Arkusz kalkulacyjny – opcje zaawansowane (edycja komórek i wypełnianie seryjne, edycja formuł, przegląd funkcji, wykorzystanie pilota danych, makrodefinicje), cz. 5

Zad. 1

Otwórz nowy plik Excela. Zapisz go pod nazwą VBAcz2.xlsm. Przejdź na kartę Deweloper i wybierz opcję Visual Basic. Następnie w oknie Project – VBAProject, kliknij prawym przyciskiem myszy na nazwie projektu VBAProject (VBAcz2.xlsm) i z menu kontekstowego wybierz opcję Insert -> Module. Z menu View na górze okna wybierz opcję Immediate Window (Ctrl + G). Zapisz projekt. W nowo otwartym oknie modułu wpisz następujący fragment kodu:

Sub Witaj()

Dim tekst As String

tekst = "Witaj na krótkim kursie języka Visual Basic for Applications"

Debug.Print tekst

End Sub

Powyższy kod tworzy procedurę o nazwie *Witaj* i deklaruje zmienną tekstową o nazwie *tekst*, do której zostaje następnie przypisany łańcuch znaków: *"Witaj na krótkim kursie języka Visual Basic for Applications"*. Następnie wyświetla ten tekst w oknie *Immediate Window*. Uruchom makro, naciskając klawisz *F5* na klawiaturze.

Zad. 2

Kliknij prawym przyciskiem myszy na nazwie projektu *VBAProject (VBAcz2.xlsm)* i z menu kontekstowego wybierz opcję *Insert -> Module*. Zapisz projekt. W nowo otwartym oknie modułu wpisz następujący fragment kodu:

Sub Zmienne()

Dim x As Integer Dim y As String Dim z As Double x = 5 y = "Jan" z = 105.632 Debug.Print x Debug.Print y Debug.Print z

End Sub

Powyższy kod tworzy procedurę o nazwie *Zmienne* i deklaruje trzy zmienne (*x*, *y*, *z*) typu liczby całkowitej, łańcucha znaków oraz liczby zmiennoprzecinkowej o podwójnej precyzji. Następnie przypisuje zmiennym odpowiednie wartości lub wyrażenia: liczbę całkowitą, łańcuch znaków oraz liczbę ułamkową, po czym wyświetla te wartości w oknie *Immediate Window*. Uruchom makro, naciskając klawisz *F5* na klawiaturze.

Zad. 3

Zmodyfikuj powyższe makro zmieniając postać polecenia Print na następującą:

Debug.Print x, y, z

Zapisz makro i uruchom je naciskając F5 na klawiaturze.

Zad. 4

W przypadku, kiedy wynik polecenia *Print* jest zbyt długi, można go umieścić w pliku tekstowym, zamiast w oknie *Immediate Window*. Kliknij prawym przyciskiem myszy na nazwie projektu *VBAProject* (*VBAcz2.xlsm*) i z menu kontekstowego wybierz opcję *Insert -> Module*. Zapisz projekt. W nowo otwartym oknie modułu wpisz następujący fragment kodu (zwróć uwagę, aby w instrukcji *Open* umieścić właściwą ścieżkę dostępu do pliku; podana ścieżka jest tylko przykładowa):

Sub DrukujdoPliku()

Dim s As String

Dim num As Integer

num = FreeFile()

Open "C:\Users\stud\Documents\pakiety1\wynik.txt" For Output As #num

s = "Witaj na krótkim kursie języka Visual Basic for Applications" Debug.Print s ' wyświetla wynik w oknie *Immediate Window* Print #num, s ' zapisuje wynik w pliku

Close #num

End Sub

Powyższy kod tworzy procedurę o nazwie *DrukujdoPliku*, deklaruje dwie zmienne (*s* oraz *num*) typu łańcucha znaków oraz liczby całkowitej, przypisuje zmiennej *num* numer kolejnego pliku, który nie został jeszcze użyty i który może być wykorzystany w instrukcji *Open*, otwiera pusty plik o nazwie *wynik.txt* w lokalizacji określonej przez ścieżkę dostępu, jednocześnie wskazując go jako miejsce, w którym ma zostać zapisany wynik działania polecenia *Print*, przypisuje zmiennej *s* łańcuch znaków *Witaj na krótkim kursie języka Visual Basic for Applications*, wyświetla tekst przypisany do zmiennej *s* w oknie *Immediate Window* oraz zapisuje ten sam tekst w pliku *wynik.txt*, po czym zamyka ten plik. Znak ' umożliwia dodanie komentarza do fragmentów kodu, zaś znak # - który plik ma zostać wykorzystany.

