

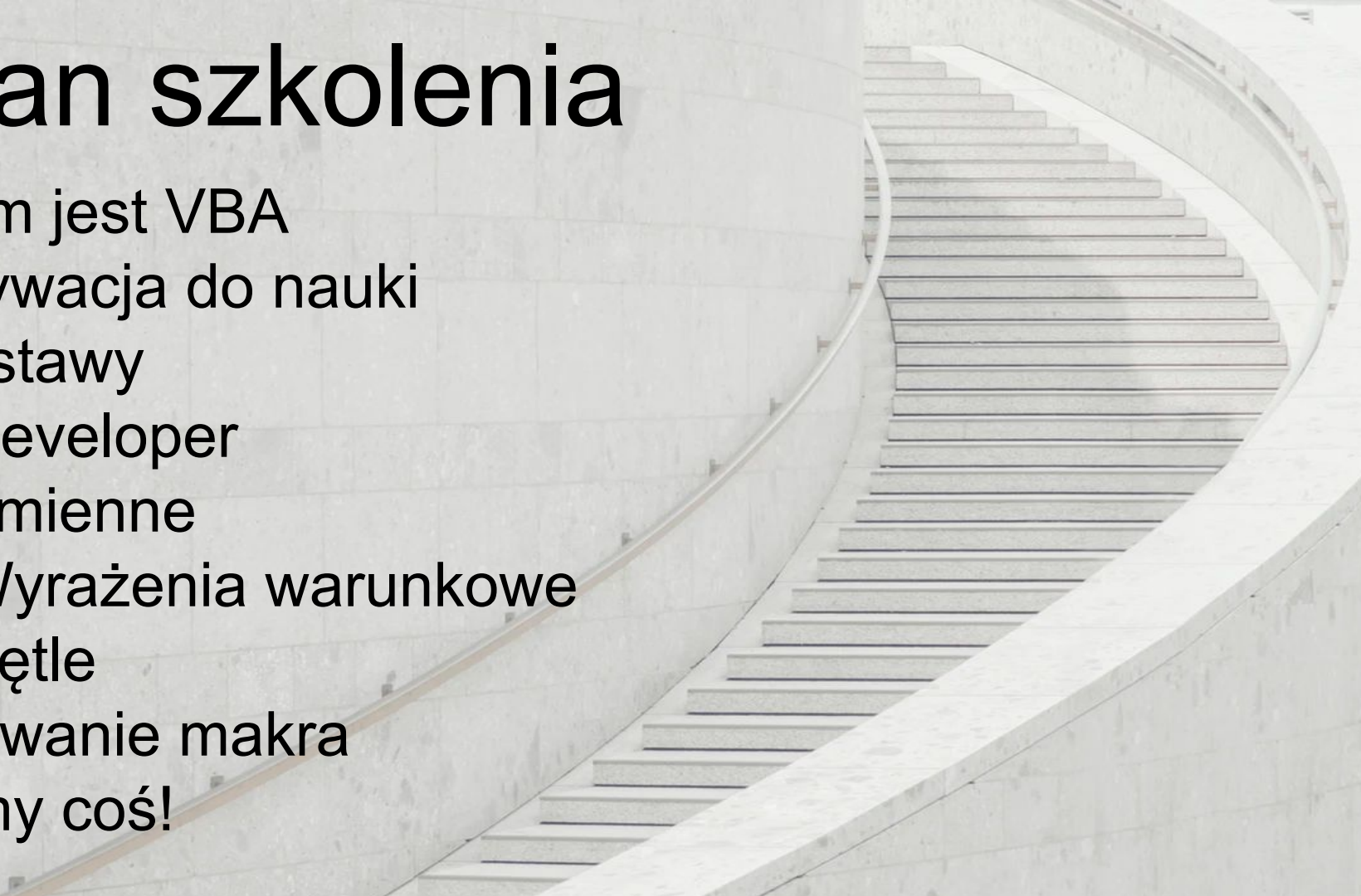
VBA od podstaw

A person wearing a white shirt is sitting at a desk, writing on a document with a pen. In the background, there is a laptop and a tablet. The image has a light blue tint.

mgr Sylwester Piątek
Joanna Waczyńska

Plan szkolenia

- Czym jest VBA
- Motywacja do nauki
- Podstawy
 - Developer
 - Zmienne
 - Wyrażenia warunkowe
 - Pętle
- Nagrywanie makra
- Zróbmy coś!



VBA

VB'

Służy przede wszystkim do
automatyzacji pracy z dokumentami

VBA

Visual Basic for Applications

Przede wszystkim w aplikacjach pakietu
MsOffice.

Ale też w AutoCAD czy WordPerfect

Skryptowy język programowania!



copy
past
VLOOKUP



VBA

Jesteś zwycięzcą!

- Oszczędzasz czas \Leftrightarrow większe zyski dla firmy.
 - Odczuwasz realny progres pracy.
 - Ułatwiasz pracę swoją i zespołu.

No i jest to fajny dodatek

 [Oferty pracy](#) [Pierwsza praca](#) [Profile pracodawców](#)

słowo kluczowe, stanowisko, firma

vba

×

miejscow

Filtruj:

Kategorie

Poziom stanowiska

Rodzaj umowy

▼

Praca: **vba** - Mamy dla Ciebie 379 ofert

Analitik ryzyka finansowego

Starszy Specjalista ds. Ryzyka Płynności

Systems Analyst

Specjalista ds. Analiz Podmiotów Finansowych

Specjalista ds. Analizy Danych

Specjalista ds. Zarządzania Portfelem

Specjalista ds. Analizy Danych Klienta

Financial Analyst/ Financial Controller

People Data Reporting Specialist

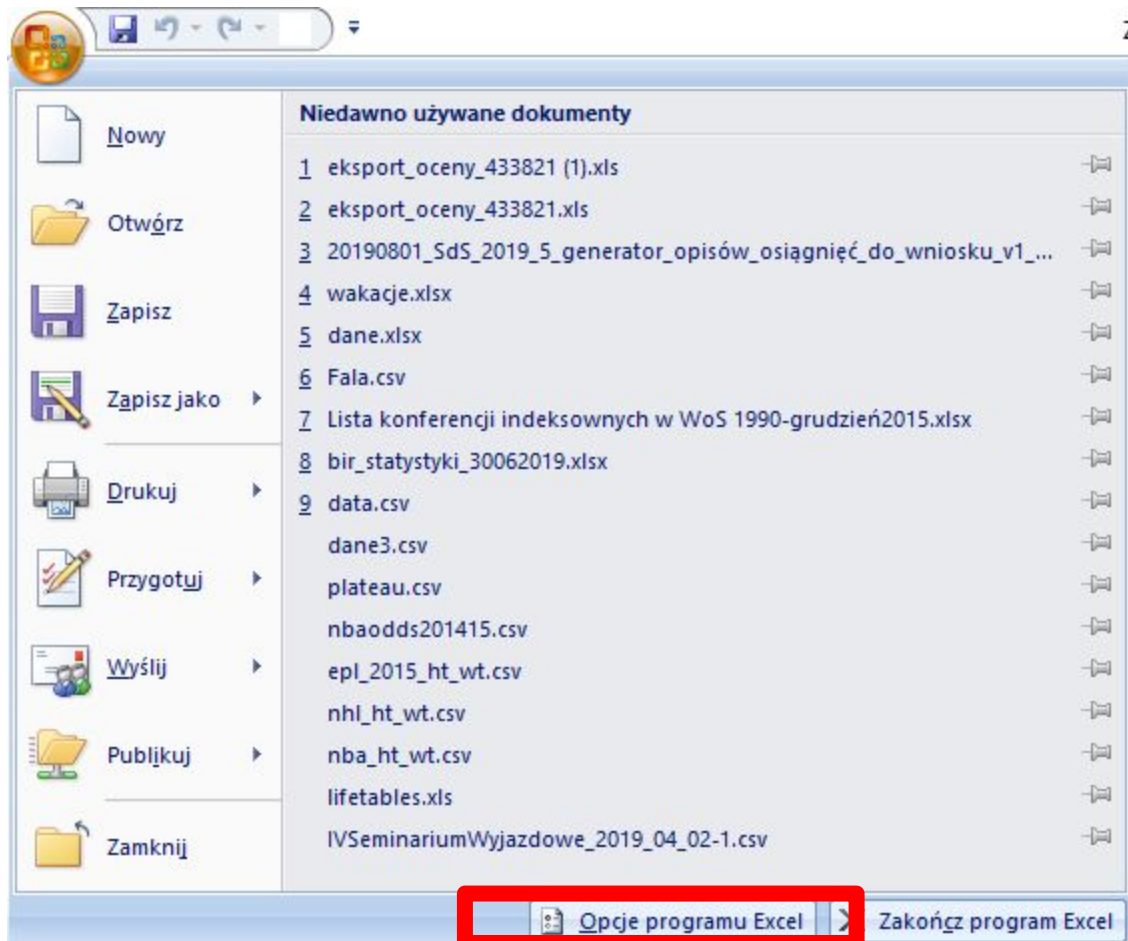
Risk Data Analyst

Data Analyst

Specjalista ds. Systemów Operacyjnych i Raportowania

Podstawy

DEVELOPER



Popularne

Formuły

Sprawdzanie

Zapisywanie

Zaawansowane

Dostosowywanie

Dodatki

Centrum zaufania

Zasoby



Zmianie najbardziej popularnych opcji programu Excel

Najczęściej używane opcje w pracy z programem Excel

☒ Pokaż minipasek narzędzi przy zaznaczaniu ⓘ☐ Wskaż podgląd danych ⓘ☒ Pokaż kartę Deweloper na Wstążce ⓘ

Schemat kolorów: Niebieski ▾

Styl etykietek ekranowych: Pokaż opisy funkcji w etykietkach ekranowych ▾

Utwórz listy do użycia podczas sortowania i w sekwencjach wypełniania: [Edytuj listy niestandardowe...](#)

