jego nazwą, polega na zadawaniu zapytań do bazy danych. Zapytania można zaliczyć do jednego z czterech głównych podzbiorów:

- **SQL DML** (ang. *Data Manipulation Language* – „język manipulacji danymi”),
- **SQL DDL** (ang. *Data Definition Language* – „język definicji danych”),
- **SQL DCL** (ang. *Data Control Language* – „język kontroli nad danymi”).
- **SQL DQL** (ang. *Data Query Language* – „język definiowania zapytań”).



### DML

DML (Data Manipulation Language) służy do wykonywania operacji na danych – do ich umieszczania w bazie, kasowania, przeglądania oraz dokonywania zmian. Najważniejsze polecenia z tego zbioru to:

- **INSERT** – umieszczenie danych w bazie,
- **UPDATE** – zmiana danych,
- **DELETE** – usunięcie danych z bazy.

Dane tekstowe muszą być zawsze ujęte w znaki pojedynczego cudzysłowu (').

## DDL

Dzięki DDL (Data Definition Language) można operować na strukturach, w których dane są przechowywane – czyli np. dodawać, zmieniać i kasować tabele lub bazy. Najważniejsze polecenia tej grupy to:

- **CREATE** (np. CREATE TABLE, CREATE DATABASE, ...) – utworzenie struktury (bazy, tabeli, indeksu itp.),
- **DROP** (np. DROP TABLE, DROP DATABASE, ...) – usunięcie struktury,
- **ALTER** (np. ALTER TABLE ADD COLUMN ...) – zmiana struktury (dodanie kolumny do tabeli, zmiana typu danych w kolumnie tabeli).
- **TRUNCATE** - powoduje usunięcie danych z tabeli (np. TRUNCATE TABLE klienci).

## DCL

DCL (Data Control Language) ma zastosowanie do nadawania uprawnień do obiektów bazodanowych. Najważniejsze polecenia w tej grupie to:

- **GRANT** – pozwala użytkownikowi lub roli na wykonywanie operacji określonej przez nadane uprawnienie (np. GRANT ALL PRIVILEGES ON EMPLOYEE TO PIOTR WITH GRANT OPTION – przyznanie wszystkich praw do tabeli EMPLOYEE użytkownikowi PIOTR z opcją pozwalającą mu nadawać prawa do tej tabeli).
- **REVOKE** – usuwa uprzednio nadane lub odmówione uprawnienie (np. REVOKE ALL PRIVILEGES ON EMPLOYEE FROM PIOTR – odebranie użytkownikowi wszystkich praw do tabeli EMPLOYEE).
- **DENY** - odmawia użytkownikowi lub roli określonego uprawnienia i zapobiega odziedziczeniu go po innych rolach

## DQL

DQL (Data Query Language) - to instrukcje, za pomocą których możesz otrzymać z bazy określone dane. Najważniejszym poleceniem jest tutaj **SELECT**.

### TCL (język kontroli transakcji)

SQL **TCL** (ang. Transactional Control Language) – „język kontroli transakcji”

- **BEGIN** - - otwiera transakcję
- **COMMIT** -zatwierdza transakcję
- **ROLLBACK**- cofnij transakcję
- **SAVEPOINT** – zapis punkt transakcji

**Język zapytań SQL.**<sup>2</sup> Dzięki kwerendom (zapytaniom) w Access możemy uzyskać określone informacje z bazy danych. Język SQL jest najpopularniejszym językiem zapytań w różnych systemach baz danych. Możemy za pomocą komend SQL tworzyć kwerendy i wydawać polecenia w bazach danych.

### Zasady pisania poleceń.

- nazwy poleceń piszemy dużymi literami (np. SELECT). Można pisać też małymi
- gdy odwołujemy się do pól jednej tabeli używamy nazwy tego pola (Nazwisko). Gdy odwołujemy się do pól kilku tabel dodajemy też nazwę tabeli z kropką (Liceum.Nazwisko).

- jeśli nazwy pól są wielowyrazowe, to przy odwołaniach ujmujemy je w nawiasy kwadratowe ([Data zadania], Zadania.[Data zadania] lub [Tabela zadań].[Data zadania] )
- podczas wpisywania kryteriów liczby wpisujemy normalnie (2 lub 3,14) teksty w cudzysłowach („Krosno”) daty pomiędzy znakiem # (#2004-09-01#)
- można stosować funkcje agregujące AVG() – średnia arytmetyczna, COUNT() – liczba rekordów, SUM() – suma wartości zawartych w polu, MAX() i MIN() – największa i najmniejsza wartość w polu
- na końcu polecenia SQL wpisujemy średnik

## Podstawy składni zapytań SQL

Składnia DQL (Data Querying Language) – składnia zapytań.