Uruchom makro naciskając *F5* na klawiaturze. Przejdź do folderu wskazanego w ścieżce dostępu i otwórz zapisany wcześniej plik *wynik.txt*.

Zad. 5

Kliknij prawym przyciskiem myszy na nazwie projektu *VBAProject (VBAcz2.xlsm)* i z menu kontekstowego wybierz opcję *Insert -> Module*. Zapisz projekt. W nowo otwartym oknie modułu wpisz następujący fragment kodu:

Public Sub Silnia()

Dim liczba As Integer Dim numer As Integer Dim silnia As Integer numer = 5 silnia = 1 For liczba = 1 To numer silnia = silnia * liczba Next liczba

Debug.Print "Wartość silni dla liczby 5:", silnia

End Sub

Powyższy kod tworzy procedurę (widoczną w oknie makr) o nazwie *Silnia*, deklaruje trzy zmienne (*liczba, numer* oraz *silnia*) typu liczby całkowitej, przypisuje zmiennej *numer* wartość 5, zmiennej *silnia* liczbę 1, następnie definiuje pętle *for* przechodzącą przez poszczególne liczby w przedziale od 1 do 5 i dla każdej z nich wyliczającą wartość silni poprzez przemnożenie aktualnej wartości silni i kolejnej liczby (od 1 do 5), jednocześnie za każdym razem aktualizując wartość silni (przypisując zmiennej *silnia* wartość wyliczoną w wyniku wykonania danej iteracji), po czym wyświetla końcową wartość zmiennej *silnia* (120).

Uruchom makro naciskając F5 na klawiaturze lub naciskając zielony przycisk na pasku narzędzi (Run Macro).

Zad. 6

Zmodyfikuj poprzednie makro poprzez wstawienie polecenia *Print* wewnątrz pętli *for* (w ten sposób możliwe jest wyświetlanie aktualnej wartości silni dla każdej iteracji pętli *for*):

Public Sub Silnia()

Dim liczba As Integer Dim numer As Integer Dim silnia As Integer numer = 5 silnia = 1 For liczba = 1 To numer silnia = silnia * liczba Debug.Print "Aktualna wartość silni dla liczby ", liczba, "to: ", silnia Next liczba

End Sub

Uruchom zmodyfikowane makro naciskając F5 na klawiaturze lub naciskając zielony przycisk na pasku narzędzi (*Run Macro*).

Zad. 7

Kliknij prawym przyciskiem myszy na nazwie projektu *VBAProject (VBAcz2.xlsm)* i z menu kontekstowego wybierz opcję *Insert -> Module*. Zapisz projekt. W nowo otwartym oknie modułu wpisz następujący fragment kodu:

Sub AktywnySkoroszyt()

Dim count As Long

For count = 1 To Workbooks.count

Debug.Print Workbooks(count).FullName

Next count

Debug.Print "Liczba skoroszytów: ", count

End Sub

Powyższy kod tworzy procedurę *AktywnySkoroszyt*, deklaruje zmienną *liczba* typu długiego (liczby, które są zbyt duże, żeby przypisać je zmiennej typu liczby całkowitej), definiuje pętle *for* przechodzącą przez poszczególne wartości od 1 do maksymalnej liczby aktywnych skoroszytów, po czym wyświetla nazwy kolejnych skoroszytów oraz ich liczbę.

Uruchom makro naciskając F5 na klawiaturze lub naciskając zielony przycisk na pasku narzędzi (Run Macro).

Źródła:

https://www.wallstreetmojo.com/vba-debug-print/

https://learn.microsoft.com/pl-pl/dotnet/visual-basic/language-reference/statements/dimstatement

https://learn.microsoft.com/en-us/office/vba/language/reference/user-interface-help/freefilefunction

https://learn.microsoft.com/en-us/office/vba/language/reference/user-interface-help/openstatement

https://learn.microsoft.com/en-us/office/vba/api/excel.workbook.close

https://stackoverflow.com/questions/42344454/vba-what-does-the-pound-sign-mean-before-variable