# Zajęcia 1.

*Język SQL, zdania CREATE TABLE, prosty SELECT*

# Utwórz nową bazę danych o nazwie Z1 i utwórz w niej tabelę o Studenci o takich kolumnach: Id\_studenta (INT), Imię (NVARCHAR(50), Nazwisko (NVARCHAR(50)), Data\_urodzenia DATE, Nr\_albumu INT). Kolumna K1 ma być kluczem.

CREATE TABLE Studenci

( Id\_studenta INT PRIMARY KEY,

Imię NVARCHAR(50) NOT NULL,

Nazwisko NVARCHAR(50) NOT NULL,

Data\_urodzenia DATE NULL,

Nr\_albumu INT UNIQUE NOT NULL)

Wprowadź przykładowe dane:

INSERT INTO Studenci VALUES (1,'Jan','Kowalski',NULL, 1001)

INSERT INTO Studenci (Id\_studenta, Imię, Nazwisko, Nr\_albumu) VALUES (2,'Anna','Nowak', 1002)

Spróbuj dopisać jeszcze taki rekord:

INSERT INTO Studenci VALUES (1,'Piotr','Wrona','1989-05-01', 1003)

Jaki komunikat się pojawia?

Wstaw ten wiersz jeszcze raz, tym razem wprowadź poprawny identyfikator:

INSERT INTO Studenci VALUES (3,'Piotr','Wrona','1989-05-01', 1003)

Wyświetl całą tabelę Studenci.

SELECT \* FROM Studenci

Wyświetl wszystkie wiersze z tabeli Studenci, ale tym razem interesuje nas tylko Nazwisko i Imię.

SELECT Nazwisko, Imię FROM Studenci

Wyświetl dane tylko studenta o identyfikatorze 3.

SELECT \* FROM Studenci

WHERE Id\_studenta = 3

Wyświetl tylko dane studentów a numerze albumu większym lub równym 1002.

SELECT \* FROM Studenci WHERE Nr\_albumu > 1002

Skasuj dane studenta o identyfikatorze 2. Zdaniem SELECT sprawdź efekt.

DELETE FROM Studenci WHERE Id\_studenta = 2

SELECT \* FROM Studenci

Korzystając z bazy danych *Northwind* (należy ją odtworzyć z kopii zapasowej, jeśli nie jest jeszcze zainstalowana) utwórz zapytanie, które wyświetli identyfikator klienta oraz jego nazwę, miasto i kraj (pola CustomerID, CompanyName, City, Country z tabeli *Customers*).

select CustomerID, CompanyName, City, Country

from Customers;

Zmodyfikuj poprzednie zdanie tak, by wyświetlani byli tylko klienci z Polski.

select CustomerID, CompanyName, City, Country

from Customers

where Country = N'Poland';

Zmodyfikuj poprzednie zdanie tak, by wyświetlani byli tylko klienci z Polski i z Niemiec.

select CustomerID, CompanyName, City, Country

from Customers

where Country = N'Poland' or Country = N'Germany';

Utwórz zapytanie, które z tabeli Customers wszystkich klientów, których nazwa (CompanyName) rozpoczyna się od litery C. Ilu ich jest?

select \*

from Customers

where CompanyName like N'C%';

go

select count(\*) as TotalQty

from Customers

where CompanyName like N'C%';

Utwórz zapytanie, które wyświetli z tabeli Customers wszystkich klientów, których nazwa (CompanyName) zawiera na początku lub w środku literę **i**, przy czym na końcu nazwy musi być litera **a**. Ilu ich jest?

select \*

from Customers

where CompanyName like N'%i%a';

go

select count(\*) as TotalQty

from Customers

where CompanyName like N'%i%a';

Utwórz zapytanie, które wyświetli z tabeli Customers wszystkich klientów, których nazwa (CompanyName) rozpoczyna się od liter **A** lub **C** lub **N** lub **P**. Ilu ich jest?

select \*

from Customers

where CompanyName like N'[ACNP]%';

go

select count(\*) as TotalQty

from Customers

where CompanyName like N'[ACNP]%';

Utwórz zapytanie, które wyświetli z tabeli Customers wszystkich klientów, których nazwa (CompanyName) rozpoczyna się od liter od **A** do **H**. Ilu ich jest?

select \*

from Customers

where CompanyName like N'[A-H]%'

order by CompanyName asc;

go

select count(\*) as TotalQty

from Customers

where CompanyName like N'[A-H]%';

Utwórz zapytanie, które wyświetli z tabeli Customers wszystkich klientów, których nazwa (CompanyName) NIE rozpoczyna się od litery od **C** ani NIE rozpoczyna się od litery **P**. Ilu ich jest?

select \*

from Customers

where CompanyName like N'[^C^P]%'

order by CompanyName asc;

go

select count(\*) as TotalQty

from Customers

where CompanyName like N'[^C^P]%';

# Wyświetl kolumnę Country z tabeli Customers. Jak zrobić, by nazwy państw nie powtarzały się?

select distinct Country

from Customers

order by Country asc;

# Wyświetl wszystkie kolumny z tabeli Customers ale tylko dane tych klientów, którzy nie mają nic wpisane do kolumny Region.

select \*

from Customers

where Region is null;

Wyświetl identyfikatory i nazwy wszystkich produktów (kolumny ProductID oraz ProductName z tabeli *Products*). Ile jest produktów w tabeli?

select ProductID, ProductName

from Products;

go

select count(\*) as TotalQty

from Products;

