

Mateusz Boryga

# Makra

*i*



# w tydzień

Odkryj **potęgę** programowania!

Helion 

Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz wydawca dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz wydawca nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redaktor prowadzący: Tomasz Gojowy

Projekt okładki: Studio Gravite/Olsztyn  
Obarek, Pokoński, Pazdrijowski, Zaprucki

Materiały graficzne na okładce zostały wykorzystane za zgodą Shutterstock.

Helion S.A.

ul. Kościuszki 1c, 44-100 Gliwice  
tel. 32 230 98 63

e-mail: [helion@helion.pl](mailto:helion@helion.pl)

WWW: <https://helion.pl> (księgarnia internetowa, katalog książek)

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

<https://helion.pl/user/opinie/janipr>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

Materiały do książki dostępne są pod adresem:

<https://ftp.helion.pl/przyklady/janipr.zip>

ISBN: 978-83-289-1343-1

Copyright © Helion S.A. 2024

Printed in Poland.

- Kup książkę
- Poleć książkę
- Oceń książkę

- Księgarnia internetowa
- Lubię to! » Nasza społeczność

# SPIIS TREŚCI

## **ROZDZIAŁ 1.**

### **AUTORSKI SCHEMAT DZIAŁANIA Z KAŻDYM WYZWANIEM W VBA \_\_\_\_\_ 7**

- 1.1. Rejestrator makr (godnie używany) da Ci dużo  
dobrego kodu \_\_\_\_\_ 8
- 1.2. Świat obiektów Excela \_\_\_\_\_ 19
- 1.3. Wzorzec matka-córki, czyli o recyklingu kodu  
słów kilka \_\_\_\_\_ 22
- 1.4. Tworzysz UserForm i wszyscy Cię podziwiają! \_\_\_\_\_ 24
- 1.5. Autorski schemat działania z każdym wyzwaniem  
w VBA \_\_\_\_\_ 30

## **ROZDZIAŁ 2.**

### **FRAGMENTY KODU VBA, KTÓRE W DZISIEJSZYM ŚWIECIE WYPADA ZNAĆ \_\_\_\_\_ 33**

- 2.1. Zaznaczenie wymarzonej kolumny \_\_\_\_\_ 35
- 2.2. Zamiana danych \_\_\_\_\_ 39
- 2.3. Zamiana liczby występującej w formie tekstu  
na liczbę \_\_\_\_\_ 42

|  |    |
|--|----|
| 2.4. Wstawienie formuły „w miejscu”                    | 47 |
| 2.5. Wstawienie formuły na końcu arkusza               | 53 |
| 2.6. Przejście do początku arkusza                     | 57 |
| 2.7. Zmiana wielkości zaznaczenia                      | 59 |
| 2.8. Formatowanie                                      | 61 |
| 2.9. Tabele  | 64 |
| 2.10. Tabele przestawne                                | 69 |
| 2.11. Przeniesienie wybranych danych do nowego arkusza | 73 |
| 2.12. Otwieranie pliku                                 | 77 |
| 2.13. Wybór pliku                                      | 78 |
| 2.14. Zamykanie pliku                                  | 79 |
| 2.15. Przejście po plikach w folderze                  | 80 |
| 2.16. Wybór folderu z listy                            | 82 |
| 2.17. Wybór pliku z listy                              | 84 |
| 2.18. Wysłanka pliku Excela jako maila w Outlooku      | 86 |

### ROZDZIAŁ 3.

## „SMACZKI” ZMIENIAJĄCE ŻYCIE 89

|   |     |
|---|-----|
| 3.1. Wyświetlanie na ekranie?                 | 90  |
| 3.2. Komunikaty na pasku dolnym               | 91  |
| 3.3. Zakres podawany przez użytkownika        | 92  |
| 3.4. Czy znaleźliśmy tę komórkę?              | 93  |
| 3.5. Co zaczynasz, od razu zakończ            | 94  |
| 3.6. Drzewo obiektów to nie bez powodu drzewo | 95  |
| 3.7. Zaznaczać czy nie zaznaczać?             | 96  |
| 3.8. Tablice                                  | 97  |
| 3.9. Jak NIE programować w VBA?               | 100 |

## ROZDZIAŁ 2.

# FRAGMENTY KODU VBA, KTÓRE W DZISIEJSZYM ŚWIECIE WYPADA ZNAĆ

To nie jest kolejna książka, w której zobaczysz fragmenty kodu oderwane od praktyki. Teorii najlepiej nauczyć się bowiem... analizując praktyczne biznesowe przykłady! Przygotowałem je dla Ciebie.