Rozdział ten przedstawia opis kroków i składni przetwarzania zapytań, zgodnie z ich faktyczną kolejnością (która jest ściśle określona).

(Step 5)	<b>SELECT</b>	-- określanie kształtu wyniku, selekcja pionowa (kolumn)
(Step 1)	<b>FROM</b>	-- określenie źródła (źródeł) i relacji między nimi
(Step 2)	<b>WHERE</b>	-- filtracja rekordów
(Step 3)	<b>GROUP BY</b>	-- grupowanie rekordów
(Step 4)	<b>HAVING</b>	-- filtrowanie grup
(Step 6)	<b>ORDER BY</b>	-- sortowanie wyniku

Domyślnie, sortowanie odbywa się w sposób rosnący (ASC ang. ascending). Dla każdej z kolumn możemy określić właściwy kierunek sortowania rekordów :

- rosnąco **ASC** ascending – jest to domyślny sposób i nie trzeba go jawnie określać.
- malejąco **DESC** descending – konieczne jest jawne określenie przy atrybucie, po którym chcemy w ten sposób sortować.

## Niektóre polecenia z języka SQL

### 1. Wyszukiwanie danych przy użyciu instrukcji SELECT

Polecenie SELECT umożliwi wyświetlenie wierszy i kolumn z jednej lub kilku tabel. W połączeniu z funkcjami może wykonywać operacje na danych tj.: obliczenia, formatowanie wyników i tworzenie podsumowania.

**Składnia:** SELECT kolumna1,kolumna2 FROM tabela;

**Przykład:** SELECT \* FROM klient;

W przypadku \* wybierane są wszystkie kolumny z tabeli. Po klauzuli FROM wpisujemy nazwę tabeli z jakiej wypisać chcemy dane.

### 2. Słowo kluczowe DISTINCT

Słowo kluczowe DISTINCT służy do wyświetlenia wartości bez ich powtórzeń. Stosowane jest w przypadku gdy interesuje nas jakie wartości występują w kolumnie a nie ich częstość występowania.

**Składnia:** SELECT [DISTINCT] kolumna1, kolumna2 FROM tabela;

### 3. Aliasy kolumn

W przypadku wybrania kolumny o mało intuicyjnej lub trudnej nazwie możemy wypisać alias dla kolumny z użyciem słowa AS.

**Składnia:** SELECT kolumna1, kolumna2 [AS alias] FROM tabela;

### 4. Filtrowanie wierszy

Wynik zapytań możemy filtrować poprzez spełnienie określonych warunków umieszczonych za klauzulą WHERE. Warunki mogą być połączone za pomocą operatorów.

**Składnia:** SELECT kolumna1, kolumna2 FROM tabela [WHERE warunek1];

## 5. Sortowanie

Wyniki zapytań sortować stosujemy przez klauzule ORDER BY umieszczoną zawsze na końcu zapytania. Chcąc odwrócić wynik sortowania stosujemy dodatkowe słowo DESC.

**Składnia:** SELECT kolumna1, kolumna2 FROM tabela1 ORDER BY kolumna1 [DESC|ASC];

## 6. Omówienie zasad związanych z grupowaniem danych

Grupowanie polega na podzieleniu zbioru wierszy na grupy, które mają pewną wspólną cechę. Grupowania wykonujemy w celu użycia funkcji agregujących nie w stosunku do całego zbioru wierszy, ale do poszczególnych grup wierszy. Do zgrupowania używamy nowej klauzuli GROUP BY wraz z wyborem kolumny lub wyrażenia, według którego mają być pogrupowane wiersze. Na liście klauzuli SELECT mogą się znaleźć tylko kolumny i wyrażenia, według których zapytanie jest grupowane, oraz wywołania funkcji agregujących.

**Składnia:** SELECT kolumna1, kolumna2 FROM tabela1 GROUP BY(kolumna1);

## 7. Sortowanie i filtrowanie pogrupowanych danych w zestawie wyników

Przed dokonaniem grupowania można zastosować filtrowanie z użyciem klauzuli WHERE, która wybierze tylko część wierszy z tabeli. Klauzula WHERE wykonuje się przed grupowaniem, więc nie można w tej klauzuli sprecyzować warunku zawierającego funkcje grupowe. Aby taki warunek zawrzeć w zapytaniu należy zastosować dodatkową klauzulę **HAVING** wraz z odpowiednim warunkiem. Jest ona odpowiednikiem klauzuli WHERE, tylko, że wykonuje się ona po procesie grupowania.

