

SELECT

```
SELECT [predykat] { * | tabela.* | [tabela.]pole1 [AS alias1] [, [tabela.]pole2 [AS alias2] [, ...]] }  
FROM wyrażenie_tabelowe [, ...] [IN zewnętrzna_baza_danych]  
[WHERE... ]  
[GROUP BY... ]  
[HAVING... ]  
[ORDER BY... ]  
[WITH OWNERACCESS OPTION]
```

predykat	Jeden z następujących predykatów: ALL, DISTINCT, DISTINCTROW lub TOP. Predykat służy do ograniczenia liczby zwracanych rekordów. Jeśli nie podano żadnego predykatu, domyślnie przyjmuje się predykat ALL.
*	Wskazuje, iż wybrane mają być wszystkie pola z podanej tabeli lub tabel.
tabela	Nazwa tabeli zawierającej pola, które składają się na rekordy będące rezultatem instrukcji.
pole1, pole2	Nazwy pól zawierających dane, które mają być pobrane. Jeśli podano kilka pól, są one pobierane w kolejności podanej na liście.
alias1, alias2	Nazwy, które zostaną użyte jako nagłówki kolumn w miejsce nazw kolumn z tabeli.
wyrażenie tabelowe	Nazwa tabeli lub tabel zawierających pobierane dane.
zewnętrzna baza danych	Nazwa bazy danych zawierającej tabelę wymienioną w wyrażeniu tabelowym, o ile nie znajdują się one w bieżącej bazie danych.

WHERE

Określa te rekordy z tabel wymienionych w klauzuli FROM, których dotyczy instrukcja SELECT, UPDATE lub DELETE.

GROUP BY

Scala w jeden rekord rekordy o identycznych wartościach wskazanych na liście pól. Jeśli w instrukcji SELECT zamieszczono funkcje agregacji SQL, na przykład Sum lub Count, to dla każdego tak powstałego rekordu obliczane są odpowiednie wartości zagregowane.

HAVING

Określa, które ze zgrupowanych rekordów pokazuje instrukcja SELECT z klauzulą GROUP BY. Po zgrupowaniu rekordów przez klauzulę GROUP BY klauzula HAVING wyświetla te rekordy spośród zgrupowanych, które spełniają warunki wymienione w klauzuli HAVING.

ORDER BY

Sortuje rekordy będące wynikiem kwerendy rosnąco lub malejąco według wartości określonego pola lub pól.

WITH OWNERACCESS OPTION

W środowisku wielodostępnym z chronioną grupą roboczą deklaracja ta jest używana w kwerendzie do nadania użytkownikowi uruchamiającemu kwerendę tych samych uprawnień co posiadane przez właściciela kwerendy.

KWERENDA TWORZĄCA TABELĘ.

```
SELECT Lp, rrr INTO prb1  
FROM prb;
```

KWERENDA AKTUALIZUJĄCA TABELĘ.

```
UPDATE prb SET prb.rrr = [prb].[rrr]*"1,5";
```

KWERENDA TWORZĄCA (DEFINIUJĄCA) TABELĘ.

```
CREATE TABLE Dane_osobowe (Lp INTEGER CONSTRAINT MyFieldConstraint  
PRIMARY KEY, Imię TEXT (30), Nazwisko TEXT (50), DataUrodzenia DATE, Adres  
TEXT (70), Wpłata CURRENCY);
```

KWERENDA USUWAJĄCA DANE.

```
DELETE prb.Lp, prb.rrr  
FROM prb  
WHERE (((prb.rrr)=9));
```

KWERENDA WYBIERAJĄCA.

```
SELECT Tabela1.Oddział1, Avg(Tabela1.Cena1) AS [Średnia cena]  
FROM Tabela1  
GROUP BY Tabela1.Oddział1  
ORDER BY Tabela1.Oddział1;
```

KWERENDA AKTUALIZUJĄCA TABELĘ.

```
UPDATE Tabela1 SET Tabela1.Cena1 = Tabela1.Cena1*2
WHERE Oddział="krawat";
```

KWERENDA WYBIERAJĄCA TABELĘ.

```
SELECT [Oceny końcowe Klasa I].JPolski, Count(*) AS [Ilość ocen], [Oceny końcowe Klasa I].JPolski*[Ilość ocen] AS Razem
FROM [Oceny końcowe Klasa I]
WHERE ((([Oceny końcowe Klasa I].JPolski)<"7"))
GROUP BY [Oceny końcowe Klasa I].JPolski;
```

KWERENDA WYBIERAJĄCA TABELĘ – SPRZĘŻENIA.

```
SELECT [ZZJP Test SQL KWERENDA].JPolski, [ZZJP Test SQL KWERENDA].[Ilość ocen], [ZZJP Test SQL KWERENDA].Razem, [ZZWOK Test SQL KWERENDA].WiedzaOKult, [ZZWOK Test SQL KWERENDA].[Ilość ocen], [ZZWOK Test SQL KWERENDA].Razem
FROM [ZZJP Test SQL KWERENDA] LEFT JOIN [ZZWOK Test SQL KWERENDA] ON [ZZJP Test SQL KWERENDA].JPolski = [ZZWOK Test SQL KWERENDA].WiedzaOKult;
```

KWERENDA WYBIERAJĄCA TABELĘ.

```
SELECT [Oceny końcowe Klasa I].WiedzaOKult, Count(*) AS [Ilość ocen], [Oceny końcowe Klasa I].WiedzaOKult*[Ilość ocen] AS Razem
FROM [Oceny końcowe Klasa I]
WHERE ((([Oceny końcowe Klasa I].WiedzaOKult)<"7"))
GROUP BY [Oceny końcowe Klasa I].WiedzaOKult;
```