

# Porządkowanie danych za pomocą makr i formuł w Excelu



# Porządkowanie danych za pomocą makr i formuł w Excelu



Autorzy  
Piotr Dynia, Mariusz Kowalski, Robert Kuźma

Redaktor prowadzący  
Rafał Janus

Wydawca  
Monika Kijok

Opracowanie graficzne okładki  
Piotr Fedorczyk

Opracowanie graficzne  
Zbigniew Korzański

Koordynator produkcji  
Mariusz Jezierski

Korekta  
Zespół

ISBN 978-83-269-3307-3

Nakład: 1000 egz.

Wydawnictwo Wiedza i Praktyka sp. z o.o.

03-918 Warszawa, ul. Łotewska 9a

Tel. 22 518 29 29, faks 22 617 60 10, e-mail: rjanus@wip.pl

NIP: 526-19-92-256

Numer KRS: 0000098264 – Sąd Rejonowy dla m.st. Warszawy, Sąd Gospodarczy  
XIII Wydział Gospodarczy Rejestrowy. Wysokość kapitału zakładowego: 200.000 zł

Skład i łamanie: Triograf Dariusz Kołacz

Miller Druk Sp. z o. o.,

03-301 Warszawa, ul. Jagiellońska 82, tel.: 22 614 17 67

Copyright © by Wydawnictwo Wiedza i Praktyka sp. z o.o.  
Warszawa 2014

„Porządkowanie danych za pomocą makr i formuł w Excelu” wraz z przysługującym Czytelnikom innymi elementami dostępnymi w subskrypcji (e-letter, strona WWW i inne) chronione są prawem autorskim. Przedruk materiałów opublikowanych w publikacji „Porządkowanie danych za pomocą makr i formuł w Excelu” oraz w innych dostępnych elementach subskrypcji – bez zgody wydawcy – jest zabroniony. Zakaz nie dotyczy cytowania publikacji z powołaniem się na źródło.

Publikacja „Porządkowanie danych za pomocą makr i formuł w Excelu” została przygotowana z zachowaniem najwyższej staranności i wykorzystaniem wysokich kwalifikacji, wiedzy i doświadczenia autorów oraz konsultantów. Zaproponowane w książce „Porządkowanie danych za pomocą makr i formuł w Excelu” oraz w innych dostępnych elementach subskrypcji wskazówki, porady i interpretacje nie mają charakteru porady prawnej. Ich zastosowanie w konkretnym przypadku może wymagać dodatkowych, pogłębionych konsultacji. Publikowane rozwiązania nie mogą być traktowane jako oficjalne stanowisko organów i urzędów państwowych. W związku z powyższym redakcja nie może ponosić odpowiedzialności prawnej za zastosowanie zawartych w publikacji „Porządkowanie danych za pomocą makr i formuł w Excelu” lub w innych dostępnych elementach subskrypcji wskazówek, przykładów, informacji itp. do konkretnych przykładów.

Wydawca nie odpowiada za treść zamieszczonej reklamy; ma prawo odmówić zamieszczenia reklamy, jeżeli jej treść lub forma są sprzeczne z linią programową bądź charakterem publikacji oraz interesem Wydawnictwa Wiedza i Praktyka.

**Sprawdzanie poprawności wprowadzanych danych**

Wprowadzanie danych z ograniczeniami .....	6
Inne zastosowania sprawdzania poprawności danych .....	9

**Organizacja i porządkowanie danych za pomocą makr**

Część I. Wprowadzenie .....	14
Część II. Makra pomocne przy pracy z komórkami scalonymi .....	34
Część III. Makra pomocne przy zaznaczaniu określonych komórek .....	73
Część IV. Inne pomocne makra .....	103

**Przygotowanie danych do analizy z wykorzystaniem****funkcji tekstowych**

Zestawienie dotyczące zatrudnienia .....	120
Format zwróconych wyników .....	124
Uniwersalna formuła do wyodrębniania danych .....	125

**Zmiana układu tabeli z wykorzystaniem formuł**

Mało przejrzysty układ listy kontraktów .....	128
Tabela posortowana malejąco według wielkości kontraktów .....	131
Zmiana układu innych kategorii danych z zestawienia bazowego .....	138

**Szybsze obliczenia dzięki podziałowi tabeli na zakresy**

Arkusz wyjściowy – tygodniowa sprzedaż poszczególnych produktów .....	142
Określenie obszarów działania formuły .....	147
Rozszerzenie zakresu danych źródłowych .....	153

**Dynamiczne sortowanie danych w Excelu**

Automatyczne sortowanie list o zmiennej zawartości .....	156
--	-----

**Automatyczne generowanie posortowanej kopii tabeli**

Arkusz do autosortowania tabel .....	168
Zmiana porządku sortowania .....	175
Ochrona przed powtórzeniami .....	177

**Analiza długich list - funkcje bazy danych**

Wyszukiwanie z użyciem funkcji DB.POLE .....	180
--	-----

**Korespondencja seryjna w Office 2010 na przykładach**

Spersonalizowane dokumenty seryjne .....	188
Seryjne nadruki adresowe na kopertach .....	203
Automatyczna wysyłka wiadomości e-mail .....	209

