

1. Wykonaj zapytanie

SELECT

    firstname, lastname, city, country

FROM HR.Employees;

-----

2. Zaznacz [mysz]
- 

3. Wykonaj zapytanie

SELECT

    firstname, lastname, city, country

FROM HR.Employees

WHERE country = 'USA';

-----

4. Wykonaj zapytanie

/\*

SELECT

    firstname, lastname, city, country

FROM HR.Employees

WHERE country = 'USA'

ORDER BY lastname;

\*/

-----

5. Napisz instrukcję SELECT, która zwróci wszystkie wiersze i wszystkie kolumny z Sales.Customers.

drag-and-drop

-----

6. Napisz instrukcję SELECT, która zwróci contactname, address, postcode, city, a country.
-

7. Napisz instrukcję SELECT dla tabeli Sales.Customers pokazując tylko kolumnę country.

-----

8. Zmodyfikuj poprzednią formułę tak aby wartości się nie powtarzały.

-----

9. W jakich okolicznościach będą zwrócone te same wartości [Sales.Customers]?

SELECT

city, region

FROM Sales.Customers;

SELECT DISTINCT

city, region

FROM Sales.Customers;

-----

10. Napisz formułę SELECT zwracającą kolumny contactname i contacttitle z tabeli Sales.Customers, przypisz alias "C" do tabeli. Użyj aliasu tabeli C, aby poprzedzić nazwy dwóch potrzebnych kolumn na liście SELECT.

-----

11. Napisz formułę SELECT aby zwrócić kolumny: contactname, contacttitle, companyname z tabeli Sales.Customers. Przypisz kolumnom aliasy: Name, Title, Company Name.

-----

12. Napisz zapytanie, aby wyświetlić kolumnę productname z tabeli Production.Products, używając "P" jako aliasu tabeli i nazwy produktu jako aliasu kolumny.

13. Programista napisał zapytanie, aby pobrać dwie kolumny (miasto i region) z tabeli Sales.Customers. Po wykonaniu zapytania zwraca tylko jedną kolumnę. Twoim zadaniem jest przeanalizowanie zapytania, poprawienie go w celu zwrócenia dwóch kolumn i wyjaśnienie, dlaczego zapytanie zwróciło tylko jedną kolumnę.

SELECT

city country

FROM Sales.Customers;

-----

14. Napisz instrukcję SELECT, aby wyświetlić kolumny categoryid i productname z tabeli Production.Products.

---

15. Popraw instrukcję SELECT w zadaniu 1, dodając wyrażenie CASE, które generuje kolumnę wyników o nazwie categoryname. Nowa kolumna powinna zawierać category ID na podstawie dostarczonej wcześniej tabeli mapowania. Użyj wartości "Other" dla identyfikatorów kategorii, które nie zostały znalezione w tabeli odwzorowań.

---

16. Zmodyfikuj instrukcję SELECT w zadaniu 2, dodając nową kolumnę o nazwie iscampaign. W tej kolumnie zostanie wyświetlony opis "Campaign Products" dla kategorii Beverages, Produce, Seafood oraz opis "Non-Campaign Products" dla wszystkich pozostałych kategorii.

17. Napisz instrukcję SELECT, która zwróci kolumnę productname z tabeli Production.Products (użyj aliasu tabeli "p") i kolumny categoryname z tabeli Production.Categories (użyj aliasu tabeli "c"), używając inner join.

Powiedzmy, że istnieje nowy wiersz w tabeli Production.Categories, a ta nowa kategoria produktów nie zawiera żadnych produktów powiązanych z nią w tabeli Production.Products. Czy ten wiersz zostanie uwzględniony w wyniku instrukcji SELECT zapisanej w zadaniu? Proszę wytłumaczyć.

---

18. Programista napisał to zapytanie. Wykonaj zapytanie dokładnie tak, jak zostało napisane w oknie zapytania i obserwuj wynik. Otrzymujesz błąd. Co to jest komunikat o błędzie? Jak myślisz, dlaczego masz ten błąd?

