

**97.3.**

Utwórz zestawienie zawierające dla każdego roku w latach 1945–2014 minimalną i maksymalną miesięczną stopę bezrobocia odnotowaną w tych latach.

Dla otrzymanego zestawienia sporządź wykres liniowy. Pamiętaj o prawidłowym i czytelnym opisie wykresu.

**97.4.**

Znajdź najdłuższy nierosnący ciąg miesięcznych stóp bezrobocia w kolejnych miesiącach w latach 1945–2014. Podaj miesiąc i rok początkowy oraz miesiąc i rok końcowy znalezione-go ciągu oraz jego długość.

**Przykład**

Dla pliku zawierającego następujące dane:

	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII
2002	12,90	12,90	12,50	13,00	12,50	12,20	11,00	11,70	11,50	11,30	11,40	11,60
2003	14,20	14,40	14,30	<b>14,40</b>	<b>13,60</b>	<b>13,20</b>	<b>13,10</b>	<b>13,00</b>	<b>13,00</b>	<b>13,00</b>	<b>12,90</b>	<b>12,80</b>
2004	<b>12,50</b>	12,60	13,30	12,90	12,60	12,30	12,30	12,40	12,40	12,50	12,90	13,40

długość najdłuższego nierosnącego ciągu stóp bezrobocia wynosi 10 (ciąg liczb: 14,40; 13,60; 13,20; 13,10; 13,00; 13,00; 13,00; 12,90; 12,80; 12,50). Miesiąc i rok początku ciągu to IV 2003. Miesiąc i rok końca ciągu to I 2004.

**97.5.**

Podaj liczbę lat, w których stopa bezrobocia w każdym miesiącu była większa od stopy bezrobocia w tym samym miesiącu roku poprzedniego.

**2.4. Bazy danych****Zadanie 98.****Wiązka zadań *Dziennik ocen***

Dane są trzy pliki tekstowe: `uczniowie.txt`, `przedmioty.txt` i `oceny.txt`, w których zapisano oceny wystawiane uczniom w pewnym technikum informatycznym w okresie od 1.09.2014 r. do 18.12.2014 r.

Pierwszy wiersz każdego z plików jest wierszem nagłówkowym, a kolumny w wierszach rozdzielone są znakami tabulacji.

Plik o nazwie `uczniowie.txt` zawiera informacje dotyczące uczniów szkoły. W każdym wierszu znajduje się: identyfikator ucznia (`ID_ucznia`), jego imię (`Imie`), nazwisko (`Nazwisko`) oraz oznaczenie klasy za pomocą rzymskiej liczby i litery (`Klasa`).

**Przykład**

Id_ucznia	Imie	Nazwisko	Klasa
123/2011	Wojciech	Banasik	IV E
124/2011	Monika	Baranowska	IV E
125/2011	Janusz	Czerwinski	IV E

Plik `przedmioty.txt` zawiera identyfikator przedmiotu (`Id_przedmiotu`) oraz nazwę przedmiotu (`Nazwa_przedmiotu`).

### Przykład

<code>Id_przedmiotu</code>	<code>Nazwa_przedmiotu</code>
1	j.polski
2	j.angielski
3	j.niemiecki

W pliku `oceny.txt` zapisane są w każdym wierszu: identyfikator oceny (`Id_oceny`), data wystawienia oceny (`Data`), identyfikator ucznia (`Id_ucznia`), identyfikator przedmiotu (`Id_przedmiotu`) oraz ocena (`Ocena`).

### Przykład

<code>Id_oceny</code>	<code>Data</code>	<code>Id_ucznia</code>	<code>Id_przedmiotu</code>	<code>Ocena</code>
1	2014-09-08	704/2014	1	2
2	2014-09-08	312/2012	1	4
3	2014-09-08	649/2013	3	5

Korzystając z danych zawartych w tych plikach oraz z dostępnych narzędzi informatycznych, wykonaj poniższe polecenia. Każdą odpowiedź umieść w pliku `wyniki.txt`, poprzedzając ją numerem odpowiedniego zadania.

#### 98.1.

Imiona dziewcząt w zestawieniu kończą się na literę „a”. Podaj klasy, w których ponad 50% wszystkich uczniów to dziewczęta.

#### 98.2.

Podaj daty, kiedy w szkole wystawiono więcej niż 10 jedynek jednego dnia.