Ówczesny dziekan Wydziału Matematyki i Informatyki (bardzo mądry człowiek!), na którym studiowałem informatykę, mawiał: „Moją ulubioną metodą programowania jest kopiuj-wklej”. Już po chwili od zakończenia studiów zgodziłem się z Nim w pełni.

Miałem w głowie Jego myśl przez całe zawodowe życie. Bazowałem często na fragmentach kodu, których godzinami szukałem na forach dyskusyjnych. Sprawdzałem pierwszy odnaleziony fragment i... nie działał. Sprawdzałem kolejne i... także nie działały. Zanim doszedłem do działającego fragmentu kodu, mijały godziny. Zanim odpowiednio dołączyłem go do mojego kodu, mijały kolejne. Jednak ta radość z w pełni działającego kodu — bezcenna!

Oczywiście (jak się domyślasz) fragmenty, które można było zarejestrować, rejestrowałem i przekształcałem!

Gdy już miałem kod, to działałem tak, aby stał się uniwersalny. Na koniec w wielu przypadkach docierałem do wzorca matka-córki.

W tej książce umieszczam całą serię przydatnych biznesowo fragmentów do samodzielnej analizy.

Stosuję przy tym formę zagadek, bo dzięki ich rozwiązywaniu najszybciej zrozumiesz wszystkie meandry świata VBA. Pokazuję Ci wejście i wyjście danego zagadnienia. Zadaję pytania. Spróbuj sobie najpierw odpowiedzieć na nie w głowie, a dopiero później przeczytać, czy Twoja odpowiedź jest podobna do mojego toku myślenia.

Jeżeli tak będzie, to myślę, że powinniśmy polubić się na moim fan-page'u na Facebooku „Mateusz Boryga — Twój trener”.

Jeżeli nie, to myślę, że powinniśmy polubić się na moim fan-page'u na Facebooku „Mateusz Boryga — Twój trener”. A dodatkowo potrzebujesz sprawdzić, czy Twoja wersja też zadziała. Jest duża szansa, że tak! I to jest właśnie najpiękniejsze w wymyślaniu kodu! Jest wiele dróg prowadzących do sukcesu.

Opisuję też tło każdego fragmentu kodu, żeby można było wynieść jak najwięcej dodatkowej wartości z analizy. Pokazuję zatem, co rejestrowałem na początku, żeby otrzymać dużą część kodu. I jak to należało przekształcić, żeby było uniwersalnie. Prezentuję też to, czego szukałem w internecie i co było najtrudniejsze do przekształcenia. Wiem, że w ten sposób nauczysz się programowania w VBA najlepiej i najszybciej.

Aha, i najważniejsze! Każdy z tych przykładów działa. Nie ma za co!

## 2.1. ZAZNACZENIE WYMARZONEJ KOLUMNY

Jest taki fragment kodu, którego będziesz używać non stop. Ten fragment będzie pojawiał się za każdym razem, gdy zechcesz zaznaczyć odpowiednią kolumnę, aby później na niej dowolnie działać.

„Więcej szczegółów, Mateuszu”. Proszę bardzo...

Wyobraź sobie świat, w którym możesz zaznaczyć dowolną kolumnę po jej nazwie występującej w pierwszym wierszu, wierszu zwanym nagłówkiem. Taki świat, w którym nie ogranicza Cię pierwotne umiejscowienie kolumny. Też taki świat, w którym to umiejscowienie jest możliwe do zmiany, bo Twój automat potrafi ją wychwycić.

To piękny świat, prawda? Jak go stworzyć?