**Automatyzacja seryjnych wydruków z wykorzystaniem makr**

Utworzenie tabeli seryjnej .....	220
Definiowanie wszystkich wartości zmiennych .....	222
Dopasowywanie kodu makra .....	226

**Ochrona wybranych zakresów komórek**

Blokowanie dostępu do komórek .....	230
-------------------------------------	-----

**Usprawnienie częstego filtrowania za pomocą makr**

Samodzielne przygotowanie arkusza .....	237
---	-----

Typowym wyzwaniem podczas pracy z Excelem jest uporządkowanie danych i doprowadzenie ich do postaci, w której nadają się do właściwej obróbki. Taka sytuacja zdarza się bardzo często przy importowaniu danych z zewnętrznych źródeł, ale także, gdy przychodzi analizować tabele zaprojektowane przez współpracowników. W wielu przypadkach porządkowanie danych czy przebudowa tabel wiąże się ze żmudnymi i czasochłonnymi operacjami. Niniejsza książka prezentuje wiele rozwiązań, dzięki którym uprościsz te niewdzięczne zadania i ograniczysz do minimum czas potrzebny na ich wykonanie.

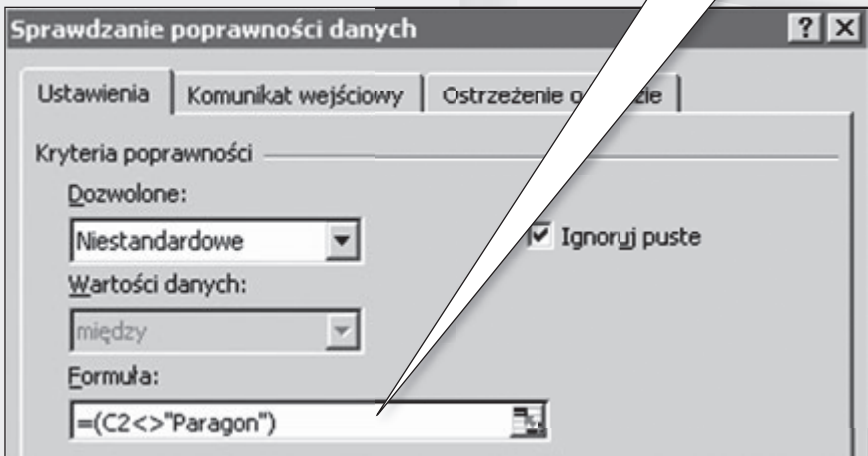
Użytkownicy poszukują w Excelu sposobów, które ułatwiłyby pracę, np. zamieniały liczby rzymskie na arabskie, zaznaczyły puste wiersze, automatyzowały seryjne wydruki czy kopiowały za jednym razem dany arkusz określoną liczbę razy. Niestety czasami czeka nas rozczarowanie, ponieważ Excel nie oferuje takich poleceń i te operacje trzeba wykonać krok po kroku. Pochłania to czas i jest źródłem wielu błędów, np. użytkownik może zaznaczyć wiersz, w którym były ukryte ważne dane.

Od teraz nie będziesz już tracić niepotrzebnie czasu i nerwów na takie zadania. W tej książce pokazujemy, jak napisać makra, które bezbłędnie wykonają żmudne zadania. Poznasz 20 makr, które nie mają swoich odpowiedników w poleceniach Excela i na pewno będziesz mieć okazję z nich skorzystać w pracy ze swoimi skoroszytami. Ponadto dowiesz się, gdzie przechowywać makra, jak je uruchamiać, a także jak dostosować opisane narzędzia do swoich potrzeb.

# Sprawdzanie poprawności wprowadzanych danych

**B**udujemy niekiedy w Excelu szablony zestawień, które później będą wypełniane przez innych użytkowników. Poza wpisaniem nagłówków kolumn/wierszy czy ustawieniem kolorystyki określamy również ograniczenia wprowadzanych liczb i dat. Ciekawym udoskonaleniem takich szablonów może być rzadko stosowane ograniczenie możliwości wprowadzania danych, w zależności od zawartości innych komórek – wcześniej wypełnionych. O tym, jak tworzyć takie szablony, przeczytamy w tym haśle. Dowiemy się również:

Excel udostępnia bardzo przyjazne użytkownikowi narzędzie do kontrolowania danych wpisywanych do arkusza.



### WPROWADZANIE DANYCH Z OGRANICZENIAM



Potrzeba blokowania komórek czy określania poprawnych wpisów niekiedy pojawi się w wielu zadaniach. W celu uniknięcia błędów w skoroszytach Excela można ograniczyć możliwość wpisywania danych w taki sposób, aby zostały spełnione określone warunki. Takie działania będą wymuszały wprowadzanie poprawnych danych. Przykładowo o odpowiednich wartościach wynikających z obliczeń lub mieszczących się w zadanych przedziałach czy w odpowiednich formatach (np. aby w kolumnach z datami czy kwotami uniemożliwić wpisywanie tekstu, a tam, gdzie dozwolone jest jego wprowadzanie, określić długość, np. powyżej 10 znaków).