Podczas tworzenia nowych skoroszytów

Użyj tej czcionki: Czcionka tekstu podstawowego ▾

Rozmiar czcionki: 11 ▾

Widok domyślny dla nowych arkuszy: Widok normalny ▾

Dołącz następującą liczbę arkuszy: 3 ▴ ▾

Personalizowanie kopii pakietu Microsoft Office

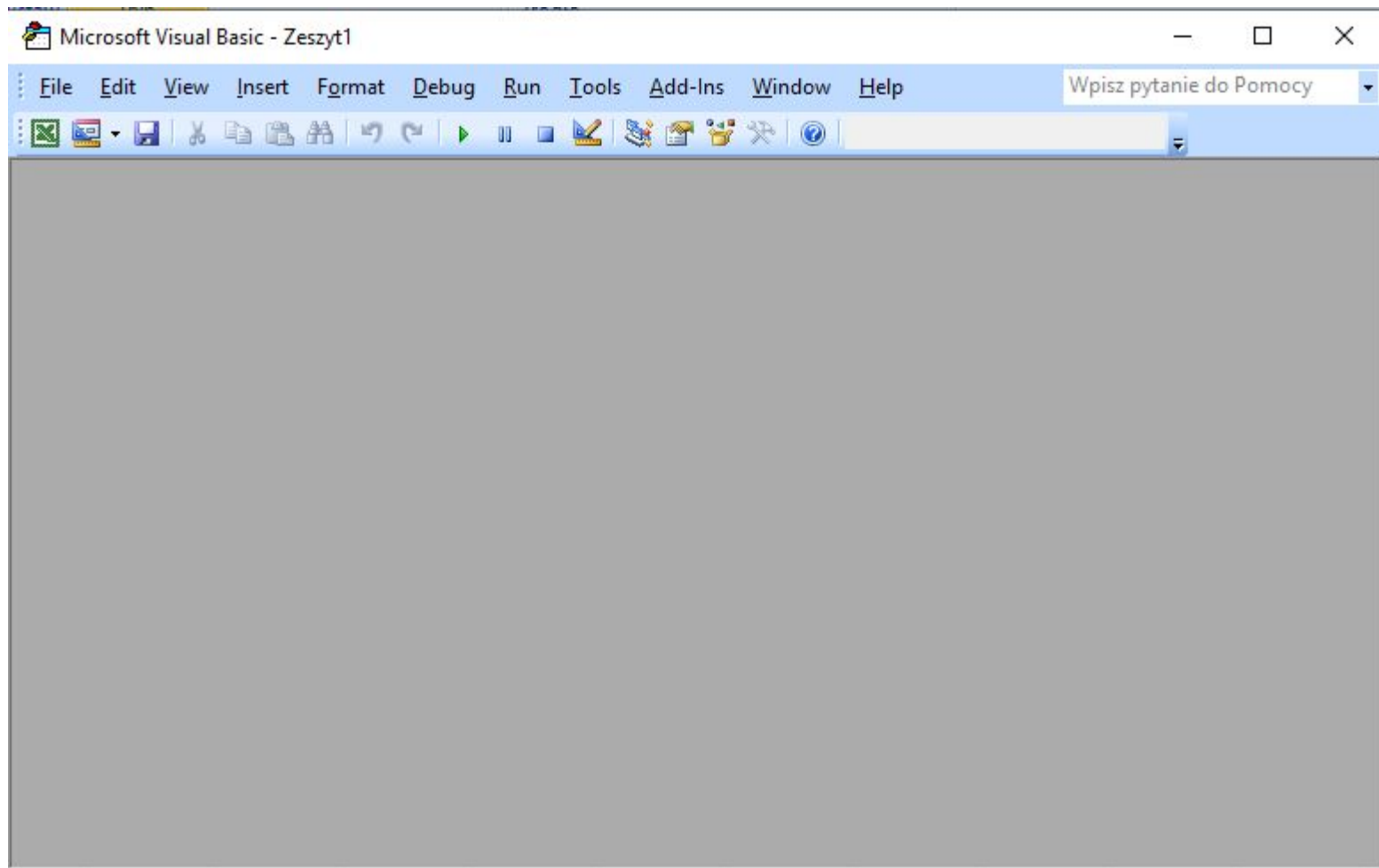
Nazwa użytkownika: Asia

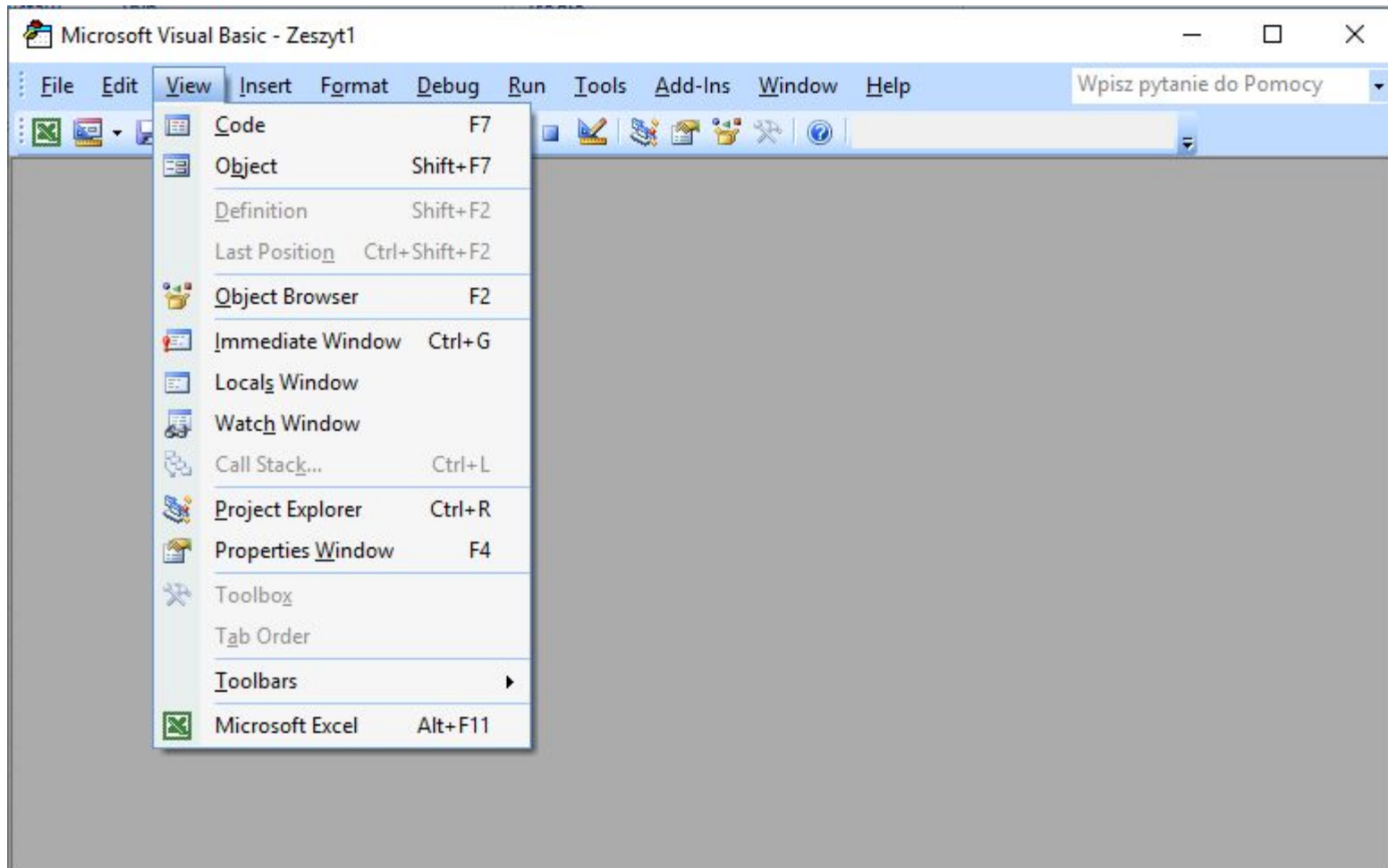
Wybierz języki, których chcesz używać w pakiecie Microsoft Office: [Ustawienia języka...](#)