#### 98.3.

Podaj, z dokładnością do dwóch miejsc po przecinku, średnie ocen z języka polskiego dla każdej klasy czwartej.

#### 98.4.

Podaj zestawienie zawierające dla każdego przedmiotu liczbę piątek wystawionych w kolejnych miesiącach od września do grudnia łącznie we wszystkich klasach.

#### 98.5.

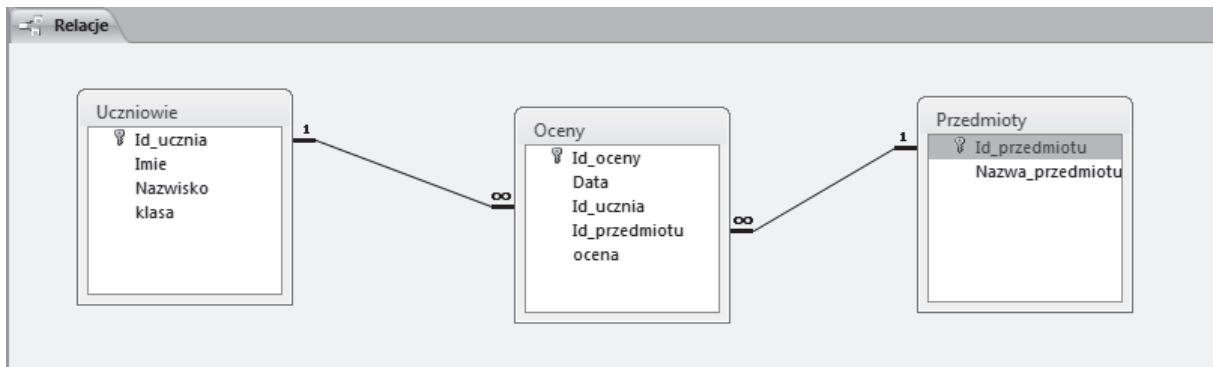
Podaj zestawienie imion i nazwisk uczniów klasy II A, którzy nie otrzymali żadnej oceny z przedmiotu sieci komputerowe.

### Komentarz do zadania

Przedstawiamy rozwiązanie zadania za pomocą programu Access z pakietu MS Office 2010 oraz w języku SQL.

Kluczami głównymi (PRIMARY KEY) w tabelach `Uczniowie`, `Oceny`, `Przedmioty` będą odpowiednio kolumny `Id_ucznia`, `Id_przedmiotu` oraz `Id_oceny`, to one jed-

noznacznie identyfikują rekordy w tabelach. Tabele połączone są relacjami typu jeden do wielu:



98.1.

Aby wyszukać klasy, w których ponad 50% wszystkich uczniów to dziewczęta, skorzystamy z informacji zawartych w tabeli Uczniowie. Utworzymy kwerendy pomocnicze, za pomocą których wyznaczmy liczbę dziewcząt (kryterium w kolumnie Imię: Like ”\*a”) w klasach (liczymy liczbę identyfikatorów uczniów) oraz liczbę wszystkich uczniów w klasach.

Pole:	klasa	PoliczOfId_ucznia1: Id_ucznia	Imię
Tabela:	Uczniowie	Uczniowie	Uczniowie
Podsumowanie:	Grupuj według	Policz	Gdzie
Sortuj:			
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Kryteria:			Like ”*a”
lub:			

Pole:	Klasa	liczba_uczniow: Id_ucznia
Tabela:	Uczniowie	Uczniowie
Podsumowanie:	Grupuj według	Policz
Sortuj:		
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kryteria:		
lub:		

W kolejnej kwerendzie dla każdej klasy obliczymy wyrażenie  $\frac{\text{liczba dziewcząt}}{\text{liczba wszystkich uczniów}}$  przy czym wyrażenie to powinno być większe od 0,5.

Pole:	Klasa	udzial_dziewczat: [pom1b]![Liczba_dziewczat]/[pom1a]![liczba_uczniow]
Tabela:	pom1a	
Sortuj:		
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kryteria:		>0,5
lub:		

Zapytanie w języku SQL:

```

SELECT Klasa
FROM Uczniowie
GROUP BY Klasa
HAVING SUM(CASE WHEN Imię LIKE “%a” THEN 1 ELSE 0 END) >COUNT(*) / 2
  
```

**98.2.**

Chcąc wyszukać daty, kiedy w szkole wystawiono więcej niż 10 jedynek jednego dnia, odfiltrujemy spośród wszystkich ocen jedynki, a następnie posłużymy się funkcjami agregującymi (policzymy dla każdej daty liczby jedynek i odfiltrujemy te dni, w których liczba jedynek przekroczyła 10).

