

```
=====
VBA - OBJECT VARIABLES (WORKBOOK)
=====
```

#### EXAMPLE 2: WORKBOOK OBJECT VARIABLE

```
-----  
Sub ObjectVariableWorkbook()  
    Dim Wb As Workbook  
  
    On Error GoTo ErrorHandler  
  
    ' Open workbook and assign to variable  
    Set Wb = Workbooks.Open("C:\Users\Akshay\OneDrive\Documents\DataFile.xlsx")  
  
    ' Activate workbook  
    Wb.Activate  
  
    MsgBox "Workbook " & Wb.Name & " is opened and activated!"  
  
    ' Release object variable  
    Set Wb = Nothing  
  
    Exit Sub  
  
ErrorHandler:  
    MsgBox "Error is: " & Err.Number & Err.Description  
End Sub
```

#### How It Works:

- Dim Wb As Workbook = declare object variable
  - Set Wb = Workbooks.Open = assign workbook
  - Wb.Activate = activate that workbook
  - Wb.Name = get workbook name
  - Set Wb = Nothing = release object (important!)
- 

#### KEY POINTS - WORKBOOK OBJECT

- ```
-----  
✓ Declare: Dim Wb As Workbook  
✓ Open file: Set Wb = Workbooks.Open(path)  
✓ Activate: Wb.Activate  
✓ Access name: Wb.Name  
✓ Error handling: On Error GoTo  
✓ Release memory: Set Wb = Nothing (important)  
✓ Always clean up object variables
```
- 

#### ERROR HANDLING

```
-----  
On Error GoTo ErrorHandler  
→ If error happens, jump to ErrorHandler  
  
ErrorHandler:  
    MsgBox "Error is: " & Err.Number & Err.Description  
    → Show error number and message  
  
    Err.Number = error code  
    Err.Description = error message
```

---

## RELEASING OBJECT VARIABLES

---

IMPORTANT: Set Wb = Nothing

Why?

- Freed memory
- Closes file reference
- Prevents locks
- Good practice

Always do:

Set Wb = Nothing  
At end of Sub

---

## OBJECT VARIABLE WORKFLOW

---

1. Declare: Dim Wb As Workbook
2. Create/Assign: Set Wb = Workbooks.Open(path)
3. Use: Wb.Activate, Wb.Name, etc.
4. Handle errors: On Error GoTo
5. Release: Set Wb = Nothing

=====

FILE: VBA\_Object\_Variables\_Workbook.txt

=====