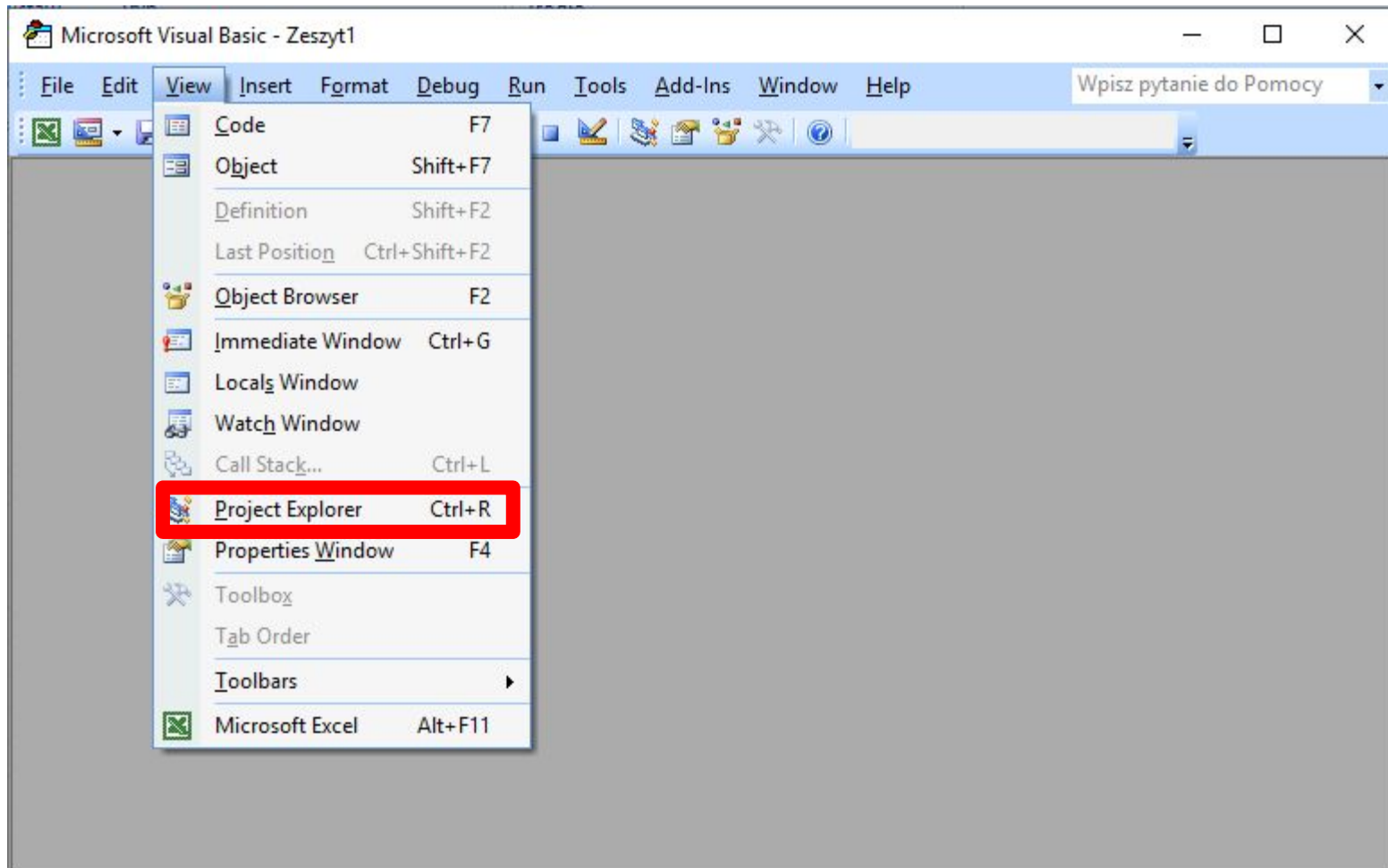
OK

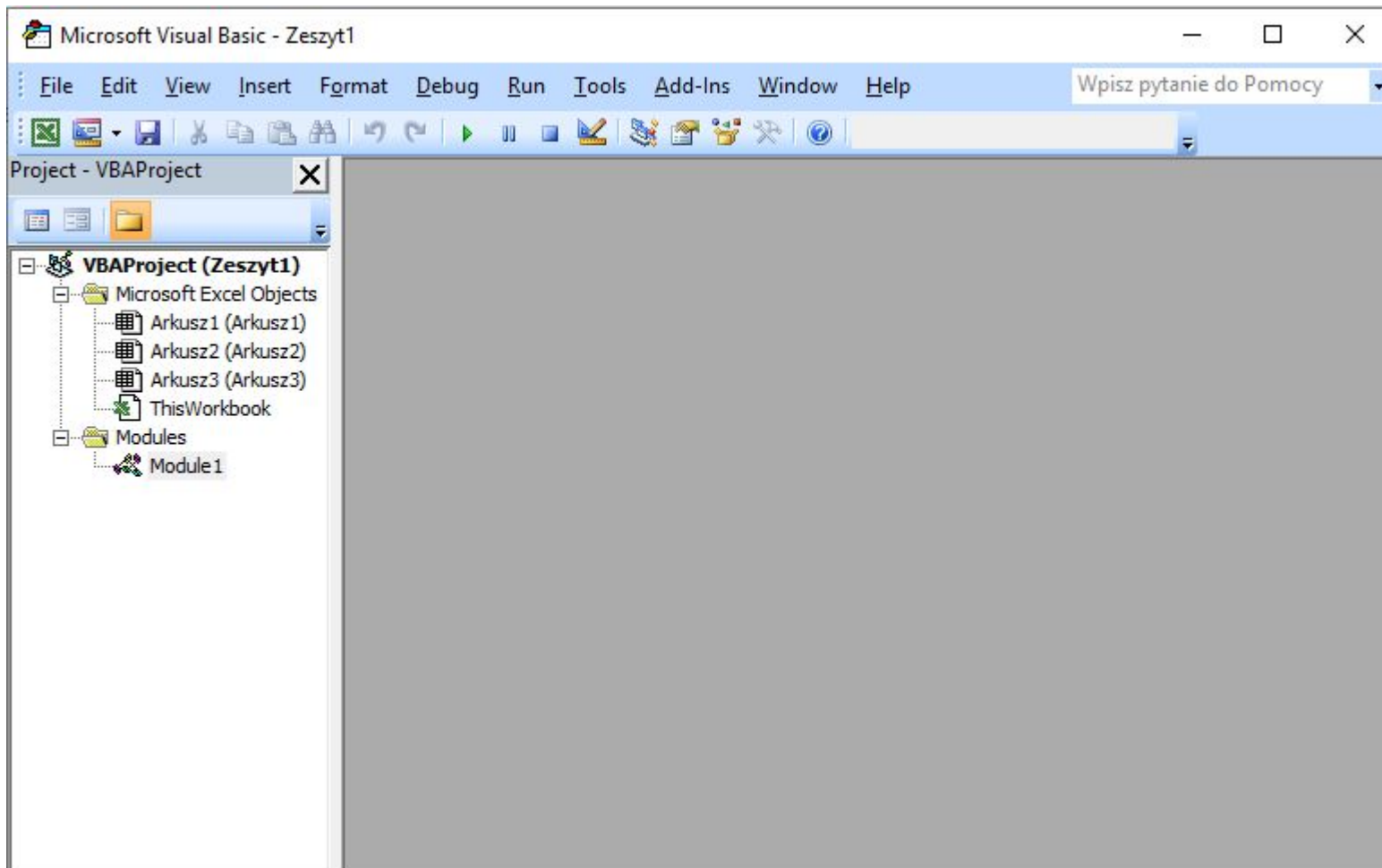
Anuluj

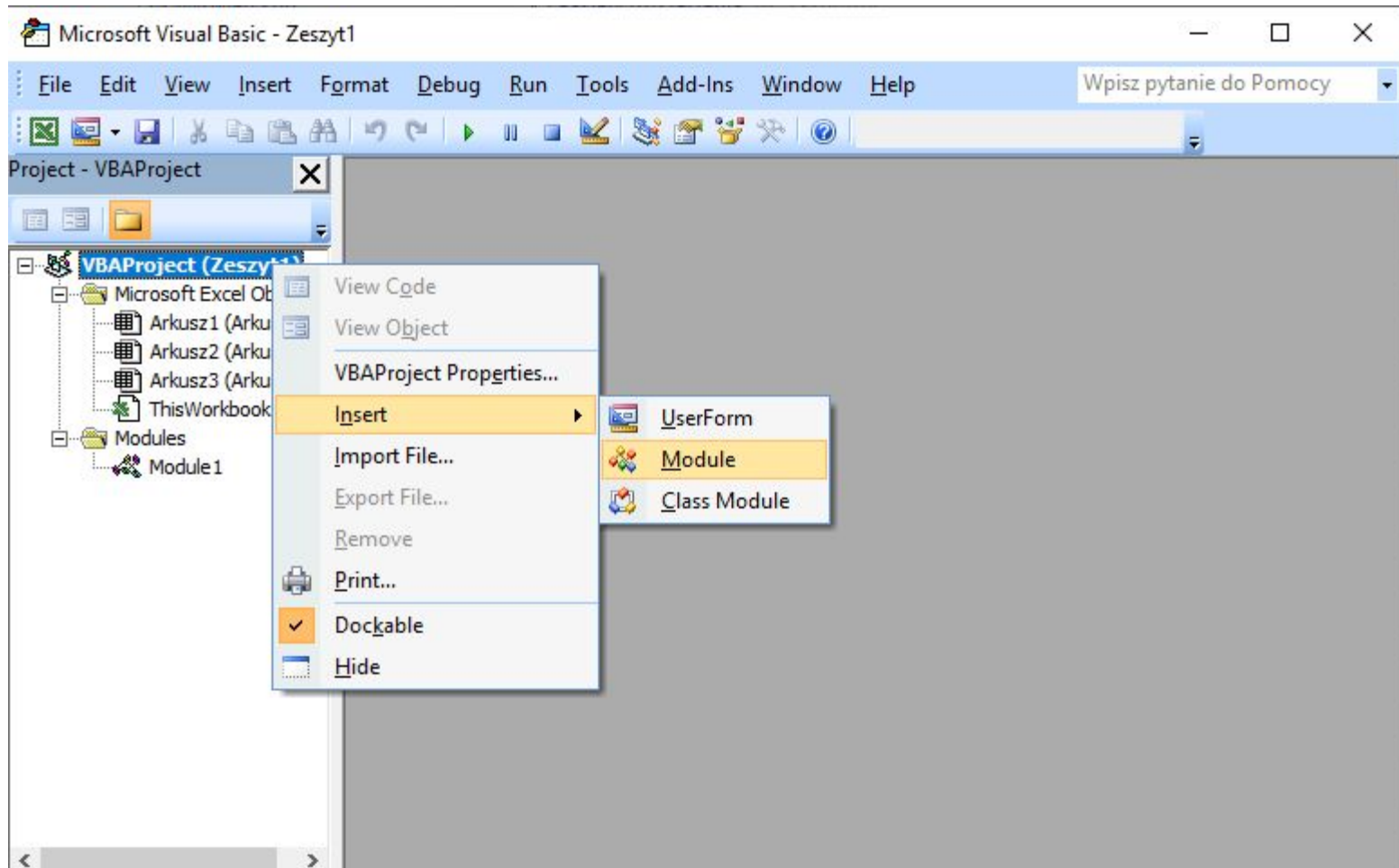
Naciskamy Alt + F11

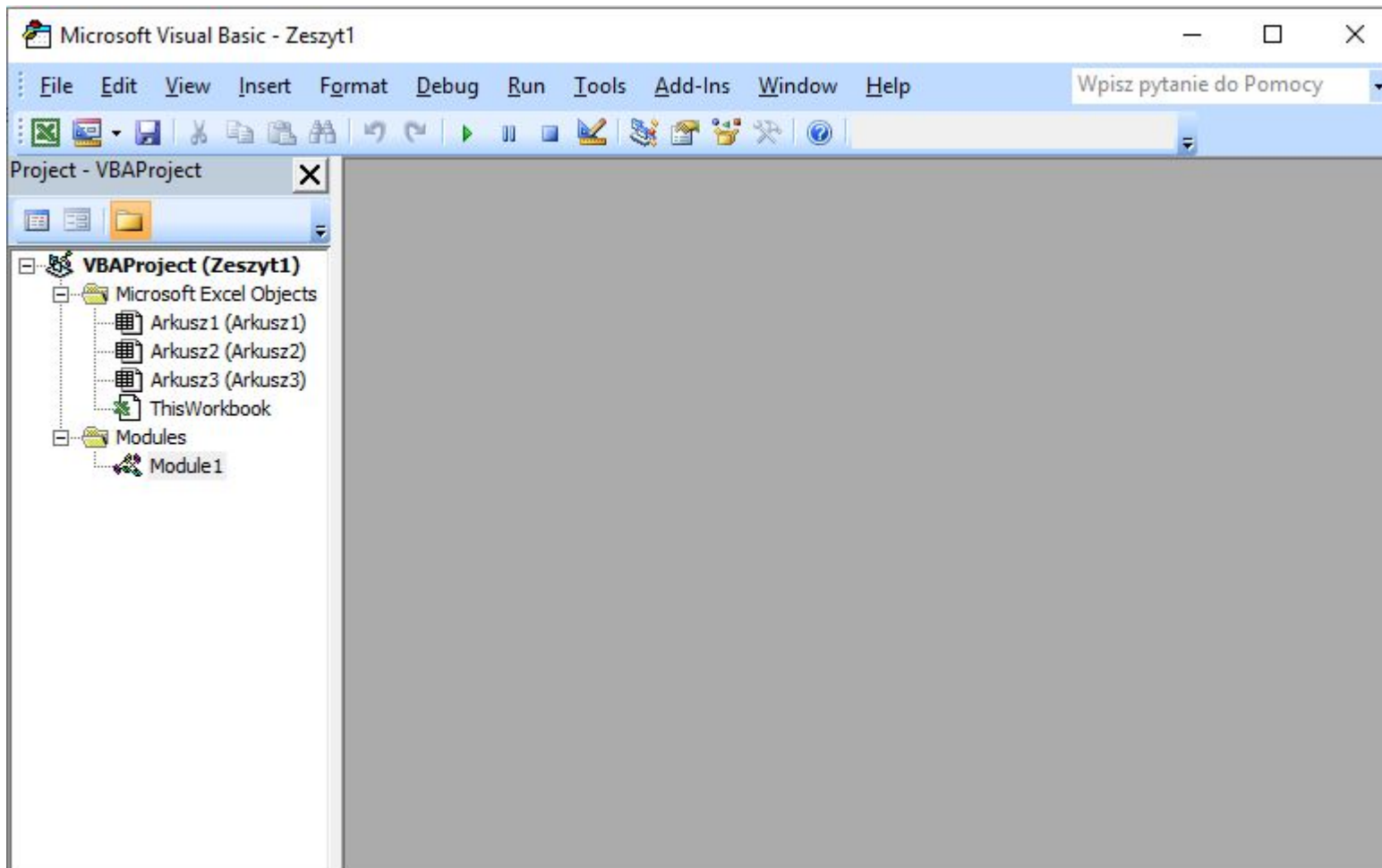


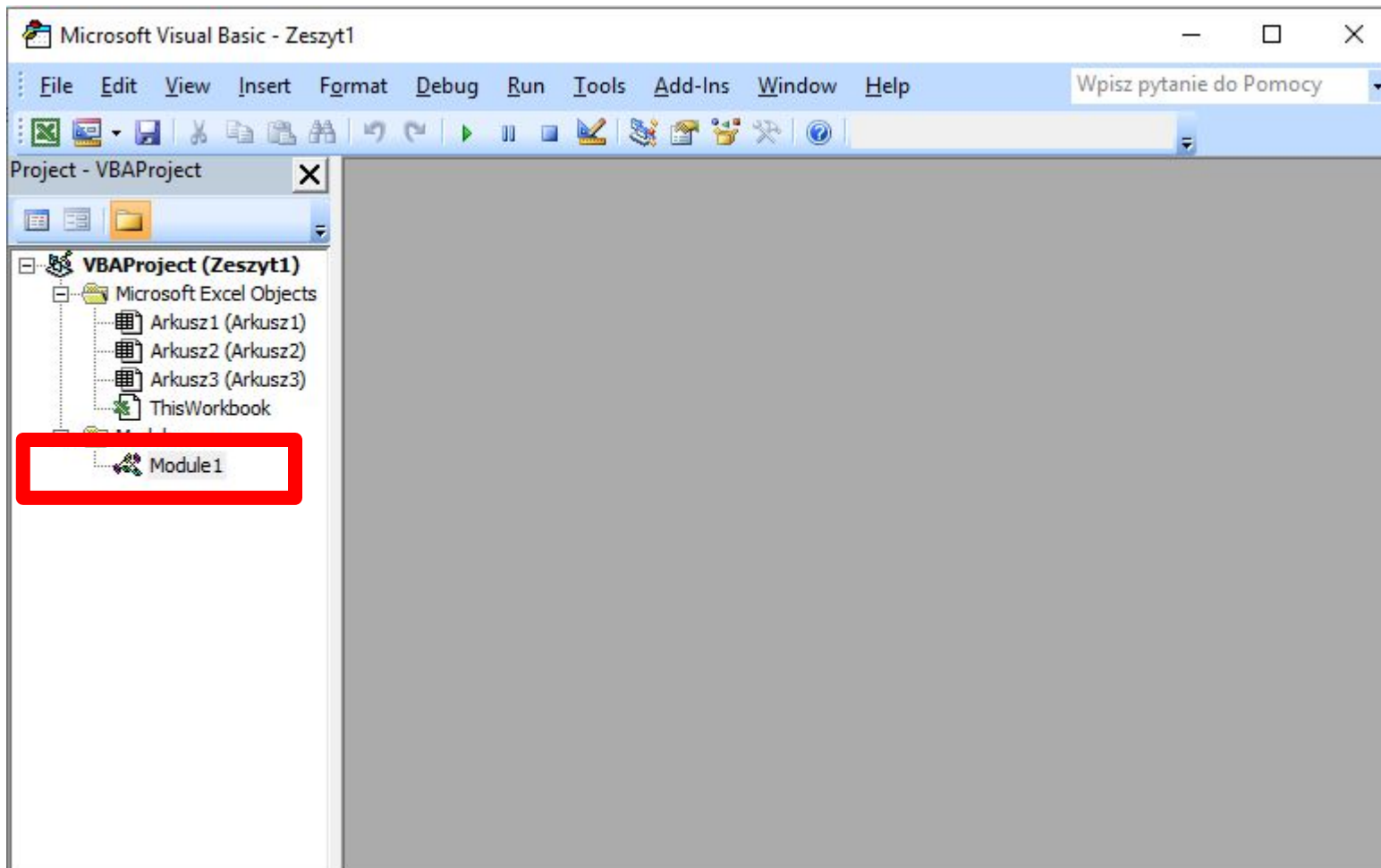


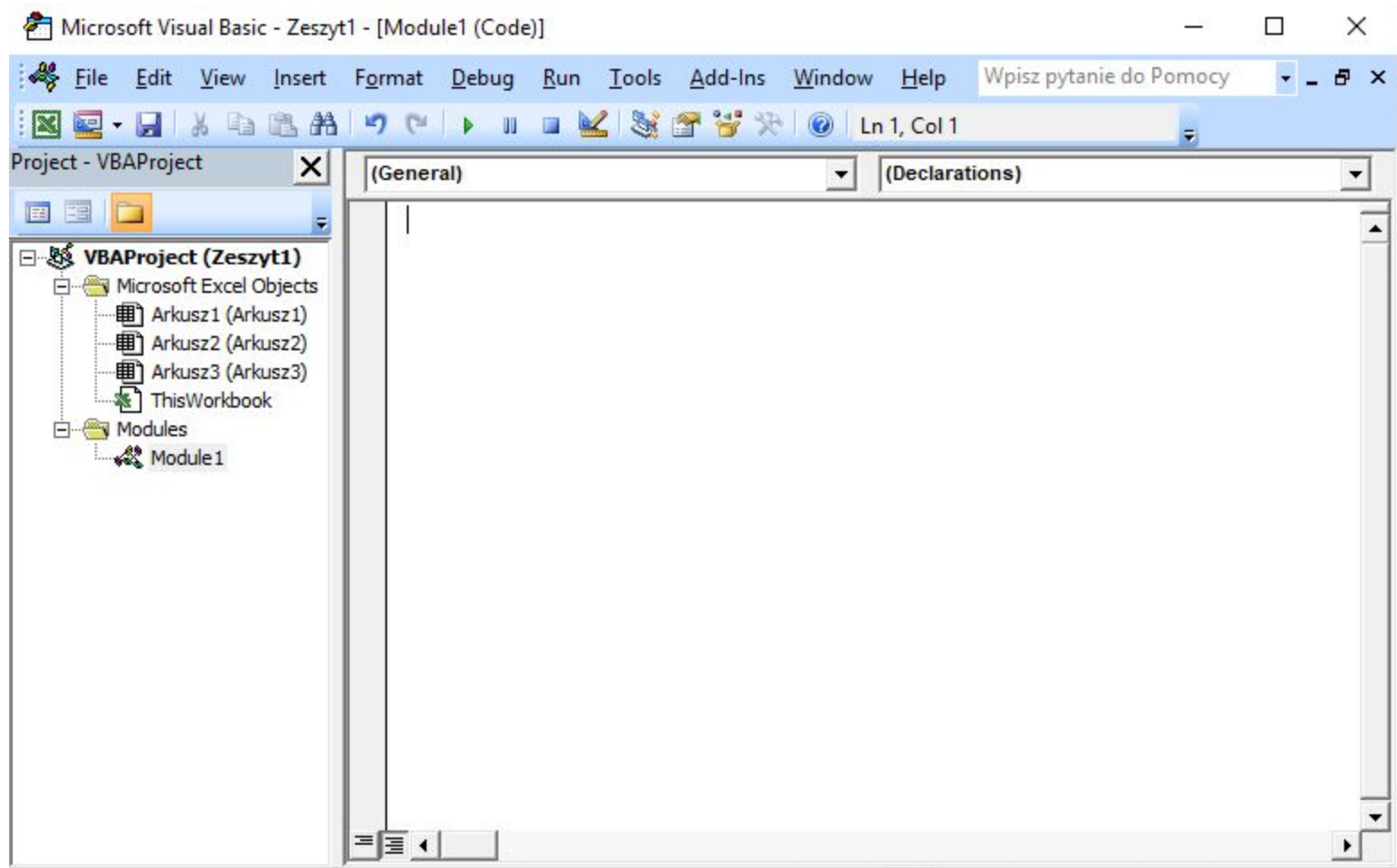




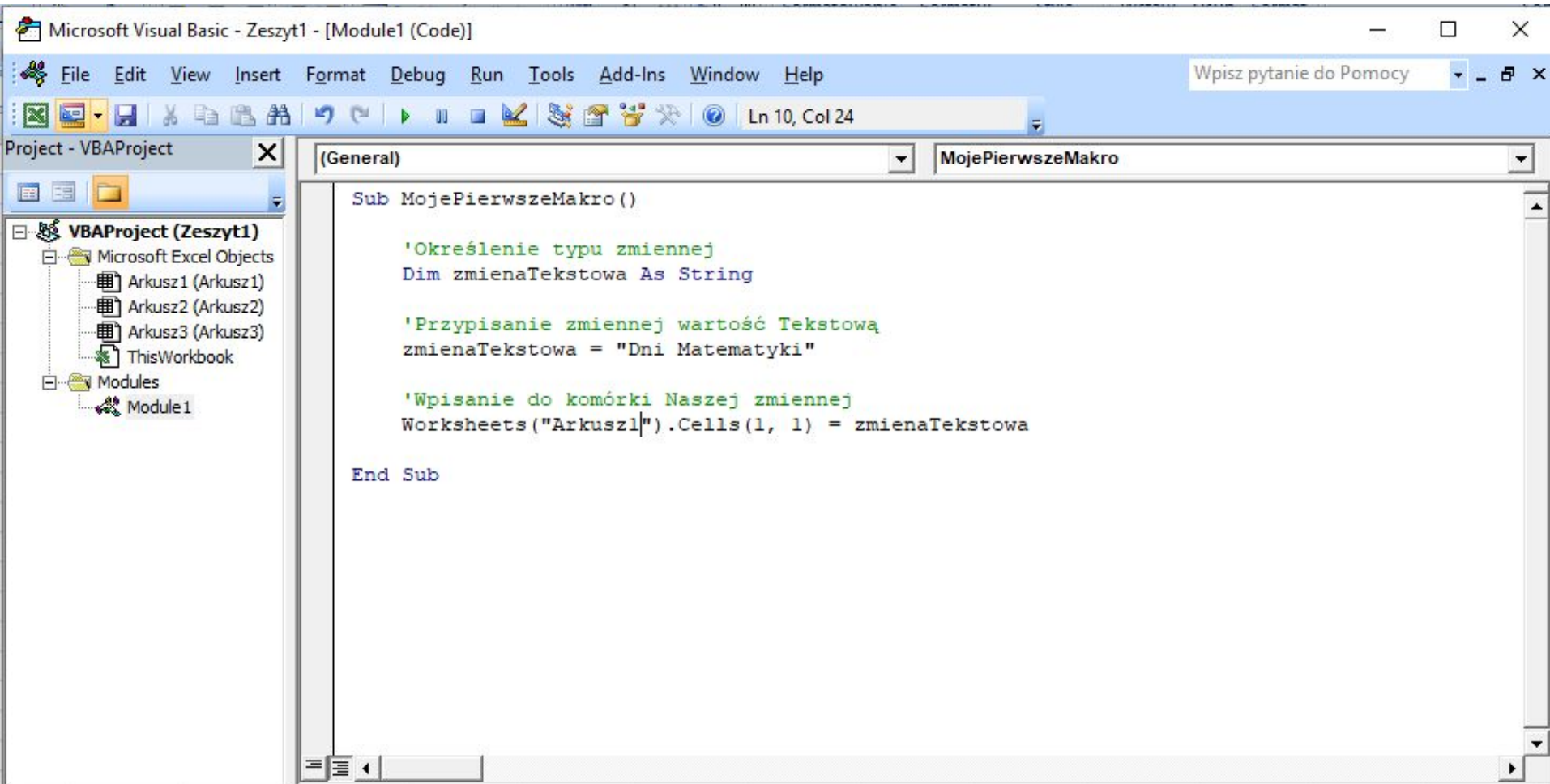


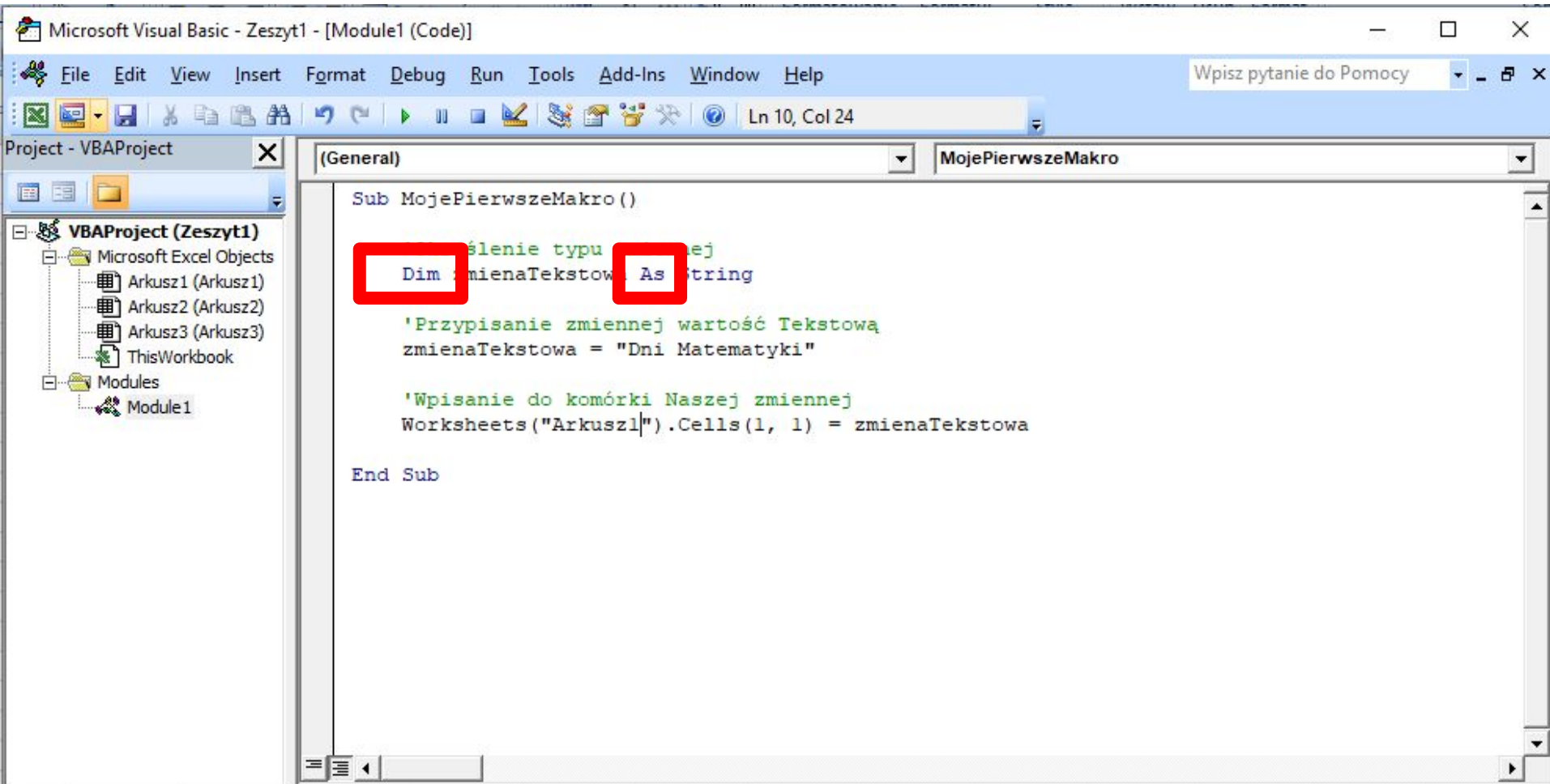


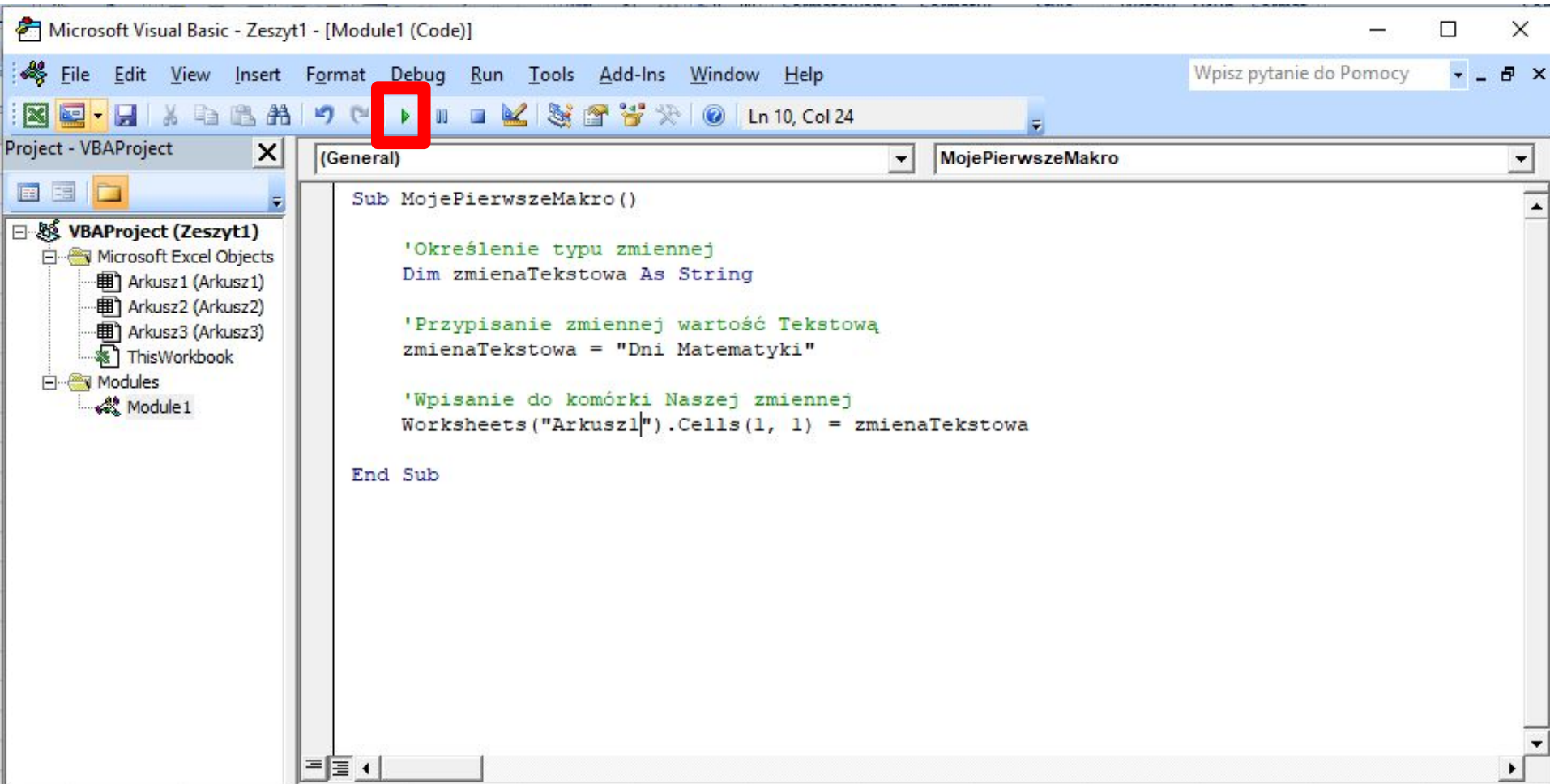


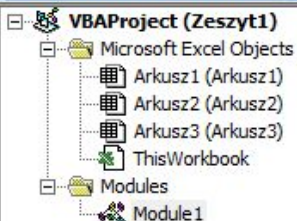


Typy zmiennych







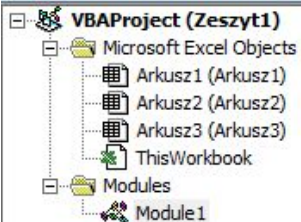


```
Sub MojeDrugieMakro()  
    Dim zmiennaTypuByte As Byte 'liczby naturalne od 0 do 255  
    Dim zmiennaTypuInteger As Integer 'liczby całkowite z zakresu od -32768 do 32767  
    Dim zmiennaTypuLong As Long 'liczby całkowite z zakresu od -2147483648 do 2147483647  
  
    'liczby rzeczywiste i obejmuje wartości z zakresu od -3.4·1038 do -1.4·10-45 dla wartości ujemnych  
    'oraz od 1.4·10-45 do 3.4·1038 dla wartości dodatnich  
    Dim zmiennaTypuSingle As Single  
  
    'liczby rzeczywiste i obejmuje wartości z zakresu od -1.80·10308 do -4.94·10-324 dla wartości  
    'ujemnych oraz od 4.94·10-324 do 1.80·10308 dla wartości dodatnich  
    Dim zmiennaTypuDouble As Double  
  
    'liczby z zakresu od -922337203685477.5808 do 922337203685477.5807  
    Dim zmiennaTypuCurrency As Currency  
  
    zmiennaTypuByte = 2  