Warunek na ocenę został założony za pomocą klauzuli WHERE. Klauzula GROUP BY umożliwia podział wierszy na kategorie na podstawie wartości w kolumnie Data i skorzystanie z funkcji grupującej (POLICZ) dla różnych ocen. Dalej klauzulą HAVING ograniczamy zestawienie do takiego, w którym liczba jedynek jest większa od 10.

Pole:	Data	ocena	ocena
Tabela:	Oceny	Oceny	Oceny
Podsumowanie:	Grupuj według	Policz	Gdzie
Sortuj:		Malejąco	
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Kryteria:		>10	1
lub:			

Zapytanie w języku SQL:

```
SELECT Oceny.Data, Count(*)
FROM Oceny
WHERE Oceny.ocena=1
GROUP BY Oceny.Data
HAVING Count(*)>10
```

**98.3.**

Aby ustalić z dokładnością do dwóch miejsc po przecinku średnie ocen z języka polskiego każdej klasy czwartej, odfiltrujemy oceny z języka polskiego uczniów klas IV, a następnie skorzystamy z funkcji agregującej Średnia (AVG).

Pole:	klasa	ocena	Nazwa_przedmiotu
Tabela:	Uczniowie	Oceny	Przedmioty
Podsumowanie:	Grupuj według	Średnia	Grupuj według
Sortuj:			
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Kryteria:	Like 'IV*'		Like 'j.polski'
lub:			

Zapytanie w języku SQL:

```
SELECT Uczniowie.klasa, AVG(oceny.ocena)
FROM Oceny
JOIN Przedmioty ON Przedmioty.id_przedmiotu = Oceny.id_przedmiotu
JOIN Uczniowie ON Uczniowie.id_ucznia = Oceny.id_ucznia
WHERE Przedmioty.nazwa_przedmiotu = 'j.polski' AND Uczniowie.klasa LIKE 'IV%'
GROUP BY Uczniowie.klasa
```

## 98.4.

W rozwiązaniu utworzymy kwerendę krzyżową, w której nagłówkiem kolumny będzie wyodrębniamy z daty numer miesiąca (funkcja MONTH), zaś nagłówkiem wiersza — nazwa przedmiotu. Jako wartość wstawiamy ocenę lub identyfikator oceny i stosujemy funkcję `Policz`. Dodatkowo należy pamiętać o klauzuli `WHERE`, za pomocą której odfiltrujemy same piątki.

Widok projektu:

Pole:	Miesiąc: Month([Ocer	Nazwa_przedmiotu	ocena	ocena
Tabela:		Przedmioty	Oceny	Oceny
Podsumowanie:	Grupuj według	Grupuj według	Policz	Gdzie
Krzyżowe:	Nagłówek kolumny	Nagłówek wiersza	Wartość	
Sortuj:				
Kryteria:				5
lub:				

Zapytanie w języku SQL:

```
SELECT Przedmioty.Nazwa_przedmiotu, Month(Oceny.Data), Count(*)
FROM Oceny
JOIN Przedmioty ON Przedmioty.id_przedmiotu = Oceny.id_przedmiotu
WHERE Oceny.ocena=5
GROUP BY Przedmioty.Nazwa_przedmiotu, Month(Oceny.Data)
```

