**Podstawowe polecenia SQL**  
  
**1. Pobieranie danych**  
  
SELECT kolumna1, kolumna2 FROM tabela;  
  
• Pobiera określone kolumny z tabeli.  
  
SELECT \* FROM tabela;  
  
• Pobiera wszystkie kolumny.  
  
**2. Filtrowanie wyników (WHERE)**  
  
SELECT \* FROM tabela WHERE warunek;  
  
Przykład:  
  
SELECT \* FROM users WHERE age > 18;  
  
• Pobiera tylko użytkowników starszych niż 18 lat.  
  
**3. Sortowanie wyników (ORDER BY)**  
  
SELECT \* FROM tabela ORDER BY kolumna ASC|DESC;  
  
Przykład:  
  
SELECT \* FROM products ORDER BY price DESC;  
  
• Sortuje produkty malejąco według ceny.  
  
**4. Grupowanie wyników (GROUP BY)**  
  
SELECT kolumna, COUNT(\*) FROM tabela GROUP BY kolumna;  
  
Przykład:  
  
SELECT category, COUNT(\*) FROM products GROUP BY category;  
  
• Zlicza produkty w każdej kategorii.  
  
**5. Filtrowanie grup (HAVING)**  
  
SELECT category, COUNT(\*) FROM products GROUP BY category HAVING COUNT(\*) > 10;  
  
• Pokazuje tylko te kategorie, które mają więcej niż 10 produktów.  
  
**Agregacja danych (funkcje agregujące)**  
• COUNT(\*) – liczy liczbę wierszy  
• AVG(kolumna) – zwraca średnią wartość kolumny  
• SUM(kolumna) – sumuje wartości w kolumnie  
• MAX(kolumna) – zwraca maksymalną wartość  
• MIN(kolumna) – zwraca minimalną wartość  
  
Przykład:  
  
SELECT AVG(salary) FROM employees;  
  
• Oblicza średnią pensję pracowników.  
  
**Łączenie tabel (JOIN)**  
  
**1. INNER JOIN – zwraca tylko pasujące rekordy**  
  
SELECT users.name, orders.amount   
FROM users   
INNER JOIN orders ON [users.id](http://users.id) = orders.user\_id;  
  
• Pobiera użytkowników, którzy mają zamówienia.  
  
**2. LEFT JOIN – zwraca wszystkie rekordy z lewej tabeli i pasujące z prawej**  
  
SELECT users.name, orders.amount   
FROM users   
LEFT JOIN orders ON [users.id](http://users.id) = orders.user\_id;  
  
• Pokazuje wszystkich użytkowników, nawet jeśli nie mają zamówień.  
  
**3. RIGHT JOIN – zwraca wszystkie rekordy z prawej tabeli i pasujące z lewej**  
  
SELECT users.name, orders.amount   
FROM users   
RIGHT JOIN orders ON [users.id](http://users.id) = orders.user\_id;  
  
**4. FULL JOIN – zwraca wszystkie rekordy z obu tabel**  
  
SELECT users.name, orders.amount   
FROM users   
FULL JOIN orders ON [users.id](http://users.id) = orders.user\_id;  
  
**Pobieranie określonej liczby rekordów (TOP, LIMIT)**  
  
**W SQL Server:**  
  
SELECT TOP 5 \* FROM users;  
  
• Pobiera pierwsze 5 wierszy.  
  
**W MySQL/PostgreSQL:**  
  
SELECT \* FROM users LIMIT 5;