    Worksheets("Arkusz1").Cells(1, 1) = zmiennaTypuByte  
    zmiennaTypuByte = "2" 'kompilator potrafi przełożyć sobie tekst "2" na liczbę  
    Worksheets("Arkusz1").Cells(2, 1) = zmiennaTypuByte  
End Sub
```

VBAProject (Zeszyt1)

- Microsoft Excel Objects
 - Arkusz1 (Arkusz1)
 - Arkusz2 (Arkusz2)
 - Arkusz3 (Arkusz3)
 - ThisWorkbook
- Modules
 - Module1

```
Sub MojeTrzecieMakro()  
    Dim zmiennaTypuDate As Date 'Służy do przechowywania informacji o dacie i/lub czasie.  
    zmiennaTypuDate = #10/25/2019# 'Wybrana data  
    Worksheets("Arkusz1").Cells(1, 1) = zmiennaTypuDate  
    zmiennaTypuDate = Now() 'Dzisiejsza data i godzina  
    Worksheets("Arkusz1").Cells(2, 1) = zmiennaTypuDate  
End Sub
```



```
Sub MojeCzwarteMakro()  
    'Zmienne typu Boolean mogą przyjmować dwie wartości: True oraz False.  
    Dim zmiennaTypuBoolean As Boolean  
    Dim zmiennaTypuVariant As Variant 'Deklaracja zmiennej typu ogólnego  
End Sub
```

(General)

MojePiateMakro

Option Explicit wymusza jawne zadeklarowanie wszystkich zmiennych przy użyciu instrukcji Dim lub ReDim

```
Sub MojePiateMakro()
```

```
    'Zmienne typu Workbook i Worksheet są używane do przechowywania skoroszytów (plików) oraz arkuszy jako zmienne.
```

```
    Dim MainWb As Workbook
```

```
    Dim SummaryWs As Worksheet
```

```
    Set MainWb = ThisWorkbook
```

```
    SummaryWs = MainWb.Worksheets("Summary")