#### Ograniczenie uzależnione od innej komórki

Jeżeli projektujemy szablon przeznaczony do rejestrowania dokumentów potwierdzających transakcje sprzedaży (na przykład mający formę podobną do przedstawionego na rysunku 1), możemy ustawić mechanizm blokujący możliwość wprowadzenia nazwy firmy w przypadku wystawiania paragonu. Do ustawienia takiego ograniczenia wykorzystajmy narzędzie *Sprawdzanie poprawności*.



Aby to zrobić:

1. Zaznaczamy komórkę D2.
2. Z menu *Dane* wybieramy polecenie *Sprawdzanie poprawności* (w Excelu 2007/2010: uaktywniamy kartę *Dane* i w grupie poleceń *Narzędzia danych* wskazujemy *Poprawność danych*).

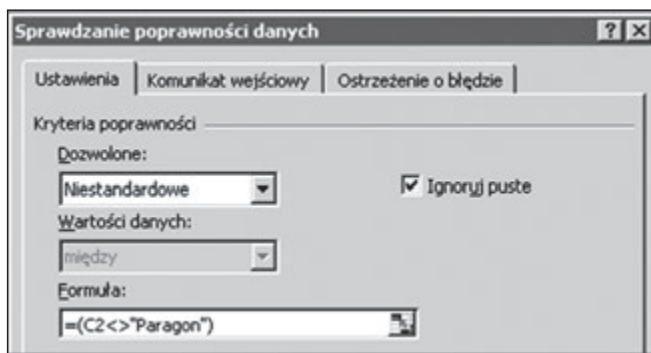
## Sprawdzanie poprawności wprowadzanych danych

	A	B	C	D
1	Lp.	Data	Forma dokumentu	Firma
2	1	2006-01-10	Paragon	
3	2	2006-01-11	Faktura	Alda Sp. z o.o.
4	3	2006-01-12	Paragon	
5	4	2006-01-13	Faktura	Gemin S.J.
6	5	2006-01-16	Faktura	Alex S.C.
7	6	2006-01-17	Paragon	
8	7	2006-01-18	Faktura	Zewir S.A.
9	8	2006-01-19	Faktura	Karid Sp. z o.o.
10	9	2006-01-20	Paragon	
11	10	2006-01-23	Paragon	
12	11	2006-01-24	Paragon	

Rys. 1. Rejestr transakcji

- Przechodzimy do zakładki *Ustawienia* i w polu *Dozwolone* wybieramy *Niestandardowe* (rys. 2).
- W polu *Formuła* wpisujemy:

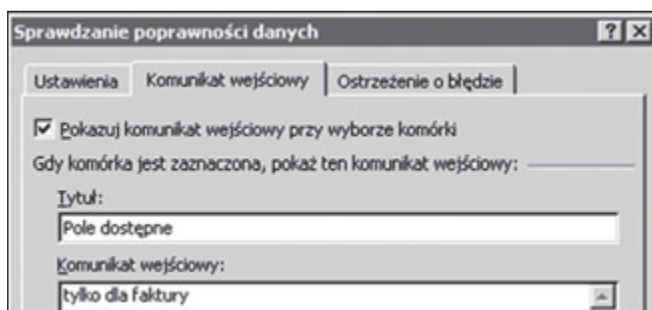
`=(C2<>"Paragon")`



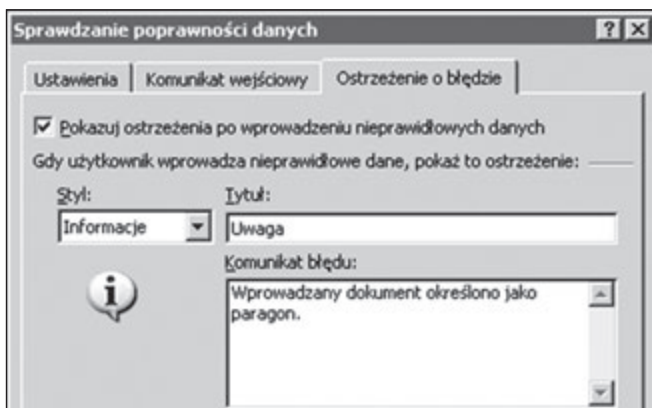
Rys. 2. Ustawienia sprawdzania poprawności danych



- Komunikaty**
- Jeżeli chcemy, aby użytkownik był informowany o ograniczeniach tego pola, możemy zdefiniować komunikat, który będzie pojawiał się przy zaznaczeniu komórek objętych sprawdzaniem poprawności. W tym celu wybieramy zakładkę *Komunikat wejściowy* i wpisujemy odpowiedni tekst (rys. 3).



Rys. 3. Ustawienia treści komunikatu



Rys. 4. Ustawienia ostrzeżenia o błędzie

- Możemy również określić, jaki komunikat będzie wyświetlany w przypadku próby wpisania niedozwolonych danych. W tym celu klikamy zakładkę *Ostrzeżenie o błędzie* i wpisujemy odpowiadający nam komunikat (rys. 4).
- Po zatwierdzeniu przedstawionych ustawień pojawiać się będą komunikaty przedstawione na rysunku 5.