SELECT

custid, contactname, orderid

FROM Sales.Customers

INNER JOIN Sales.Orders ON Customers.custid = Orders.custid;

19. Zauważ, że istnieją pełne nazwy tabel źródłowych zapisane jako aliasy tabel. Zastosuj potrzebne zmiany do instrukcji SELECT, aby działały bezbłędnie. Przetestuj zmiany, wykonując instrukcję T-SQL.

---

20. Skopiuj instrukcję T-SQL z zadania 18 i zmodyfikuj ją, aby wykorzystwała aliasy tabel "C" dla tabeli Sales.Customers i "O" dla tabeli Sales.Orders. Zmień prefiks kolumn w instrukcji SELECT z pełnymi nazwami tabel źródłowych i wykonaj instrukcję. Otrzymujesz błąd. Czemu? Zmień instrukcję SELECT, aby użyć aliasów tabeli zapisanych na początku zadania.

---

21. Skopiuj instrukcję T-SQL z zadania 20 i zmodyfikuj ją tak, aby zawierała trzy dodatkowe kolumny z tabeli Sales.OrderDetails: productid, qty i unitprice.

-----

22. Aby lepiej zrozumieć potrzebne zadania, najpierw napisz instrukcję SELECT do tabeli HR.Employees wyświetlając kolumny empid, lastname, firstname, title i mgrid.

-----

23. Skopiuj instrukcję SELECT z zadania 22 i zmodyfikuj ją, aby zawierała dodatkowe kolumny dla informacji menedżera (lastname, firstname) za pomocą samołączenia [self-join]. Przydziel aliasy odpowiednio mgrlastname i mgrfirstname, aby odróżnić nazwy menedżerów od nazwisk pracowników.

Czy podczas self-join należy używać aliasów tabel? Czy możesz użyć pełnej nazwy tabeli źródłowej jako aliasu?

-----

24. Napisz instrukcję SELECT, aby pobrać kolumny custid i contactname z tabeli Sales.Customers i kolumny orderid z tabeli Sales.Orders. Instrukcja powinna pobrać wszystkie wiersze z tabeli Sales.Customers.

Zwróć uwagę na wartości w porządku kolumny. Czy są jakieś brakujące wartości (oznaczone jako NULL)? Czemu?

-----

25. Wykonaj kod T-SQL. Nie martw się, jeśli nie rozumiesz dostarczonego kodu T-SQL, ponieważ jest on tutaj używany w celu zapewnienia bardziej realistycznego przykładu łączenia krzyżowego w następnym zadaniu.

```
SET NOCOUNT ON;
```

```
IF OBJECT_ID('HR.Calendar') IS NOT NULL
```

```
    DROP TABLE HR.Calendar;
```

```
CREATE TABLE HR.Calendar (
```

```
    calendardate DATE CONSTRAINT PK_Calendar PRIMARY KEY
```

```
);
```

```
DECLARE
```

```
    @startdate DATE = DATEFROMPARTS(YEAR(SYSDATETIME()), 1, 1),
```

```
    @enddate DATE = DATEFROMPARTS(YEAR(SYSDATETIME()), 12, 31);
```

```

WHILE @startdate <= @enddate

BEGIN

    INSERT INTO HR.Calendar (calendardate)

    VALUES (@startdate);

    SET @startdate = DATEADD(DAY, 1, @startdate);

END;

SET NOCOUNT OFF;

GO

-- observe the HR.Calendar table

SELECT

    calendardate

FROM HR.Calendar;

```

---

26. Napisz instrukcję SELECT, aby pobrać kolumny empid, firstname i lastname z tabeli HR.Employees i kolumny Calendardate z tabeli HR.Calendar.

Jaka jest liczba wierszy zwróconych przez zapytanie? W tabeli HR.Employees znajduje się dziewięć wierszy. Spróbuj obliczyć całkowitą liczbę wierszy w tabeli HR.Calendar.

-----

27. Wykonaj dostarczoną instrukcję T-SQL, aby usunąć tabelę HR.Calendar.

```

IF OBJECT_ID('HR.Calendar') IS NOT NULL

    DROP TABLE HR.Calendar;

```

-----