VBA Best Practices

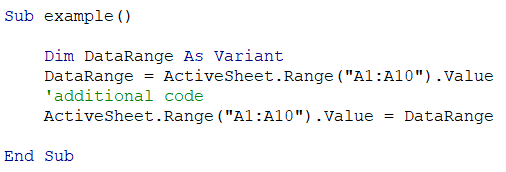
* General
  + Option Explicit – Declare variables
  + Create .xlsm file separately, containing no input data
  + Version handling – add a version number to the filename, and increment it when the code is changed

e.g.: example\_v1.0.xlsm

* + optionally add a version history page to the macro file and log what was changed, when and in which version
  + Start with recording a macro, and modify the recorded code according to your needs
  + To avoid flashing (it can happen when you use multiple files and activate them) use Application.ScreenUpdating = False
  + Activate Workbooks, Sheets even if you have only one file and one sheet
  + Use descriptive variable naming – use camel case: e.g: fileNames, or pascal case: e.g.: FileName
  + Document Your Work (at least where you use complex logic)
  + Use LBound, UBound while you loop through Arrays
  + When you assign a value to a Range be careful with the data type and format (it can be different on someone else’s computer)
  + When you remove a Cell / Row in a loop, start the loop from the last row and use Do while (e.g : in separate file: "C:\Users\A46322145\Desktop\delete\_row\_example\_v*x.y*.xlsm")
  + Create log files in txt format to debug faster, name: LogFile*Timestamp*.txt, list the time and actions (step in a function, step out of a function, return value of a function, etc.)
* Performance
  + Switch default properties off when you start and on when you finish
    - Application.Calculation = xlManual -> Application.Calculation = xlAutomatic
    - Application.EnableEvents = False - > Application.EnableEvents = True
  + In long-running cycles add Do Events
  + Working with big files:
    - Copy values with “=” instead of copy function

e.g.: Range("B1:B1000").Value = Range("A1:A1000").Value

* + - Work with Arrays (Collections, Dictionaries) instead of Ranges



* + Use Excel functions in the code also

e.g.: Application.WorksheetFunction.VLookup

* + Use VB constants when available (VbOK, VbYes...)
  + Use Named ranges
  + Use With, End With to avoid long object chains multiple times
  + Use For Each instead of For if no cycle variable is needed