

# Instrukcje SQL

## Kompletny zestaw do egzaminu INF.03

### 1. Instrukcje pobierania danych (SELECT)

Instrukcja	Opis	Przykład
<b>SELECT</b>	Pobiera wybrane kolumny	SELECT imie, nazwisko FROM uczniowie;
<b>SELECT *</b>	Pobiera wszystkie kolumny	SELECT * FROM produkty;
<b>DISTINCT</b>	Eliminuje duplikaty	SELECT DISTINCT miasto FROM klienci;
<b>WHERE</b>	Filtruje wiersze	SELECT * FROM uczniowie WHERE klasa = '3TI';
<b>AND / OR</b>	Łączy warunki logiczne	SELECT * FROM produkty WHERE cena > 100 AND kategoria = 'AGD';
<b>NOT</b>	Negacja warunku	SELECT * FROM produkty WHERE NOT kategoria = 'AGD';
<b>ORDER BY ASC</b>	Sortuje rosnąco	SELECT * FROM uczniowie ORDER BY nazwisko ASC;
<b>ORDER BY DESC</b>	Sortuje malejąco	SELECT * FROM produkty ORDER BY cena DESC;
<b>LIMIT</b>	Ogranicza liczbę wyników	SELECT * FROM produkty ORDER BY cena DESC LIMIT 5;
<b>BETWEEN</b>	Zakres wartości	SELECT * FROM zamówienia WHERE data BETWEEN '2024-01-01' AND '2024-12-31';
<b>IN</b>	Lista dozwolonych wartości	SELECT * FROM uczniowie WHERE klasa IN ('1TI', '2TI');
<b>NOT IN</b>	Lista wykluczonych wartości	SELECT * FROM produkty WHERE kategoria NOT IN ('AGD', 'RTV');
<b>LIKE</b>	Wyszukiwanie wzorca	SELECT * FROM klienci WHERE nazwisko LIKE 'Kow%';
<b>IS NULL</b>	Sprawdza wartości puste	SELECT * FROM produkty WHERE opis IS NULL;
<b>IS NOT NULL</b>	Sprawdza niepuste wartości	SELECT * FROM klienci WHERE email IS NOT NULL;

### 2. Funkcje agregujące i grupowanie

Instrukcja	Opis	Przykład
<b>COUNT(*)</b>	Zlicza wszystkie wiersze	SELECT COUNT(*) FROM zamówienia;
<b>COUNT(kolumna)</b>	Zlicza niepuste wartości	SELECT COUNT(email) FROM klienci;
<b>SUM()</b>	Sumuje wartości	SELECT SUM(cena * ilosc) FROM zamówienia;
<b>AVG()</b>	Średnia wartość	SELECT AVG(ocena) FROM oceny;
<b>MIN()</b>	Wartość minimalna	SELECT MIN(cena) FROM produkty;
<b>MAX()</b>	Wartość maksymalna	SELECT MAX(cena) FROM produkty;
<b>GROUP BY</b>	Grupuje wyniki	SELECT klasa, COUNT(*) FROM uczniowie GROUP BY klasa;
<b>HAVING</b>	Filtruje grupy	SELECT klasa, AVG(ocena) FROM oceny GROUP BY klasa HAVING AVG(ocena) > 3.5;
<b>AS (alias)</b>	Nadaje nazwę kolumnie	SELECT COUNT(*) AS liczba FROM uczniowie;

### 3. Łączenie tabel (JOIN)

Instrukcja	Opis	Przykład
<b>INNER JOIN</b>	Część wspólna tabel	SELECT u.imie, o.ocena FROM uczniowie u INNER JOIN oceny o ON u.id = o.uczen_id;
<b>LEFT JOIN</b>	Wszystko z lewej + dopasowane	SELECT k.nazwa, z.data FROM klienci k LEFT JOIN zamówienia z ON k.id = z.klient_id;
<b>RIGHT JOIN</b>	Wszystko z prawej + dopasowane	SELECT p.nazwa, z.ilosc FROM zamówienia z RIGHT JOIN produkty p ON z.produkt_id = p.id;

### 4. Modyfikacja danych (DML)

Instrukcja	Opis	Przykład
<b>INSERT INTO</b>	Dodaje wiersz (wszystkie kol.)	INSERT INTO uczniowie VALUES (NULL, 'Jan', 'Kowalski', '1TI');
<b>INSERT INTO (...)</b>	Dodaje wiersz (wybrane kol.)	INSERT INTO uczniowie (imie, nazwisko) VALUES ('Jan', 'Kowalski');

