JOINY

1. Napisz polecenie zwracające nazwy produktów i firmy je dostarczające tak aby produkty bez „dostarczycieli” i „dostarczyciele” bez produktów nie pojawiali się w wyniku.

SELECT Products.ProductName, Suppliers.CompanyName

FROM Products

INNER JOIN Suppliers

On Products.SupplierID = Suppliers.SupplierID

2. Napisz polecenie zwracające jako wynik nazwy klientów, którzy złożyli zamówienia po 1 marca 1998

SELECT DISTINCT Customers.CompanyName

FROM Customers

INNER JOIN Orders

ON Orders.CustomerID=Customers.CustomerID

WHERE Orders.OrderDate > '3/1/98'

3. Napisz polecenie zwracające wszystkich klientów z datami zamówień.

SELECT Customers.CompanyName, Orders.OrderDate

FROM Customers

LEFT OUTER JOIN Orders

ON Orders.CustomerID = Customers.CustomerID

4. Napisz polecenie, które wyświetla listę dzieci będących członkami biblioteki. Interesuje nas imię, nazwisko i data urodzenia dziecka.

SELECT member.firstname, member.lastname, juvenile.birth\_date

FROM juvenile

INNER JOIN member

ON juvenile.member\_no = member.member\_no

--5. Napisz polecenie, które podaje tytuły aktualnie wypożyczonych książek

SELECT DISTINCT title.title

FROM title

INNER JOIN loan

ON loan.title\_no=title.title\_no

--6. Podaj informacje o karach zapłaconych za przetrzymywanie książki o tytule ‘Tao Teh King’.

--Interesuje nas data oddania książki, ile dni była przetrzymywana i jaką zapłacono karę

SELECT loanhist.in\_date, DAY(loanhist.in\_date - loanhist.due\_date) as DaysKept, loanhist.fine\_paid

FROM loanhist

INNER JOIN copy

ON copy.copy\_no=loanhist.copy\_no

INNER JOIN title

ON copy.title\_no=title.title\_no

WHERE title.title = 'Tao Teh King'

--7. Napisz polecenie które podaje listę książek (numery ISBN) zarezerwowanych przez osobę o nazwisku:

---Stephen A. Graff

SELECT reservation.isbn

FROM reservation

INNER JOIN member

on reservation.member\_no = member.member\_no

WHERE (member.firstname='Stephen' AND member.lastname = 'Graff' AND member.middleinitial='A')

\_--8. Wybierz nazwy i ceny produktów o cenie jednostkowej pomiędzy 20 a 30, dla każdego produktu podaj dane adresowe dostawcy

SELECT Products.ProductName, Products.UnitPrice, Suppliers.Address, Suppliers.City, Suppliers.Country

FROM Products

INNER JOIN Suppliers

ON Products.SupplierID = Suppliers.SupplierID

WHERE Products.UnitPrice BETWEEN 10 AND 20

\_--9. Wybierz nazwy produktów oraz informacje o stanie magazynu dla produktów dostarczanych przez firmę ‘Tokyo Traders’

\_

SELECT ProductName, UnitsInStock

FROM Products

INNER JOIN SUPPLIERS

ON Products.SupplierID = Suppliers.SupplierID

WHERE CompanyName = 'Tokyo Traders'

--10. Czy są jacyś klienci którzy nie złożyli żadnego zamówienia w 1997 roku, jeśli tak to pokaż ich dane adresowe

\_

SELECT CompanyName, Address, City, Country

FROM Customers

INNER JOIN Orders

ON Orders.CustomerID=Customers.CustomerID

WHERE YEAR(OrderDate) <> '1997'

--11. Wybierz nazwy i numery telefonów dostawców, dostarczających produkty, których aktualnie nie ma w magazynie

SELECT CompanyName, Phone, UnitsInStock

FROM Suppliers

INNER JOIN Products

ON Products.SupplierID = Suppliers.SupplierID

WHERE UnitsInStock = 0

--1.Wybierz nazwy i ceny produktów (baza northwind) o cenie jednostkowej pomiędzy 20 a 30, dla każdego produktu

--podaj dane adresowe dostawcy, interesują nas tylko produkty z kategorii ‘Meat/Poultry’

SELECT ProductName, UnitPrice, Address, City, Region

FROM Products

JOIN Suppliers

ON Products.SupplierID=Suppliers.SupplierID

JOIN Categories

On Products.CategoryID=Categories.CategoryID

WHERE CategoryName='Meat/Poultry'

\_--2.Wybierz nazwy i ceny produktów z kategorii ‘Confections’ dla każdego produktu podaj nazwę dostawcy.