Zmodyfikuj rozwiązanie poprzedniego zadania tak, by wyświetlana była też cena jednostkowa (kolumna UnitPrice) oraz by wyświetlane były tylko produkty, których cena jest większa od 40. Ile jest takich produktów? (można to zobaczyć w zakładce *Messages*)

select ProductID, ProductName, UnitPrice

from Products

where UnitPrice > 40;

Zmodyfikuj rozwiązanie poprzedniego zadania tak, by wyświetlane były tylko produkty, których cena jest większa od 10 i mniejsza od 20. Ile jest takich produktów?

select ProductID, ProductName, UnitPrice

from Products

where UnitPrice > 10 and UnitPrice < 20;

Zmodyfikuj rozwiązanie poprzedniego zadania tak, by wyświetlane były tylko produkty, których cena jest mniejsza od 10 lub większa od 100. Ile jest takich produktów?

select ProductID, ProductName, UnitPrice

from Products

where UnitPrice < 10 or UnitPrice > 100;

Wyświetl wszystkie dane produktów należących do kategorii 2 oraz tych, które należą do kategorii 5. Ponadto zakładamy, że interesują nas tylko produkty (z wymienionych kategorii), których cena jest z przedziału od 10 do 50 włącznie. Ile jest takich produktów?

select \*

from Products

where CategoryID in (2, 5) and UnitPrice between 10 and 50;

Wyświetl wszystkie produkty należące do kategorii 2 oraz te, które należą do kategorii 5, ponadto ich cena jest z przedziału od 10 do 50 włącznie i nazwa rozpoczyna się od liter od A do M. Ile jest takich produktów?

select \*

from Products

where CategoryID in (2, 5)

and UnitPrice between 10 and 50

and ProductName like N'[A-M]%';

Wyświetl wszystkie produkty (o dowolnych nazwach) należące do kategorii 2 oraz te, które należą do kategorii 5, ponadto ich cena jest z przedziału od 10 do 50 włącznie. Ponadto na tej samej liście mają się znaleźć produktu, których nazwa rozpoczyna się od litery C lub P, bez względu na to do której należą kategorii i jaka jest ich cena. Ile jest takich produktów?

select \*

from Products

where (CategoryID in (2, 5) and UnitPrice between 10 and 50)

or ProductName like N'[CP]%';

Wyświetl numery zamówień, identyfikatory klientów oraz daty zamówień z tabeli *Orders* dla zamówień z września roku 1996.

select OrderID, CustomerID, OrderDate

from Orders

where OrderDate >= '1996-09-01 00:00:00.000'

and OrderDate <= '1996-09-30 23:59:59.999';

Wykorzystując funkcję MONTH wyświetl numery zamówień, identyfikatory klientów oraz daty zamówień z tabeli *Orders* dla zamówień z września każdego roku (tzn. rok jest dowolny).

select OrderID, CustomerID, OrderDate

from Orders

where month(OrderDate) = 9;

Utwórz widok (VIEW), który wyświetli kolumny CustomerID, CompanyName, City oraz Country z tabeli Customers, przy czym chodzi nam tylko o klientów z Francji (Country = N'Germany’).

create view v\_21

as

select CustomerID, CompanyName, City, Country

from Customers

where Country = N'France';

go

select \* from v\_21;

Utwórz widok (VIEW), który wyświetli kolumny CustomerID, CompanyName, City oraz Country z tabeli Customers, przy czym chodzi nam tylko o klientów z Polski. Tym razem jednak zmień w widoku nazwy kolumn odpowiednio na IdKlienta, NazwaKlienta oraz Miejscowosc.

create view v\_22

as

select CustomerID as IdKlienta,

CompanyName as NazwaKlienta,

City as Miejscowosc,

Country as Kraj

from Customers

where Country = N'Poland';

go

select \* from v\_22;

Utwórz tabelę CustomersCopy, która będzie kopią tabeli Customers.

SELECT \* Into CustomersCopy

FROM Customers

Sprawdź strukturę nowo utworzonej tabeli – czy jest w niej klucz główny?

Dodaj do tabeli CustomersCopy kolumnę o nazwie CustomerNumber typu SMALLINT, która będzie automatycznie numerowana poprzez ustawienie własności IDENTITY(1,1).

ALTER TABLE CustomersCopy

ADD CustomerNumber INT NOT NULL IDENTITY(1,1)

Sprawdź zawartość tabeli.

Utwórz klucz główny w tabeli – ma to być kolumna CustomerNumber.

ALTER TABLE CustomersCopy

ADD PRIMARY KEY(CustomerNumber)

Z tabeli CustomersCopy skasuj dane klientów o numerach 5, 7, 15, 21 i 44 (jednym zdaniem DELETE).

delete from CustomersCopy

where CustomerNumber in (5, 7, 15, 21, 44);

Skasuj tabelę utworzoną w poprzednim zadaniu i utwórz ją jeszcze raz – jako kopię tabeli Customers.

SELECT \* Into CustomersCopy

FROM Customers

Dodaj do tabeli CustomersCopy kolumnę o nazwie CustomerNumber typu SMALLINT, która będzie automatycznie numerowana – tym razem z wykorzystaniem sekwencji. Wskazówki jak to zrobić znajdziesz w pliku z wykładu (SQL cz.2.).

drop table if exists CustomersCopy;

go

drop sequence if exists id\_seq;

go

select \* into CustomersCopy

from Customers;

go

create sequence id\_seq as int

start with 1

increment by 1;

go

alter table CustomersCopy

add CustomerNumber int not null default next value for id\_seq;

go

select \* from CustomersCopy;