**Składnia:**

SELECT kolumna1, kolumna2 FROM tabela1  
GROUP BY(kolumna1) HAVING warunek1;

### Najczęściej używane operatory:

- **Arytmetyczne** \* mnożenie, / dzielenie, ^ potęgowanie, np.  $4^2=16$  (jako  $4^2$ ), + dodawanie, - odejmowanie, **Mod** - reszta z dzielenia np.  $9 \text{ Mod } 2 = 1$
- **Porównania** = równe, <> różne, > większe, >= większe bądź równe, < mniejsze, <= mniejsze bądź równe,  
**Between ... and ...** ( pomiędzy jednym a drugim), np. Between 1 And 5 określa zbiór liczb [1; 2; 3; 4; 5]  
**Is Null** - porównujący zawartość pola z wartością Null (sprawdza czy pole jest puste)
- **Logiczne And** -koniunkcja (i), **Or**-alternatywa (lub), **Not**- negacja (nie)
- **Operatory pracujące na łańcuchach tekstowych** **Like** – tworzenie wzorca., **&** konkatencja czyli łączenie łańcuchów.

**Funkcje wbudowane** dostępne z menu kontekstowego (uaktywnianego prawym przyciskiem) opcją Buduj, lub do wpisania ręcznego

- **Daty i czasu** **Date()** - wynikiem jest bieżąca data,
- **Year(Date())** wynikiem jest bieżący rok, lub Year(2003-01-01) wynik 2003 itp..
- **\*\*Arytmetyczne Abs(liczba)** - wynikiem jest wartość absolutna ( bezwzględna ) liczby
- **\*\*Tekstowe Len(słowa)=5** zwraca długość słowa wyrażoną ilością znaków tego słowa  
**Left(komputer;2)='ko'** zwraca podaną ilość znaków tekstu od lewej strony
- **Logiczne iif** (warunek ; wyrażenie jeśli prawdziwy warunek; wyrażenie jeśli fałszywy warunek)

Funkcji tej używamy jeśli mamy do wyboru dwa różne działania, w zależności od prawdziwości warunku

- **\*\*Agregujące Min(..), Max(..), Średnia(..), Suma(..), Pierwszy(..), Ostatni(..), Policz(..)**

np. Min([Cena]) wybierająca najniższą wartość z pola Cena

Funkcje agregujące stosowane są między innymi w zapytaniach podsumowujących,

Policz([Telefon])– zliczająca ilość wierszy w tabeli źródłowej z wpisanym telefonem

### Kwerendy obliczeniowe

Przykłady:

wiek : **Year(Date()) - Year([Data urodzenia])** - oblicza wiek osoby, gdzie [Data urodzenia] jest nazwą pola z tabeli czytelnicy

Różne\_podatki : **iif ([Czy lektura]=prawda ; [Cena ]\*0,07 ; [Cena]\*0,22)** -Oblicza podatek 7% z ceny dla lektur, a 22% dla książek które nie są lekturami

Imię i naz : **[Imię]&" "&[Nazwisko]** Tworzy nowe pole z imieniem i nazwiskiem oddzielonymi spacją