	A	B	C	D	E
1	Lp.	Data	Forma dokumentu	Firma	
2	1	2006-01-10	Paragon	ABC	
3	2	2006-01-11	Faktura	Alda Sp. z	
4	3	2006-01-12	Paragon		
5	4	2006-01-13	Faktura	Gemin S.J.	
6	5				
7	6	Uwaga			
8	7	Wprowadzany dokument określono jako paragon.			
9	8	OK Anuluj			
10	9				
11	10				
12	11				
13					

Rys. 5. Widok arkusza przy próbie wpisania niedozwolonej wartości

- Zawartość komórki D2 skopiujemy do pozostałych w kolumnie D.

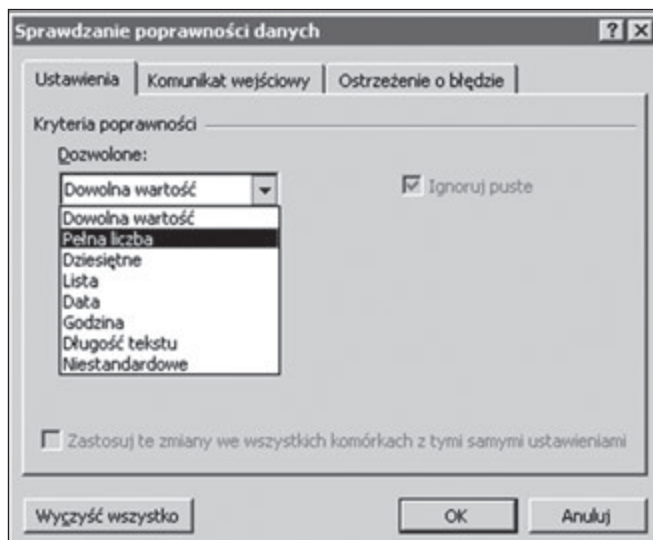
## INNE ZASTOSOWANIA SPRAWDZANIA POPRAWNOŚCI DANYCH

W podobny sposób możemy wprowadzić inne ograniczenia wprowadzania danych.

### Ograniczenie wprowadzanych liczb mieszczących się w wyznaczonym przedziale

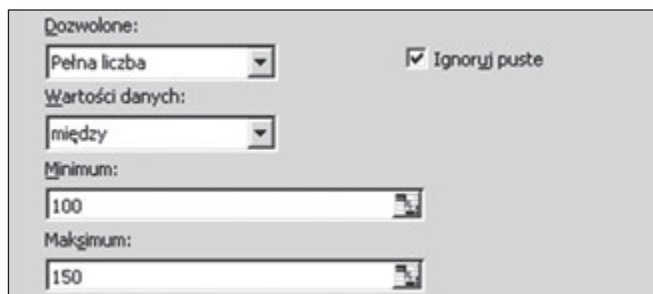
- W polu *Dozwolone* wybieramy *Pełna liczba* lub *Dziesiętne*.





Rys. 6. Wybór warunków sprawdzania poprawności danych

2. W polu *Dane* klikamy wymagany typ ograniczenia. Aby na przykład wyznaczyć granicę górną i dolną, klikamy opcję *między*.
3. Wprowadzamy minimalną, maksymalną lub określoną wartość, która będzie dozwolona.



Rys. 7. Wpisywane liczby będą musiały mieścić się w przedziale

### Ograniczenie wprowadzanych danych do określonych na liście

1. W polu *Dozwolone* wybieramy *Lista*.
2. Klikamy pole *Źródło*.
3. Wpisujemy wartości listy rozdzielone średnikami lub zaznaczamy komórki w arkuszu, a następnie naciskamy klawisz *Enter*.

### Ograniczenie wprowadzanych danych do określonego przedziału czasowego

1. W polu *Dozwolone* wybieramy *Data* lub *Godzina*.
2. W polu *Dane* określamy typ ograniczenia.
3. Wprowadzamy początkową, końcową lub określoną datę albo godzinę, która będzie dozwolona.

### Ograniczenie długości wprowadzanego tekstu

1. W polu *Dozwolone* wybieramy *Długość tekstu*.
2. W polu *Dane* wybieramy typ ograniczenia. Aby zezwolić na stosowanie pewnej maksymalnej liczby znaków, klikamy opcję *mniejsze niż lub równa*.
3. Wprowadzamy minimalną, maksymalną lub określoną długość tekstu.

Zarówno w tych przypadkach, jak i w opisanych we wstępie również możemy określić odpowiednie komunikaty i ostrzeżenia.



The image shows a screenshot of the 'Data Validation' dialog box in Microsoft Excel, specifically the 'Settings' tab. The 'Allow' dropdown is set to 'Text length'. The 'Criteria' dropdown is set to 'less than or equal to'. The 'Maximum' field contains the number '10'. There is a checked checkbox for 'Ignore empty cells'.

Rys. 8. Dozwolona długość tekstu

### Warto zapamiętać!

- ✓ Za pomocą narzędzia *Sprawdzanie poprawności* możemy łatwo wyeliminować błędy interpretacyjne czy literówki, często pojawiające się przy wprowadzaniu dużej partii danych.
- ✓ Dzięki zastosowaniu reguł poprawności mamy w zasadzie zagwarantowane poprawne wpisanie danych z równoczesnym przedstawieniem informacji na temat ograniczeń, które zostały wprowadzone do szablonu.