\_

SELECT ProductName, UnitPrice, CompanyName

FROM Products

JOIN Suppliers

ON Products.SupplierID=Suppliers.SupplierID

JOIN Categories

On Products.CategoryID=Categories.CategoryID

WHERE CategoryName='Confections'

--3. Wybierz nazwy i numery telefonów klientów , którym w 1997 roku przesyłki dostarczała firma ‘United Package’

\_

SELECT Customers.CompanyName, Customers.Phone

FROM Customers

JOIN Orders

On Customers.CustomerID = Orders.CustomerID

JOIN Shippers

ON Orders.ShipVia=Shippers.ShipperID

WHERE (Shippers.CompanyName = 'United Package' AND YEAR(Orders.OrderDate)= '1997')

--4. Wybierz nazwy i numery telefonów klientów, którzy kupowali produkty z kategorii ‘Confections

SELECT Customers.CompanyName, Customers.Phone

FROM Customers

JOIN Orders On Customers.CustomerID = Orders.CustomerID

JOIN [Order Details] ON Orders.OrderID=[Order Details].OrderID

JOIN Products ON Products.ProductID = [Order Details].ProductID

JOIN Categories ON Products.CategoryID=Categories.CategoryID

WHERE CategoryName = 'Confections'

--1.\_ Napisz polecenie, które wyświetla listę dzieci będących członkami biblioteki. Interesuje nas imię,

--nazwisko, data urodzenia dziecka i adres zamieszkania dziecka.

SELECT firstname, lastname, birth\_date, street, city, state

FROM juvenile

JOIN member on juvenile.member\_no = member.member\_no

JOIN adult ON juvenile.adult\_member\_no = adult.member\_no

--2.\_ Napisz polecenie, które wyświetla listę dzieci będących członkami biblioteki. Interesuje nas imię,

--nazwisko, data urodzenia dziecka, adres zamieszkania dziecka oraz imię i nazwisko rodzica.

SELECT A.firstname, A.lastname, birth\_date, street, city, state, B.firstname, B.lastname

FROM juvenile

JOIN member AS A on juvenile.member\_no = A.member\_no

JOIN adult ON juvenile.adult\_member\_no = adult.member\_no

JOIN member AS B on adult.member\_no=B.member\_no

--1.Napisz polecenie, które wyświetla pracowników oraz ich podwładnych.

SELECT (Szef.LastName +' '+ Szef.FirstName) as Szef, (Prac.LastName+ ' '+ Prac.FirstName) AS Pracownik

FROM Employees AS Szef

JOIN Employees AS Prac

ON Prac.ReportsTo=Szef.EmployeeID

--2.\_ Napisz polecenie, które wyświetla pracowników, którzy nie mają podwładnych.

SELECT Szef.LastName

FROM Employees AS Szef

LEFT OUTER JOIN Employees AS Prac

ON Prac.ReportsTo=Szef.EmployeeID

WHERE Prac.EmployeeID IS NULL

\_--3. Napisz polecenie, które wyświetla adresy członków, którzy mają dzieci urodzone przed 1 stycznia 1996

SELECT street, city, state

FROM adult

JOIN juvenile ON juvenile.adult\_member\_no = adult.member\_no

WHERE juvenile.birth\_date < '1/1/96'

\_--4. Napisz polecenie, które wyświetla adresy członków biblioteki, którzy mają dzieci urodzone przed 1 stycznia

--1996. Interesują nas tylko adresy takich członków biblioteki, którzy aktualnie nie przetrzymują książek.

SELECT street, city, state

FROM adult

JOIN juvenile ON juvenile.adult\_member\_no = adult.member\_no

JOIN member ON member.member\_no = adult.member\_no

JOIN loan ON member.member\_no = loan.member\_no

WHERE (juvenile.birth\_date < '1/1/96' AND due\_date > GETDATE())

--1.\_ Napisz polecenie które zwraca imię i nazwisko (jako pojedynczą kolumnę – name), oraz informacje o adresie:

--ulica, miasto, stan kod (jako pojedynczą kolumnę – address) dla wszystkich dorosłych członków biblioteki

\_

SELECT firstname+ ' '+lastname as Name, street + ' '+ city + ' '+ state + ' '+zip AS Address