```

```
    'SummaryWs = MainWb.Worksheets(1)    'niezalecane, bo kolejność arkuszy w pliku może się zmieniać
```

```
End Sub
```

Wyrażenia warunkowe

VBAProject (Zeszyt1)

- Microsoft Excel Objects
 - Arkusz1 (Arkusz1)
 - Arkusz2 (Arkusz2)
 - Arkusz3 (Arkusz3)
 - ThisWorkbook
- Modules
 - Module1

```
Sub WyrażeniaWarunkowe()  
    'Jeżeli liczba znajdująca się w 1 kolumnie, w 1 wierszu jest większa od 2  
    If Cells(1, 1).Value > 2 Then  
        'Wpisz komentarz w 1 kolumnie, w 2 wierszu  
        Cells(1, 2) = "Podana liczba jest większa od 2"  
    End If  
End Sub
```



```
Sub WyrażeniaWarunkowe()
```

```
    If Cells(1, 1).Value > 2 Then
```

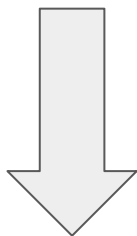
```
        Wpisz komentarz w 1 kolumnie, w 2 wierszu
```

```
        Cells(1, 2) = "Podana liczba jest większa od 2"
```

```
    End If
```

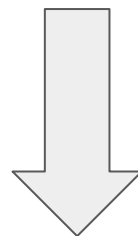
```
End Sub
```


	A	B	C	D	E
1	3				
2					
3					
4					
5					



	A	B	C	D	E
1	3 Podana liczba jest większa od 2				
2					
3					
4					
5					

	A	B	C	D	E
1	1				
2					
3					
4					
5					



	A	B	C	D	E
1	1				
2					
3					
4					
5					

```
Sub WyrażeniaWarunkowe()  
    'Jeżeli liczba znajdująca się w 1 kolumnie, w 1 wierszu jest większa od 2  
    If Cells(1, 1).Value > 2 Then  
        'Wpisz komentarz w 1 kolumnie, w 2 wierszu  