## Organizacja i porządkowanie danych za pomocą makr

**W**iele razy poszukujemy w Excelu poleceń, które ułatwiłyby nam pracę, np. zaznaczyły puste wiersze czy skopiowały za jednym razem dany arkusz określoną liczbę razy. Niestety czasami czeka nas rozczarowanie, ponieważ program Excel nie ma takich poleceń. Jesteśmy zmuszeni takie operacje wykonać krok po kroku, np. zaznaczać pojedyncze puste wiersze czy kilkanaście razy wykonywać tę samą operację kopiowania. Nie dość, że marnujemy czas na żmudne operacje, to jeszcze możemy popełnić błąd, np. zaznaczyć wiersz, w którym były ukryte ważne dane. Od teraz już nie będziemy tracić niepotrzebnie czasu i nerwów na takie zadania. W tym haśle sprawdzimy, jak napisać makra, które bezbłędnie wykonają takie operacje. Poznamy 20 makr, które nie mają swoich odpowiedników w poleceniach Excela i na pewno będziemy mieli okazję z nich skorzystać w pracy ze swoimi skoroszytami. Ponadto dowiemy się, gdzie przechowywać makra, jak je uruchamiać, a także jak dostosować opisane narzędzia do swoich potrzeb.



## CZEŚĆ I. WPROWADZENIE

### ■ JAK NAPISAĆ MAKRO



Makra to kod programowania napisany w języku Visual Basic, który pozwala na rozszerzenie funkcjonalności Excela oraz na automatyzację zadań. Na następnych stronach przedstawimy podstawowe informacje o tym, jak pisać, modyfikować i uruchamiać makra.



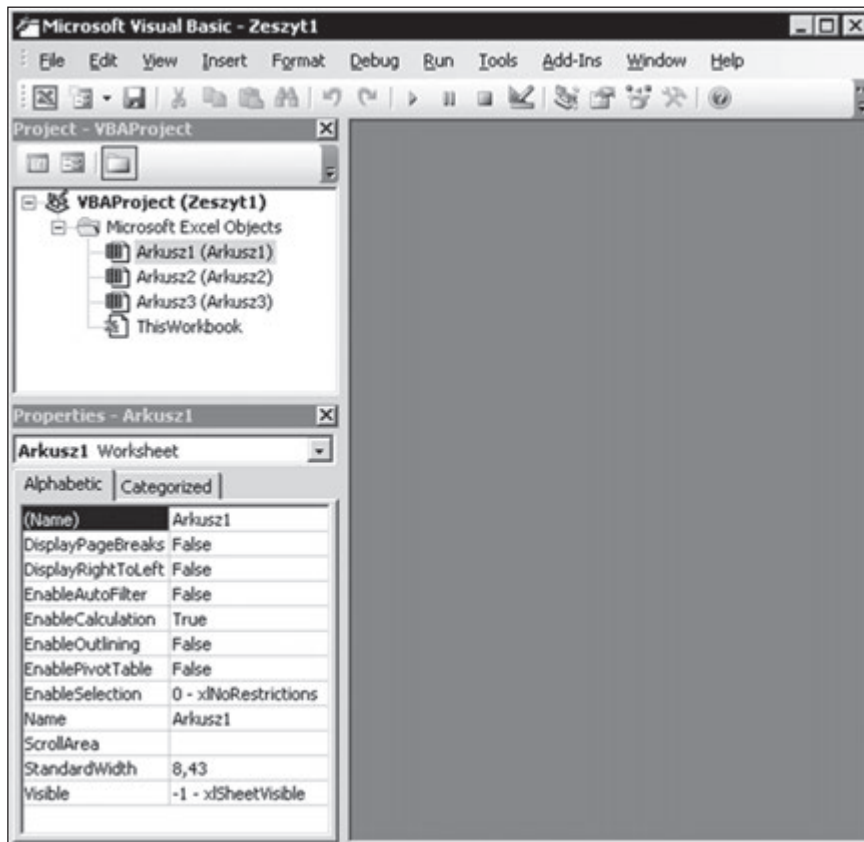
Aby napisać makro:

1. Otwieramy skoroszyt Excela, w którym chcemy wykorzystywać makro, i wybieramy polecenie menu *Narzędzia/Makro/Edytor Visual Basic* (w Excelu 2007/2010: na karcie *Deweloper*, w grupie poleceń *Kod* klikamy *Visual Basic*), aby otworzyć Edytor Visual Basic.

W efekcie pojawi się okno Edytora widoczne na rysunku 1.

Przed przystąpieniem do pisania kodu makr zalecane jest ustawienie dwóch pomocnych opcji. Wybieramy polecenie menu *Tools/Option* i w zakładce *Editor* zaznaczamy opcję *Require Variable Declaration*, a w polu *Tab Width* wpisujemy liczbę 2. Pierwsza z nich spowoduje automatyczne wpisywanie instrukcji *Option Explicit* na początku każdego nowo tworzonego modułu kodu, co pozwoli uniknąć literówek. Druga zmniejsza wcięcie dla bloku instrukcji, dzięki czemu zmniejszy się szerokość kodu.