FROM Adult JOIN member ON adult.member\_no = member.member\_no

--2. Napisz polecenie które zwraca informację o użytkownikach biblioteki o nr 250, 342, i 1675 (nr, imię i

--nazwisko członka biblioteki) oraz informacje o zarezerwowanych książkach (isbn, data)

\_

SELECT member.member\_no, firstname, lastname, isbn, log\_date

FROM member

LEFT JOIN reservation ON member.member\_no = reservation.member\_no

WHERE (member.member\_no=250 OR member.member\_no=342 OR member.member\_no=1675)

--3. Podaj listę członków biblioteki mieszkających w Arizonie

--(AZ), którzy mają więcej niż dwoje dzieci zapisanych do biblioteki

SELECT adult.member\_no

FROM adult

JOIN juvenile ON juvenile.adult\_member\_no=adult.member\_no

WHERE state = 'AZ'

GROUP BY adult.member\_no

HAVING COUNT(adult.member\_no) > 2

--4.Podaj listę członków biblioteki mieszkających w Arizonie (AZ) którzy mają więcej niż dwoje dzieci zapisanych do

--biblioteki oraz takich którzy mieszkają w Kaliforni i mają więcej niż troje dzieci zapisanych do biblioteki

SELECT adult.member\_no

FROM adult

JOIN juvenile ON juvenile.adult\_member\_no=adult.member\_no

WHERE state = 'AZ'

GROUP BY adult.member\_no

HAVING COUNT(adult.member\_no) > 2

UNION

SELECT adult.member\_no

FROM adult

JOIN juvenile ON juvenile.adult\_member\_no=adult.member\_no

WHERE state = 'CA'

GROUP BY adult.member\_no

HAVING COUNT(adult.member\_no) > 3

--6. Podaj czytelników (imię, nazwisko), którzy nigdy nie pożyczyli żadnej książki.

SELECT firstname, lastname   
FROM member  
LEFT JOIN loan ON loan.member\_no = member.member\_no  
LEFT JOIN loanhist on loanhist.member\_no = member.member\_no  
WHERE loan.isbn is NULL and loanhist.isbn is null

--1. Dla każdej kategorii produktu (nazwa), podaj łączną liczbę zamówionych przez klienta jednostek.

SELECT CategoryName, Customers.CustomerID, count(\*) as quantity

FROM Categories

JOIN PRODUCTS ON Products.CategoryID=Categories.CategoryID

JOIN [Order Details] ON [Order Details].ProductID=Products.ProductID

JOIN Orders on [Order Details].OrderID=Orders.OrderID

JOIN Customers ON Orders.CustomerID = CUstomers.CustomerID

GROUP BY CategoryName, Customers.CustomerID

--2. Dla każdego zamówienia podaj łączną liczbę zamówionych jednostek oraz nazwę klienta.

SELECT Customers.CompanyName, count(\*) as quantity

FROM Orders

JOIN Customers ON Orders.CustomerID = CUstomers.CustomerID

GROUP BY Customers.CompanyName, Customers.CustomerID

--3. Zmodyfikuj poprzedni przykład, aby pokazać tylko takie zamówienia, dla których łączna liczba jednostek jest większa niż 250.

SELECT Customers.CompanyName, count(\*) as quantity

FROM Orders

JOIN Customers ON Orders.CustomerID = CUstomers.CustomerID

GROUP BY Customers.CompanyName, Customers.CustomerID

Having count(\*) >250

--4. Dla każdego klienta (nazwa) podaj nazwy towarów, które zamówił

SELECT Customers.CompanyName, ProductName

FROm Customers

LEFT JOIN Orders ON Orders.CustomerID = Customers.CustomerID

JOIn [Order Details] ON [Order Details].OrderID = Orders.OrderID

JOIN Products ON [Order Details].ProductID = Products.ProductID

Order BY Customers.CompanyName

--5. Dla każdego klienta (nazwa) podaj wartość poszczególnych zamówień. Gdy klient nic nie zamówił też powinna

--pojawić się informacja

SELECT Customers.CompanyName, SUM(unitPrice\*Quantity\*(1-Discount))

FROM Customers

LEFT JOIN Orders ON Orders.CustomerID = Customers.CustomerID

JOIn [Order Details] ON [Order Details].OrderID = Orders.OrderID

GROUP BY Customers.CustomerID, Customers.CompanyName