        Cells(1, 2) = "Podana liczba jest większa od 2"  
    Else 'W przeciwnym razie  
        'Wpisz komentarz w 1 kolumnie, w 2 wierszu  
        Cells(1, 2) = "Podana liczba jest mniejsza od 2"  
    End If  
End Sub
```

```
Sub WyrażeniaWarunkowe()
```

```
    'Jeżeli liczba znajdująca się w 1 kolumnie, w 1 wierszu jest większa od 2
```

```
    If Cells(1, 1).Value > 2 Then
```

```
        'Wpisz komentarz w 1 kolumnie, w 2 wierszu
```

```
        Cells(1, 2) = "Podana liczba jest większa od 2"
```

```
    Else 'przeciwnym razie
```

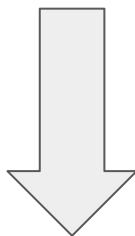
```
        'Wpisz komentarz w 1 kolumnie, w 2 wierszu
```

```
        Cells(1, 2) = "Podana liczba jest mniejsza od 2"
```

```
    End If
```

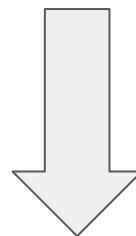
```
End Sub
```

	A	B	C	D	E
1	3				
2					
3					
4					
5					



	A	B	C	D	E
1	3 Podana liczba jest większa od 2				
2					
3					
4					
5					

	A	B	C	D	E
1	1				
2					
3					
4					
5					



	A	B	C	D	E
1	1 Podana liczba jest mniejsza od 2				
2					
3					
4					
5					

Petle

Microsoft Visual Basic - Zeszyt1 - [Module1 (Code)]

File Edit View Insert Format Debug Run Tools Add-Ins Window Help Wpisz pytanie do Pomocy

Ln 53, Col 1

Project - VBAProject

VBAProject (Zeszyt1)

- Microsoft Excel Objects
 - Arkusz1 (Arkusz1)
 - Arkusz2 (Arkusz2)
 - Arkusz3 (Arkusz3)
 - ThisWorkbook
- Modules
 - Module1