Wprowadzone zmiany będą dotyczyły wszystkich skoroszytów. Makra w tej publikacji są pisane przy zastosowaniu obu opcji.

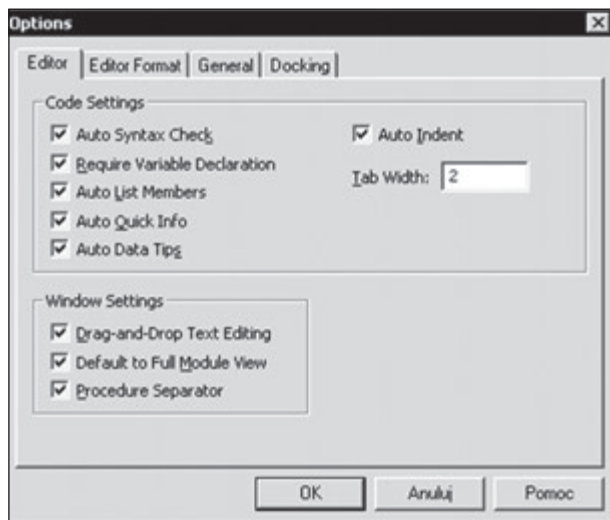


Rys. 1. Okno Edytora Visual Basic

! Makra powinniśmy przechowywać w module standardowym, choć będą działać także, jeśli wpisujemy je w każdym innym module.

2. Wybieramy polecenie menu *Insert/Module*, aby wstawić nowy moduł standardowy.





Rys. 2. Zalecane ustawienia Edytora Visual Basic



Rys. 3. Kod modułu standardowego po otwarciu

3. Pojawi się okno kodu modułu widoczne na rysunku 3. W oknie kodu, pod instrukcją *Option Explicit* wpisujemy przykładowy kod makra widoczny na rysunku 4.

```
Sub MojeMakro()  
    Dim nazwa As String  
    Dim data As Date  
  
    nazwa = Application.UserName  
    data = Now  
    MsgBox "Makro zostało uruchomione przez: " & nazwa & vbCrLf & data  
End Sub
```

Rys. 4. Kod przykładowego makra



Po wpisaniu pierwszej linii zostanie automatycznie wpisana ostatnia linia makra. Uzupełniamy więc pozostałe instrukcje.

Przedstawione makro składa się z dwóch zmiennych o nazwach: *nazwa* i *data*, do których przypisywane są odpowiednio: nazwa użytkownika pakietu Office oraz aktualna data. Na końcu makra wyświetlana jest informacja o tym, kto i o której godzinie uruchomił makro.

4. Teraz możemy zamknąć okno Edytora Visual Basic lub wybrać polecenie menu *File/Close and Return to Microsoft Excel*.

## JAK UTWORZYĆ MAKRO W SKOROSZYCIE MAKR OSOBISTYCH

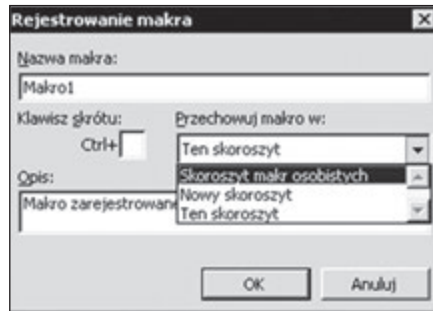
W poprzednim rozdziale pokazaliśmy, jak napisać makro wykorzystywane w skoroszycie, w którym zostało

wstawione. W przypadku gdy chcemy używać tego samego narzędzia w wielu plikach, to możemy umieścić je w skoroszycie makr osobistych. Zawsze ma on taką samą nazwę: Personal.xls (w Excelu 2007/2010: *Personal.xlsm*) i przy uruchamianiu Excela jest zawsze otwierany, ale jako ukryty. Skoroszyt ten nie jest domyślnie dostępny w Excelu.



Aby go utworzyć:

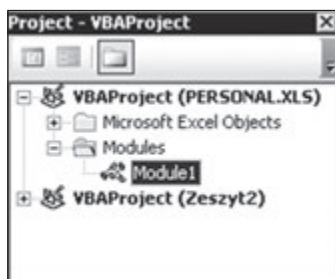
1. Wybieramy polecenie menu *Narzędzia/Makro/Zarejestruj nowe makro* (w Excelu 2007/2010: na karcie *Developer*, w grupie poleceń *Kod* klikamy *Zarejestruj makro*).
2. W oknie *Rejestrowanie makra*, w polu *Przechowuj makro w* wybieramy *Skoroszyt makr osobistych* i naciskamy *OK*.



Rys. 5. Rejestrowanie nowego makra w skoroszycie makr osobistych

3. Pojawi się pasek narzędzi *Zatrzymaj rejestrowanie*. Naciskamy na nim przycisk *Zatrzymaj rejestrowanie* (w Excelu 2007/2010: na karcie *Developer*, w grupie poleceń *Kod* wybieramy *Zatrzymaj rejestrowanie*).