**Przetestuj!** Plik *Fragmenty kodu VBA.xlsm*, arkusz *Zaznaczanie kolumny*.

### Przed

|   | A         | B         | C                        | D         | E  |
|---|-----------|-----------|--------------------------|-----------|----|
| 1 | Kolumna A | Kolumna B | Nazwa wymarzonej kolumny | Kolumna D |    |
| 2 | 1         | 2         |                          | 3         | 4  |
| 3 | 5         | 6         |                          | 7         | 8  |
| 4 | 9         | 10        |                          |           | 12 |
| 5 | 13        | 14        |                          | 15        | 16 |
| 6 |           |           |                          |           |    |

### Po

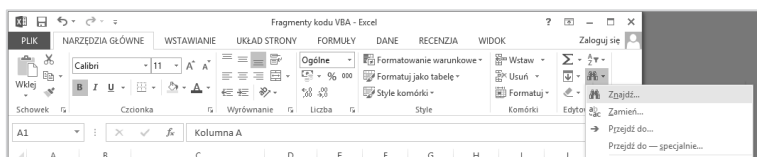
|   | A         | B         | C                        | D         | E |
|---|-----------|-----------|--------------------------|-----------|---|
| 1 | Kolumna A | Kolumna B | Nazwa wymarzonej kolumny | Kolumna D |   |
| 2 | 1         | 2         | 3                        | 4         |   |
| 3 | 5         | 6         | 7                        | 8         |   |
| 4 | 9         | 10        |                          | 12        |   |
| 5 | 13        | 14        | 15                       | 16        |   |
| 6 |           |           |                          |           |   |

## Ja zapytam!

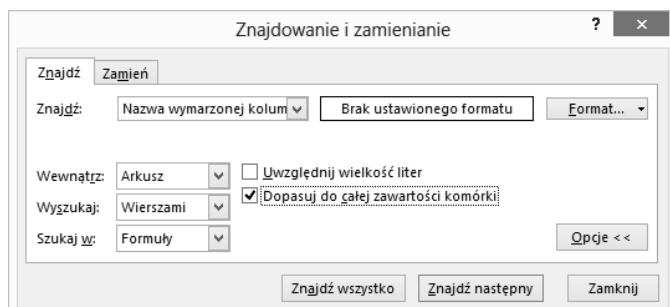
- Jakie kroki w Excelu przeprowadzą Cię ze świata „przed” do świata „po”? Spoiler: to będą późniejsze kroki do zarejestrowania!
- Których kroków nie da się zarejestrować i jaki masz na nie pomysł?

## Rozwiązanie

- Zaznaczenie pierwszego wiersza z nagłówkiem.
- *Narzędzia główne/Znajdź i zaznacz/Znajdź... (Ctrl+F, Find).*



- ◆ *Opcje >>/Dopasuj do całej zawartości komórki (zaznaczone)* — ta opcja została zaznaczona, aby w przypadku znalezienia kolumny *Nazwa wymarzonej kolumny 2* makro nie zatrzymało się przypadkowo na niej.



- Względne przejście jedną komórkę w dół.

|   | A         | B         | C                        | D         | E |
|---|-----------|-----------|--------------------------|-----------|---|
| 1 | Kolumna A | Kolumna B | Nazwa wymarzonej kolumny | Kolumna D |   |
| 2 | 1         | 2         | 3                        | 4         |   |
| 3 | 5         | 6         | 7                        | 8         |   |
| 4 | 9         | 10        |                          | 12        |   |
| 5 | 13        | 14        |                          | 16        |   |



- (Dopisane w VBA). Zapamiętanie pierwszej komórki, od której rozpoczniemy późniejsze zaznaczenie.
  - ◆ Zapamiętanie następuje w zmiennej obiektowej — zobaczysz, że to zmienna obiektowa po zwrocie Set, użytym w VBA.
  - ◆ Teoria:
    - Set NazwaZmiennejObiektowej = Obiekt
- (Dopisane w VBA). Zapamiętanie ostatniej w kolumnie wypełnionej komórki, na której zakończymy późniejsze zaznaczenie.
  - ◆ Zapamiętanie również następuje w zmiennej obiektowej — jak wyżej.
  - ◆ Zastosowany trik.
    - Przejście do ostatniej komórki w arkuszu w wybranej kolumnie.

|         | A | B | C | D | E |
|---------|---|---|---|---|---|
| 1048562 |   |   |   |   |   |
| 1048563 |   |   |   |   |   |
| 1048564 |   |   |   |   |   |
| 1048565 |   |   |   |   |   |
| 1048566 |   |   |   |   |   |
| 1048567 |   |   |   |   |   |
| 1048568 |   |   |   |   |   |
| 1048569 |   |   |   |   |   |
| 1048570 |   |   |   |   |   |
| 1048571 |   |   |   |   |   |
| 1048572 |   |   |   |   |   |
| 1048573 |   |   |   |   |   |
| 1048574 |   |   |   |   |   |
| 1048575 |   |   |   |   |   |
| 1048576 |   |   |   |   |   |