(General) ProstaPetlaFor

```
Sub ProstaPetlaFor()  
    Dim Wiersz As Integer  
  
    For Wiersz = 1 To 10 'Zaczynij od Wiersz = 1, skoncz gdy Wiersz = 10  
        Cells(Wiersz, 1) = "Jest to " & Wiersz & " wiersz"  
    Next Wiersz 'Weź następny Wiersz  
End Sub
```

→

	A
1	Jest to 1 wiersz
2	Jest to 2 wiersz
3	Jest to 3 wiersz
4	Jest to 4 wiersz
5	Jest to 5 wiersz
6	Jest to 6 wiersz
7	Jest to 7 wiersz
8	Jest to 8 wiersz
9	Jest to 9 wiersz
10	Jest to 10 wiersz

Pętla "For...Next"

Microsoft Visual Basic - Zeszyt1 - [Module1 (Code)]

File Edit View Insert Format Debug Run Tools Add-Ins Window Help Wpisz pytanie do Pomocy

Ln 53, Col 1

Project - VBAProject

(General) ProstaPetlaFor

```
Sub ProstaPetlaFor()  
    Dim Wiersz As Integer  
    For Wiersz = 1 To 10 'Zaczynij od Wiersz = 1, skończ gdy Wiersz = 10  
        Cells(Wiersz, 1) = "Jest to " & Wiersz & " wiersz"  
    Next Wiersz 'Weź następny Wiersz  
End Sub
```

VBAProject (Zeszyt1)

- Microsoft Excel Objects
 - Arkusz1 (Arkusz1)
 - Arkusz2 (Arkusz2)
 - Arkusz3 (Arkusz3)
 - ThisWorkbook
- Modules
 - Module1

→

	A
1	Jest to 1 wiersz
2	Jest to 2 wiersz
3	Jest to 3 wiersz
4	Jest to 4 wiersz
5	Jest to 5 wiersz
6	Jest to 6 wiersz
7	Jest to 7 wiersz
8	Jest to 8 wiersz
9	Jest to 9 wiersz
10	Jest to 10 wiersz

Pętla "For...Next"

Microsoft Visual Basic - Zeszyt1 - [Module1 (Code)]

File Edit View Insert Format Debug Run Tools Add-Ins Window Help Wpisz pytanie do Pomocy

Ln 67, Col 1

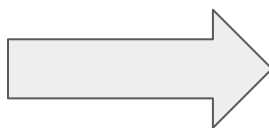
Project - VBAProject

VBAProject (Zeszyt1)

- Microsoft Excel Objects
 - Arkusz1 (Arkusz1)
 - Arkusz2 (Arkusz2)
 - Arkusz3 (Arkusz3)
 - ThisWorkbook
- Modules
 - Module1

(General) PetlaDoLoopUntil

```
Sub PetlaDoLoopUntil()  
    Dim i As Byte 'Nasza zmienna pomocnicza,  
    'jeśli jej do niczego nie przypiszemy, i:= 0  
    Do  
        i = i + 1  
        Cells(i, 2) = i & " krok." 'wpisz do komórki komentarz  
    Loop Until i = 10 'Powtórz do momentu, kiedy i = 10  
End Sub
```



	A	B
1		1 krok.
2		2 krok.
3		3 krok.
4		4 krok.
5		5 krok.
6		6 krok.
7		7 krok.
8		8 krok.
9		9 krok.
10		10 krok.

Microsoft Visual Basic - Zeszyt1 - [Module1 (Code)]

File Edit View Insert Format Debug Run Tools Add-Ins Window Help Wpisz pytanie do Pomocy

Ln 67, Col 1


Project - VBAProject

VBAProject (Zeszyt1)

- Microsoft Excel Objects
 - Arkusz1 (Arkusz1)
 - Arkusz2 (Arkusz2)
 - Arkusz3 (Arkusz3)
- ThisWorkbook
- Modules
 - Module1

(General) PetlaDoLoopUntil

```
Sub PetlaDoLoopUntil()  
    Dim i As Byte 'Nasza zmienna pomocnicza,  
    jej do niczego nie przypiszemy, i:= 0  
    Do  
        i = i + 1  
        Cells(i, 2) = i & " krok." 'wpisz do komórki komentarz  
    Loop Until i = 10 'Powtórz do momentu, kiedy i = 10  
End Sub
```



	A	B
1		1 krok.
2		2 krok.
3		3 krok.
4		4 krok.
5		5 krok.
6		6 krok.
7		7 krok.
8		8 krok.
9		9 krok.
10		10 krok.

Microsoft Visual Basic - Zeszyt1 - [Module1 (Code)]

File Edit View Insert Format Debug Run Tools Add-Ins Window Help Wpisz pytanie do Pomocy

Ln 79, Col 1

Project - VBAProject

(General) PetlaDoWhileLoop

```
Sub PetlaDoWhileLoop()  
    Dim i As Byte 'Zmienna Pomocnicza  
    Do While i < 10 'Powtarzaj czynności dopóki i < 10  
        i = i + 1 'zwiększ wartość zmiennej pomocniczej  
        Cells(i, 3) = i 'Wpisz liczbę do 3 kolumny  
    Loop  
End Sub
```

VBAProject (Zeszyt1)

- Microsoft Excel Objects
 - Arkusz1 (Arkusz1)
 - Arkusz2 (Arkusz2)
 - Arkusz3 (Arkusz3)
 - ThisWorkbook
- Modules
 - Module1



	A	B	C
1			1
2			2
3			3
4			4
5			5
6			6
7			7
8			8
9			9
10			10

Pętla "Do While...Loop"

Microsoft Visual Basic - Zeszyt1 - [Module1 (Code)]

File Edit View Insert Format Debug Run Tools Add-Ins Window Help Wpisz pytanie do Pomocy

Ln 79, Col 1

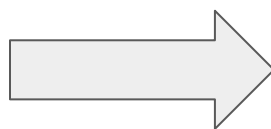
Project - VBAProject

(General) PetlaDoWhileLoop

```
Sub PetlaDoWhileLoop()  
    Dim i As Byte 'Zmienna Pomocnicza  
    Do While i < 10 'Powtarzaj czynności dopóki i < 10  
        i = i + 1 'zwiększ wartość zmiennej pomocniczej  
        Cells(i, 3) = i 'Wpisz liczbę do 3 kolumny  
    Loop  
End Sub
```

VBAProject (Zeszyt1)

- Microsoft Excel Objects
 - Arkusz1 (Arkusz1)
 - Arkusz2 (Arkusz2)
 - Arkusz3 (Arkusz3)
 - ThisWorkbook
- Modules
 - Module1



	A	B	C
1			1
2			2
3			3
4			4
5			5
6			6
7			7
8			8
9			9
10			10

Pętla "Do While...Loop"

Nagrywanie makr

Rejestrowanie makra

Nazwa makra:

Makro4

Przechowaj makro w:

Ten skoroszyt

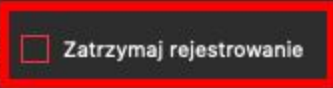
Klawisz skrót:

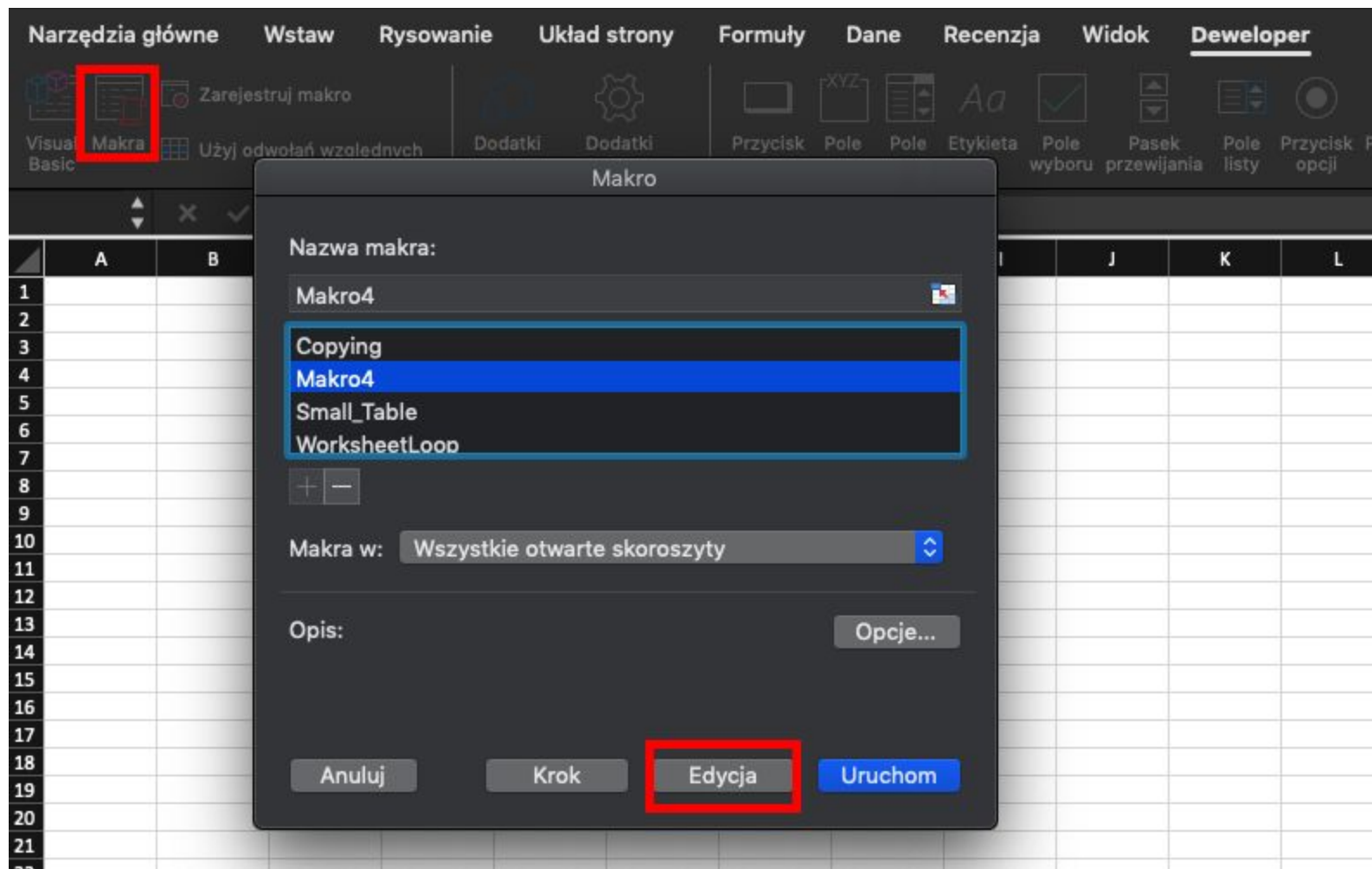
Opcja+Cmd+

Opis:

Anuluj

OK





Przykłady

(General)

WorksheetLoop

```
Sub WorksheetLoop()  
    Dim WS_Count As Integer  
    Dim Iter As Integer  
    Dim WS_Summary As Worksheet  
  
    ' Set WS_Count equal to the number of worksheets in the active workbook.  
    WS_Count = ActiveWorkbook.Worksheets.Count  
    Set WS_Summary = ActiveWorkbook.Worksheets("Summary")  
  
    ' Begin the loop.  
    For Iter = 1 To WS_Count  
        If ActiveWorkbook.Worksheets(Iter).Name <> "Summary" Then  
            WS_Summary.Range("B" & Iter) = ActiveWorkbook.Worksheets(Iter).Name  
            WS_Summary.Range("C" & Iter) = Left(ActiveWorkbook.Worksheets(Iter).Name, 4)  
            WS_Summary.Range("D" & Iter) = Right(ActiveWorkbook.Worksheets(Iter).Name, 3)  
        End If  
    Next Iter  
  
End Sub
```

(General)

Small_Table

```
Sub Small_Table()  
    Dim SheetCount As Integer  
    ActiveWorkbook.Worksheets("Summary").Select  
    Range("B2").Select  
    SheetCount = Cells(Rows.Count, 1).End(xlUp).Row  
  
    RowNumber = 2  
    While ActiveWorkbook.Worksheets("Summary").Range("B" & RowNumber) <> ""  
        SheetName = ActiveWorkbook.Worksheets("Summary").Range("B" & RowNumber)  
        ActiveWorkbook.Worksheets(SheetName).Select  
        Cells(2, 5).Value = "Rok"  
        Cells(3, 5).Value = "Miesiąc"  
        Cells(4, 5).Value = "Liczba transakcji"  
        Cells(5, 5).Value = "Suma przychodu"  
        Cells(6, 5).Value = "Średnia kwota transakcji"  
  
        Dim LastRow As Long  
        'Find the last non-blank cell in column A(1)  
        Range("A2").Select  
        LastRow = Cells(Rows.Count, 1).End(xlUp).Row  
  
        Range("F2") = Left(SheetName, 4)  
        Range("F3") = Right(SheetName, 3)  
        Range("F4") = LastRow - 1  
        Range("F5").Formula = "=sum(c2:c" & LastRow & ")"  
        Range("F6").Formula = "=F5/F4"  
        Range("F6").NumberFormat = "#,##0.00"  
  
        RowNumber = RowNumber + 1  
    'End While  
Wend  
  
End Sub
```

(General)

Copying

Sub Copying()

Range("B2").Select

Range(Selection, Selection.End(xlToRight)).Select

Range(Selection, Selection.End(xlDown)).Select

Selection.Copy

Range("H2").Select

ActiveSheet.Paste

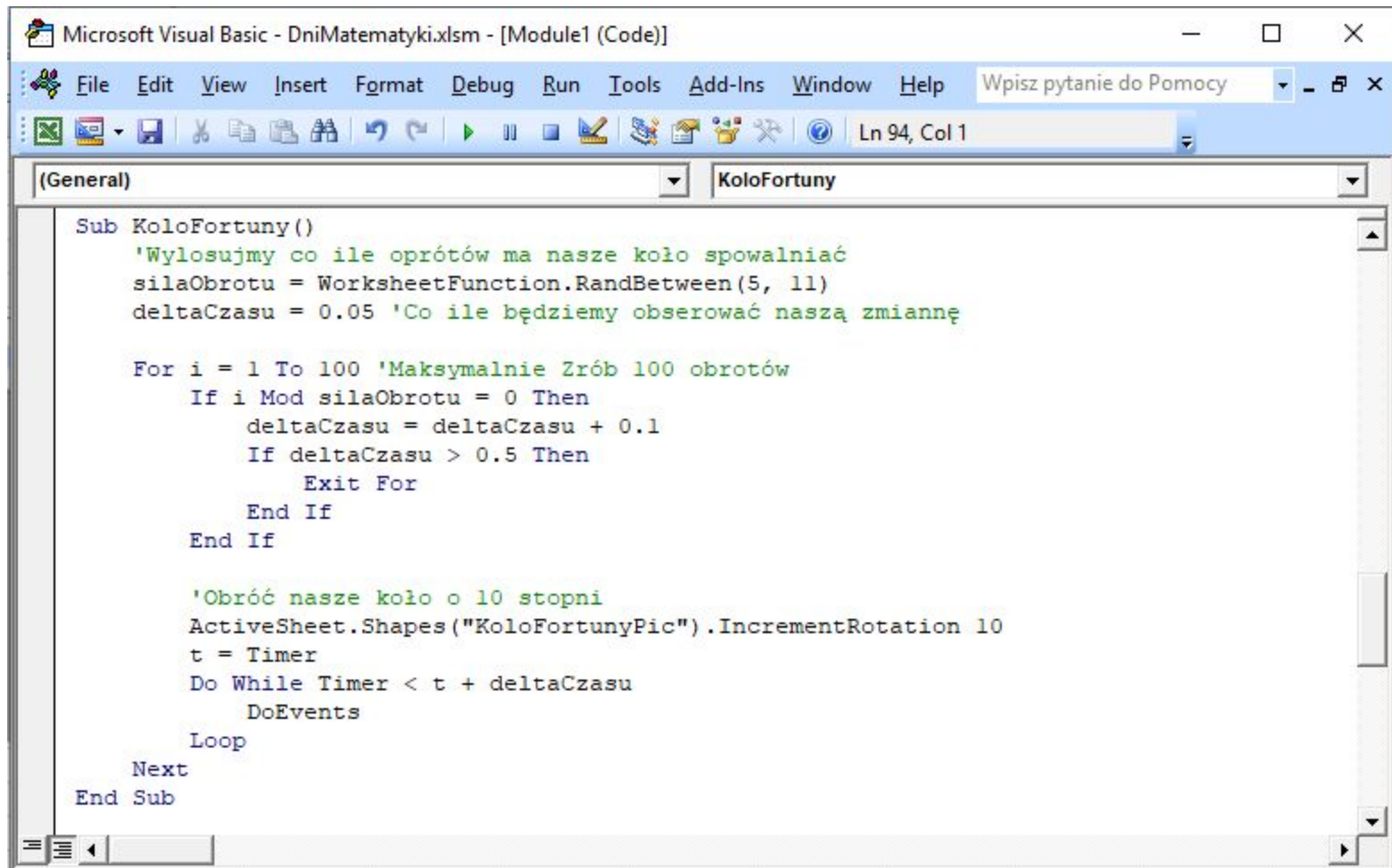
Range("M2").Select

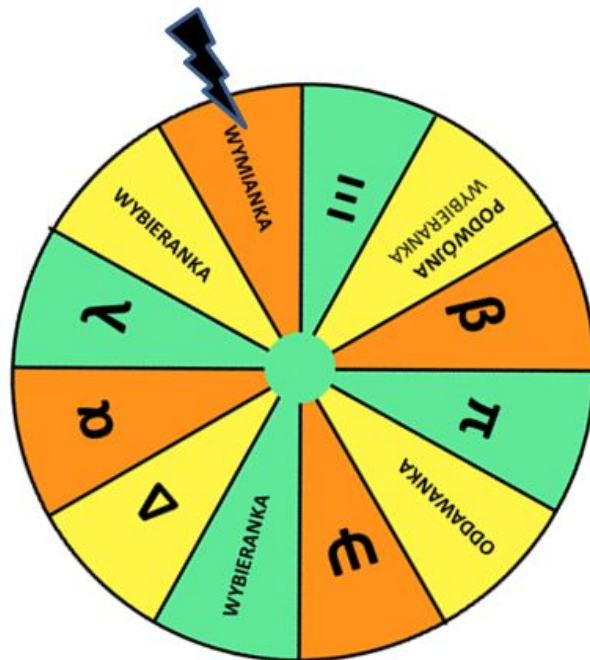
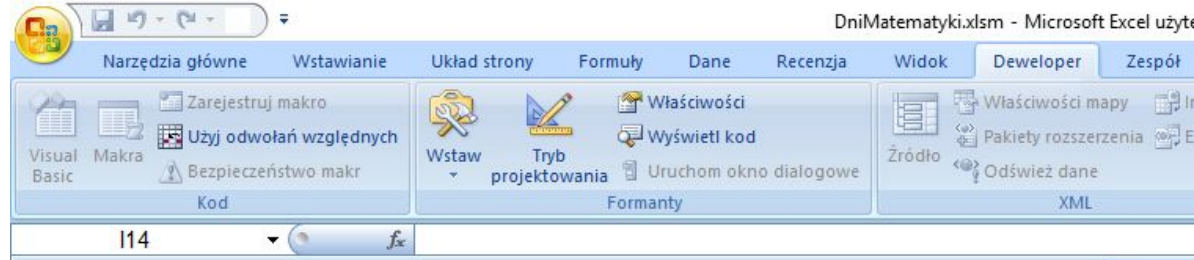
Selection.PasteSpecial Paste:=xlPasteValues, Operation:=xlNone, SkipBlanks _

:=False, Transpose:=False|

End Sub

Koło fortuny





Start

Dzięki!