Instrukcja	Opis	Przykład
UPDATE ... SET	Modyfikuje dane	UPDATE produkty SET cena = 99.99 WHERE id = 5;
DELETE FROM	Usuwa wiersze	DELETE FROM zamowienia WHERE data < '2020-01-01';

## 5. Definicja struktury (DDL)

Instrukcja	Opis	Przykład
CREATE DATABASE	Tworzy bazę danych	CREATE DATABASE sklep;
DROP DATABASE	Usuwa bazę danych	DROP DATABASE stara_baza;
CREATE TABLE	Tworzy tabelę	CREATE TABLE uczniowie (id INT PRIMARY KEY AUTO_INCREMENT, imie VARCHAR(50));
DROP TABLE	Usuwa tabelę	DROP TABLE stara_tabela;
ALTER TABLE ADD	Dodaje kolumnę	ALTER TABLE uczniowie ADD email VARCHAR(100);
ALTER TABLE DROP	Usuwa kolumnę	ALTER TABLE uczniowie DROP COLUMN email;
ALTER TABLE MODIFY	Zmienia typ kolumny	ALTER TABLE uczniowie MODIFY imie VARCHAR(100);

## 6. Ograniczenia (CONSTRAINTS)

Ograniczenie	Opis	Przykład
PRIMARY KEY	Klucz główny (unikalny, niepusty)	id INT PRIMARY KEY
AUTO_INCREMENT	Automatyczne numerowanie	id INT PRIMARY KEY AUTO_INCREMENT
FOREIGN KEY	Klucz obcy (relacja)	FOREIGN KEY (klient_id) REFERENCES klienci(id)
NOT NULL	Wartość wymagana	imie VARCHAR(50) NOT NULL
UNIQUE	Wartość unikalna	email VARCHAR(100) UNIQUE
DEFAULT	Wartość domyślna	status VARCHAR(20) DEFAULT 'aktywny'

## 7. Zarządzanie użytkownikami i uprawnieniami (DCL)

Instrukcja	Opis	Przykład
CREATE USER	Tworzy użytkownika	CREATE USER 'jan'@'localhost' IDENTIFIED BY 'haslo123';
DROP USER	Usuwa użytkownika	DROP USER 'jan'@'localhost';
GRANT	Nadaje uprawnienia	GRANT SELECT, INSERT ON sklep.produkty TO 'jan'@'localhost';
GRANT ALL	Nadaje wszystkie uprawnienia	GRANT ALL PRIVILEGES ON sklep.* TO 'admin'@'localhost';
REVOKE	Odbiera uprawnienia	REVOKE INSERT ON sklep.produkty FROM 'jan'@'localhost';
FLUSH PRIVILEGES	Odświeża uprawnienia	FLUSH PRIVILEGES;

### Uprawnienia do nadawania:

Uprawnienie	Co pozwala
SELECT	Odczytywanie danych
INSERT	Wstawianie danych
UPDATE	Modyfikowanie danych
DELETE	Usuwanie danych
CREATE	Tworzenie tabel/baz
DROP	Usuwanie tabel/baz
ALL PRIVILEGES	Wszystkie powyższe

## 8. Przydatne funkcje SQL

Funkcja	Opis	Przykład
ROUND()	Zaokrąglanie	SELECT ROUND(cena, 2) FROM produkty;
CONCAT()	Łączenie tekstów	SELECT CONCAT(imie, ' ', nazwisko) AS pelne FROM uczniowie;
UPPER()	Wielkie litery	SELECT UPPER(nazwisko) FROM klienci;
LOWER()	Małe litery	SELECT LOWER(email) FROM klienci;
LENGTH()	Długość tekstu	SELECT LENGTH(nazwa) FROM produkty;
NOW()	Aktualna data i czas	INSERT INTO logi (data) VALUES (NOW());
CURDATE()	Aktualna data	SELECT * FROM zamowienia WHERE data = CURDATE();

Funkcja	Opis	Przykład
<b>YEAR() / MONTH()</b>	Wyciąga części daty	<code>SELECT * FROM zamowienia WHERE YEAR(data) = 2024;</code>

## 9. Wzorce LIKE – ściągawka

Wzorzec	Znaczenie	Pasuje do
'Kow%'	Zaczyna się od "Kow"	Kowalski, Kowal, Kowacki
'%ski'	Kończy się na "ski"	Kowalski, Nowicki, Malinowski
'%owa%'	Zawiera "owa"	Kowalski, Noworolnik
'Ja_'	"Ja" + dokładnie 1 znak	Jan, Jak, Jar
'___'	Dokładnie 3 znaki	Jan, Ewa, Ola

**Powodzenia na egzaminie!** 🎓