- Stamtąd przejście w górę (tę operację można zarejestrować za pomocą skrótu: *Ctrl+strzałka w górę*).

|   | A  | B  | C  | D  | E |
|---|----|----|----|----|---|
| 5 | 13 | 14 | 15 | 16 |   |
| 6 |    |    |    |    |   |
| 7 |    |    |    |    |   |

- (Dopisane w VBA). Zaznaczenie zakresu komórek kolumny o wybranej przez nas nazwie.

### Kod

```
Sub MatkaZaznaczenieKolumny()  
    Call ZaznaczenieKolumny("Nazwa wymarzonej kolumny")  
End Sub  
  
Sub ZaznaczenieKolumny(NazwaKolumny)  
    Rows("1:1").Select  
    Selection.Find(What:=NazwaKolumny, After:=ActiveCell,  
        ↪LookIn:=xlFormulas, LookAt:=xlWhole, SearchOrder:=xlByRows,  
        ↪SearchDirection:=xlNext, MatchCase:=False, SearchFormat:  
        ↪=False).Activate  
    ActiveCell.Offset(1, 0).Range("A1").Select  
    Set KomorkaStart = ActiveCell  
    Set KomorkaStop = Cells(Rows.Count, ActiveCell.Column).End(xlUp)  
    Range(KomorkaStart, KomorkaStop).Select  
End Sub
```

# PROGRAM PARTNERSKI

— GRUPY HELION —



1. ZAREJESTRUJ SIĘ
2. PREZENTUJ KSIĄŻKI
3. ZBIERAJ PROWIZJĘ

Zmień swoją stronę WWW w działający bankomat!

**Dowiedz się więcej i dołącz już dzisiaj!**

<http://program-partnerski.helion.pl>

GRUPA  
**Helion** 

# Nauka programowania dla nieprogramistów

Jeśli zastanawiasz się nad przekwalifikowaniem i karierą w branży informatycznej albo chcesz rozwinąć swoje umiejętności o programowanie, ale wydaje Ci się ono czarną magią, zapewniamy – w programowaniu nie ma nic z magii. To proces polegający na tworzeniu zbioru instrukcji, dzięki którym komputer wykona to, czego od niego oczekujesz. Jak się nauczyć pisać te instrukcje? Jeśli dysponujesz pakietem Microsoft Office i arkuszem kalkulacyjnym Excel, to masz też dostęp do języka programowania VBA (Visual Basic for Applications).

Autor tego poradnika pokaże Ci, że programowania najłatwiej nauczyć się właśnie w VBA. Pośród wielu użytecznych narzędzi, Microsoft Excel posiada również rejestrator makr. To on zamienia klikane w Excelu działania na pełnoprawny kod VBA. Dzięki właściwemu użyciu rejestratora możliwe jest więc proste generowanie kodu, na którym można dalej pracować.

- Wygeneruj kod za pomocą rejestratora makr Excela
- Zamień go na uniwersalny kod, przydatny w wielu projektach
- Zobacz najpopularniejsze gotowe do użycia fragmenty kodu VBA
- Poznaj dobre praktyki programowania w VBA
- Przejdź na wyższy poziom kodowania i doceń, jak wiele czasu dzięki temu oszczędzasz

**Mateusz Boryga** – analityk HR, trener IT i manager kadry szkoleniowej. Szczególnie upodobał sobie edukowanie w zakresie Excela – stworzył cykl szkoleniowy *Excel w minutę* i kurs wideo *Excel w jeden dzień. Uwolnij potęgę danych!* Jego warsztaty tworzenia magii w Excelu na Talent Days i Jobiconie zapętniły całe sale. Ma jedną misję: chce sprawić, by o każdym, kogo uczy, mówiono: „umie w Excela!”.

**Helion** 



helion.pl



HELION S.A.  
ul. Kościuszki 1c  
44-100 Gliwice  
tel.: 32 230 98 63  
helion@helion.pl

**KOD KORZYŚCI**  
Sięgnij po więcej! ▶



ISBN 978-83-289-1343-1



9 788328 913431

Cena: 39,90 zł