## T40: Kwerenda wybierająca w języku SQL.

### Przykładowe kwerendy będą oparte o tabelę STUDENCI.

1. **SELECT Nazwisko, Imie FROM Studenci;**
2. **SELECT \* FROM Studenci;** (Z tabeli Studenci wybieramy wszystkie pola)
3. **SELECT \* FROM Studenci ORDER BY Nazwisko;** (Z tabeli Studenci wybieramy wszystkie pola i sortujemy według pola Nazwisko)
4. **SELECT \* FROM Studenci ORDER BY Nazwisko DESC;** (Sortujemy według pola Nazwisko według odwrotnej kolejności)
5. **SELECT DISTINCT Imie FROM Studenci;** (Usuwanie duplikatów. Tabela z imionami. Każde imię zostanie wypisane tylko jeden raz)
6. **SELECT \* FROM Studenci WHERE Miasto="Krosno";**
7. **SELECT \* FROM Studenci WHERE Miasto<>"Krosno" AND Imie="Anna";**
8. **SELECT \* FROM Studenci WHERE Ocena BETWEEN 3,5 AND 4,5;**
9. **SELECT \* FROM Studenci WHERE Miasto LIKE "K\*";** \* - zastępuje dowolną liczbę znaków ? – zastępuje pojedynczy znak
10. **SELECT \* FROM Studenci WHERE Miasto IN ("Krosno", "Kielce", „Płock”);** osoby mieszkające w jednym z trzech wybranych miast
11. **SELECT Nazwisko, Imie, [Dochod]\*0,22 AS Podatek FROM Studenci;** (Wyświetlamy dodatkową kolumnę o nazwie podatek, która wylicza 22% z pola dochód)
12. **SELECT NAZWISKO, IMIE, IIF([olimpiada]="T", "olimpijczyk";"" ) AS płeć FROM studenci;**
13. Wybieranie danych z kilku tabel  
**SELECT Kontakty.Nazwisko, Kontakty.Imie, Zadania.Data  
FROM Kontakty  
INNER JOIN Zadania  
ON Kontakty.IDOsoby=Zadania.IDOsoby;**
14. **SELECT Count(\*) AS [Nie zdali] FROM Studenci WHERE Zdal="nie";** (Oblicza liczbę studentów, którzy nie zdali egzaminu. Wynik w nowym polu o nazwie Nie zdali)
15. **SELECT Min(dochod) AS [mini], Max(dochod) as [maks] FROM Studenci WHERE zdal="nie";**

16. Grupowanie rekordów **SELECT Miasto, Count(Miasto) AS [Liczba osób] FROM Studenci GROUP BY Miasto;**
17. Modyfikowanie danych **UPDATE Studenci SET Zdal="tak" WHERE Imie="Anna";** Jeśli osoba w tabeli Studenci ma na imię Anna pole Zdal będzie równa TAK – wszystkie Anie zdały egzamin
18. **UPDATE Studenci SET Zdal="tak" WHERE Imie=[Podaj imię osoby];** Przed wykonaniem polecenia pojawia się pytanie o imię, które można wpisać z klawiatury
19. **DELETE FROM Studenci WHERE Imie="Anna";**
20. Tworzenie nowej tabeli **SELECT Nazwisko, Imie, Miasto INTO Persona FROM Studenci WHERE Zdal="nie";** tworzymy nową tabelę o nazwie PERSON i wklejamy do niej pola NAZWISKO, IMIE i MIASTO dla studentów, którzy nie zdali egzaminu warunkowe
21. **SELECT NAZWISKO, IMIE, IIF([plec]="K","kobieta","mężczyzna") AS PŁEĆ FROM studenci;** w nowym polu płeć pojawia się napis kobieta lub mężczyzna w zależności od zawartości pola PLEC
22. **SELECT Count(\*) AS [Nie zdali] FROM Studenci WHERE Zdal="nie";** liczba studentów, którzy nie zdali
23. **SELECT Min(dochod) AS [mini], Max(dochod) as [maks] FROM Studenci WHERE zdal="nie";**w polach minimalne i maksymalne dochody
24. **SELECT count(\*) AS [ILE\_KOBIET] FROM studenci WHERE [plec]="K";** w nowym polu wyświetl liczbę kobiet w tabeli
25. **SELECT count(\*) AS [ILE\_IMION] FROM studenci WHERE [imie]=[Podaj imię kobiety];** wyświetl zapytanie o imię osoby i wyświetl liczbę osób o tym imieniu
26. **SELECT sum([punkty]) AS [suma punktów] FROM studenci WHERE [imie]=[Podaj imię kobiety];** wyświetl sumę punktów, które zdobyły kobiety o podanym w zapytaniu imieniu
27. **SELECT avg([dochod]) AS [dochody] FROM studenci WHERE YEAR([urodzony])=1974;** Wyświetl średnią z dochodów dla osób urodzonych w roku 1974
28. **SELECT count(\*) AS [imiona] FROM studenci WHERE len([imie])=5;** Wyświetl ilość imion dokładnie z pięcioma literami

### Zadanie:

Na podstawie zadań z importu wybierz dowolną bazę danych (od 1 do 5) a następnie wykonaj 15 dowolnych zapytań sql do danej bazy. Jeśli jest to zadanie domowe, lub materiał omawiany na lekcji online należy wykonać z tego zadania sprawozdanie z odpowiednim opisem kwerend. Patrz opis jak wyżej na podstawie przykładu.