4. Otwieramy Edytor Visual Basic. W okienku *Project* – *VBAProject*, w drzewku projektów będzie widoczny skoroszyt *Personal.xls* (w Excelu 2007/2010: *Personal.xlsx*). Wskazujemy znak + po lewej stronie okna, aby rozwinąć projekt tego skoroszytu, a następnie klikamy dwukrotnie moduł *Module1*.



Rys. 6. Okienko drzewka projektów VBA ze skoroszytem *Personal.xls*

5. Zostanie otwarte okno kodu modułu z zarejestrowanym makrem. Usuwamy widoczny kod i wpisujemy kod swojego makra.
6. Wybieramy polecenie menu *File/Save Personal.xls* (w Excelu 2007/2010: *Save Personal.xlsx*), a następnie zamykamy Edytor Visual Basic.

Jeśli będziemy chcieli utworzyć kolejne makro w skoroszycie makr osobistych, to postępujemy analogicznie, począwszy od kroku 4. Nie usuwamy kodu w kroku 5 i wpisujemy kolejne makra pod ostatnią linią. Pamiętajmy przy tym, że nie możemy w jednym module mieć dwóch tak samo nazwanych makr.

## JAK UTWORZYĆ DODATEK ZAWIERAJĄCY MAKRA

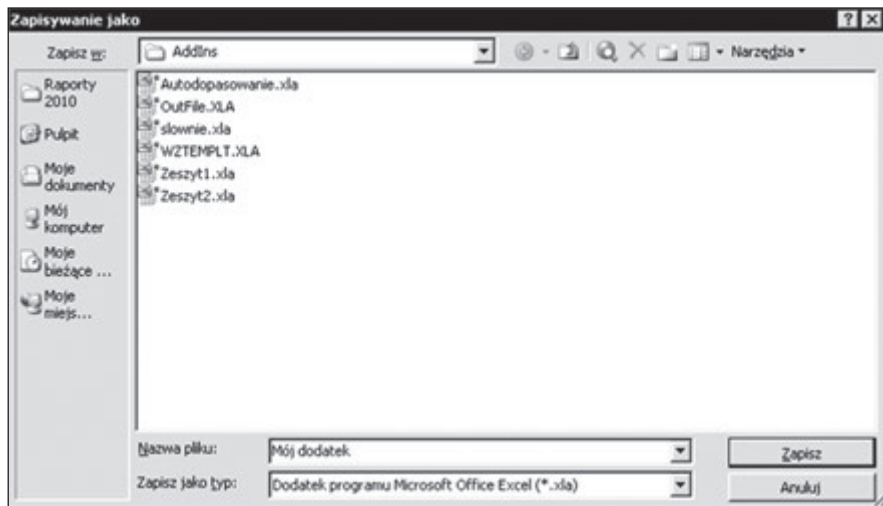
Innym sposobem przechowywania makr wykorzystywanych przy wielu skoroszytach jest utworzenie dodatku z makrami.



W tym celu:

1. Tworzymy nowy skoroszyt za pomocą polecenia *Plik/Nowy* (w Excelu 2007/2010: naciskamy *Przycisk pakietu Office/Nowy*, a następnie w oknie *Nowy skoroszyt* zaznaczamy *Pusty skoroszyt* i naciskamy przycisk *Tworzymy*).
2. Otwieramy Edytor Visual Basic i wybieramy polecenie menu *Insert/Module*, aby wstawić moduł standardowy.
3. W oknie kodu, które się pojawi, wpisujemy kod swojego makra.
4. Zalecane jest założenie hasła ochrony projektu. Jeśli chcemy zabezpieczyć swój kod, to w oknie Edytora VB wybieramy polecenie menu *Tools/VBAProject Properties*, w zakładce *Protection*, w polu *Password* podajemy hasło ochrony projektu, w polu *Confirm Password* je potwierdzamy, a następnie naciskamy *OK*.
5. Zamykamy okno Edytora Visual Basic. W oknie Excela wybieramy polecenie menu *Plik/Zapisz jako* (w Excelu 2007/2010: wybieramy *Przycisk pakietu Office/Zapisz jako/Inne formaty*), w polu *Nazwa pliku* podajemy nazwę swojego dodatku, a poniżej w polu *Zapisz jako typ* wskazujemy *Dodatek programu Excel (\*.xla)* [w Excelu 2007/2010: *Dodatek programu Excel 97–2003 (\*.xla)* lub *Dodatek programu Excel (\*.xlam)*]. Naciskamy przycisk *Zapisz*.

Dodatki zapisane w wersji Excel 2007 z rozszerzeniem \*.xlam nie mogą być wykorzystane w starszych wersjach arkusza kalkulacyjnego. Jeśli użytkownicy naszego dodatku będą używali różnych wersji Excela, to lepiej utworzyć dodatek w wersji 97–2003 z rozszerzeniem \*.xla, który będzie mógł być wykorzystywany również w nowszych wersjach programu.