## 98.5.

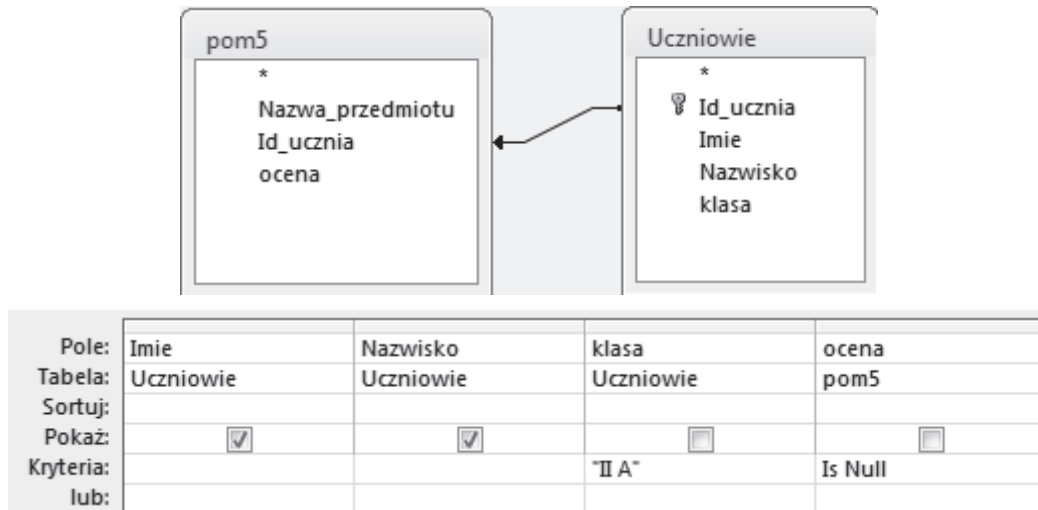
Rozwiązywanie poprzedzimy utworzeniem kwerendy pomocniczej, w wyniku której otrzymamy oceny uczniów uczęszczających na zajęcia z przedmiotu sieci komputerowe.

Widok projektu:

Pole:	Nazwa_przedmiot	Id_ucznia	ocena
Tabela:	Przedmioty	Oceny	Oceny
Sortuj:			
Pokaż:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kryteria:	Like "sieci komp**"		
lub:			

Dalej, korzystając z wyników powyższego zapytania pomocniczego oraz tabeli `Uczniowie` (złącze pomiędzy tabelami typu `RIGHT JOIN`), uzyskamy zestawienie uwzględniające wszystkie rekordy z tabeli `Uczniowie`, niezależnie od tego czy uczniowie mają ocenę, czy nie (pole wypełnione wartościami `NULL`). Kolejnym krokiem jest wybranie rekordów z wartością `NULL`. Dodatkowo w kryterium dla pola `Klasa` podajemy "II A".

Widok projektu:



Zapytanie w języku SQL:

```
SELECT Ucen.Imie, Ucen.Nazwisko
FROM Ucen
WHERE Ucen.Klasa = 'II A' AND (SELECT COUNT(*) FROM Oceny JOIN
Przedmioty ON Przedmioty.id_przedmiotu = Oceny.id_przedmiotu WHERE
Oceny.id_ucznia = Ucen.id_ucznia AND Przedmiotu.nazwa LIKE 'sieci komp%') = 0
```

### Rozwiązanie

98.1.

I A, I C

98.2.

2014-11-11

2014-10-14

98.3.

Klasa	Średnia
IV A	3,53
IV B	3,45
IV C	3,40
IV D	3,90
IV E	3,51

98.4.

Przykładowa poprawna odpowiedź:

Nazwa przedmiotu	9	10	11	12
administracja bazami danych	3	13	7	1
administracja sieciowymi systemami operacyjnymi				2
biologia	8	20	9	5
chemia	18	31	31	13
diagnostyka i naprawa urzadzen techniki komputerowej	8	13	9	13
edukacja dla bezpieczenstwa	9	15	14	8
fizyka	14	21	31	9
geografia		1	1	
historia	2	2		
historia i spoleczenstwo - przedmiot uzupelniajacy	8	4	6	3
informatyka	33	31	34	25
j.angielski	50	85	72	50
j.niemiecki	66	106	84	53
j.polski	37	81	62	41
jezyk angielski zawodowy w branzy informatycznej	1	1	1	
matematyka	32	39	48	35
podstawy przedsiebiorczosci	6	16	16	12
projektowanie i montaz lokalnych sieci komputerowych	10	11	11	6
sieci komputerowe		2		
systemy baz danych	6	5	8	5
systemy operacyjne	4	19	12	9
urzadzenia techniki komputerowej	8	18	19	15
wiedza o kulturze	7	16	13	10
wiedza o spoleczenstwie	7	15	9	7
witryny i aplikacje internetowe	5	6	9	3
wychowanie fizyczne	23	39	34	15

**98.5.**

Imie	Nazwisko
Aneta	Duda
Mirosław	Gorski
Lukasz	Kostoczko
Donald	Krychowski
Adrian	Lubaczewski
Piotr	Nawrocki
Michał	Nowakowski
Piotr	Prusinski
Grzegorz	Tomkow
Radosław	Wojciechowski
Mariusz	Wojtyra