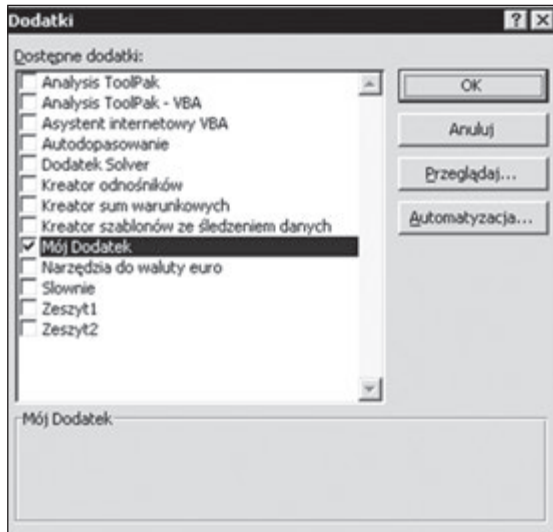


Rys. 7. Zapisywanie pliku jako dodatku

6. Zamykamy wszystkie otwarte skoroszyty i aplikację Excela, a następnie ponownie otwieramy arkusz kalkulacyjny.
7. Wybieramy polecenie menu *Narzędzia/Dodatki* (w Excelu 2007/2010: klikamy *Przycisk pakietu Office* i naciskamy przycisk *Opcje programu Excel*,

przechodzimy do zakładki *Dodatki*, na dole wybieramy *Dodatki programu Excel* i klikamy przycisk *Przechodzimy*).

8. W oknie dialogowym *Dodatki* zaznaczamy pole obok swojego dodatku i wskazujemy *OK*.



Rys. 8. Załadowanie dodatku

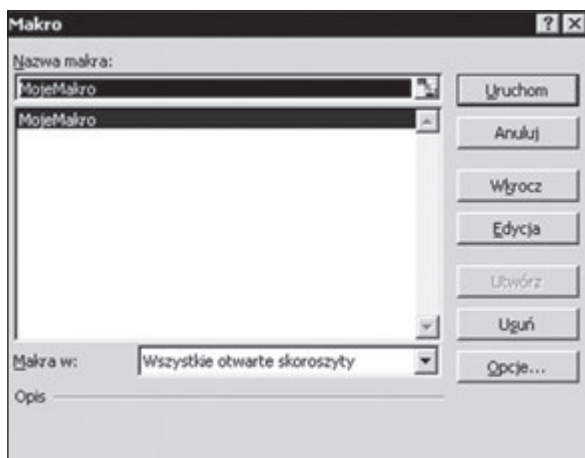
Teraz we wszystkich otwartych skoroszytach będziemy mogli wykorzystać makra zawarte w dodatku. Aby wyłączyć dodatek, przechodzimy do okna dialogowego *Dodatki*, obok swojego dodatku wyczyścimy zaznaczenie i naciskamy przycisk *OK*.

### JAK URUCHOMIĆ MAKRO

Jeśli już zapisaaliśmy kod makra w skoroszycie lub dodatku, to możemy uruchomić swoje narzędzie.

W tym celu:

1. Wybieramy polecenie menu *Narzędzia/Makro/Makra* (w Excelu 2007/2010: na karcie *Deweloper*, w grupie poleceń *Kod* lub na karcie *Widok*, w grupie poleceń *Makra* klikamy *Makra*).
2. Pojawi się okno dialogowe *Makro* z listą dostępnych makr. Na liście wybieramy makro, które chcemy uruchomić, i naciskamy przycisk *Uruchom*.



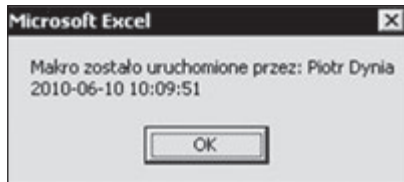
Rys. 9. Okno dialogowe służące do uruchamiania makr

Makra umieszczone w dodatkach nie będą widoczne na liście makr w tym oknie, ale po wpisaniu poprawnej nazwy takiego makra będą dostępne wszystkie przyciski po prawej stronie okna, ponieważ nazwy makr zawartych w dodatkach są ukryte.

Makro zostanie uruchomione. Kod makra przedstawiony na rysunku 4 ma za zadanie wyświetlać informacje



o tym, kto i o której godzinie uruchomił to makro. Okno komunikatu wyświetlanego przez to makro jest przedstawione na rysunku 10.



Rys. 10. Okno komunikatu wyświetlane przez makro

3. Naciskamy przycisk *OK*, aby zamknąć okno komunikatu wyświetlanego przez makro.

Uruchamianie makr w ten sposób jest dosyć skomplikowane. W następnych rozdziałach przedstawimy Wam inne sposoby uruchamiania makr w prostszy sposób.

Jeśli będziemy chcieli w przyszłości zmienić kod makra, to w oknie dialogowym *Makro* zaznaczamy jego nazwę i naciskamy przycisk *Edycja*.

## URUCHAMIANIE MAKR ZA POMOCĄ SKRÓTU KLAWIATUROWEGO

Do swojego makra możemy przypisać skrót klawiaturowy. Dzięki temu swoje narzędzie Visual Basic będziemy uruchamiali po wybraniu odpowiedniej kombinacji klawiszy na klawiaturze.