엑셀 VBA를 활용한 업무자동화

Visual Basic For Application

강사 **유영현**

엑셀 VBA를 활용한 업무자동화

강사 유영현

https://cafe.naver.com/alvinvba

강의 자료 / 실습 예시 / 수 많은 매크로 파일

닉네임을 꼭 아래와 같이!!

탈잉 유영현

VBA란? Visual Basic for Application

마이크로소프트에서 제공하는 오피스 프로그램들에 탑재 된 **프로그래밍 언어**

(Excel, Access, Power Point 등)

대량의 엑셀 파일 수집

동일한 유형의 수 많은 파일을
버튼 클릭 한번으로 2초 안에
수집할 수 있습니다.

데이터 분석 & Report 하기

매일 쌓이는 데이터를
버튼 클릭 한번으로 쉽고 빠르게
원하는 형식으로
Report 할 수 있습니다.

원하는 형태로 가공하기

데일리로 나오는 데이터를
원하는 형태로 쉽게
가공 할 수 있습니다.

엑셀로 하는 모든 작업

실무에서의 모든 엑셀작업을
VBA로 업무자동화 하여
아까운 시간을 효과적으로
줄일 수 있습니다.

코드를 작성해보자

01. 코딩하기에 적합한 여러가지 셋팅 따라하기

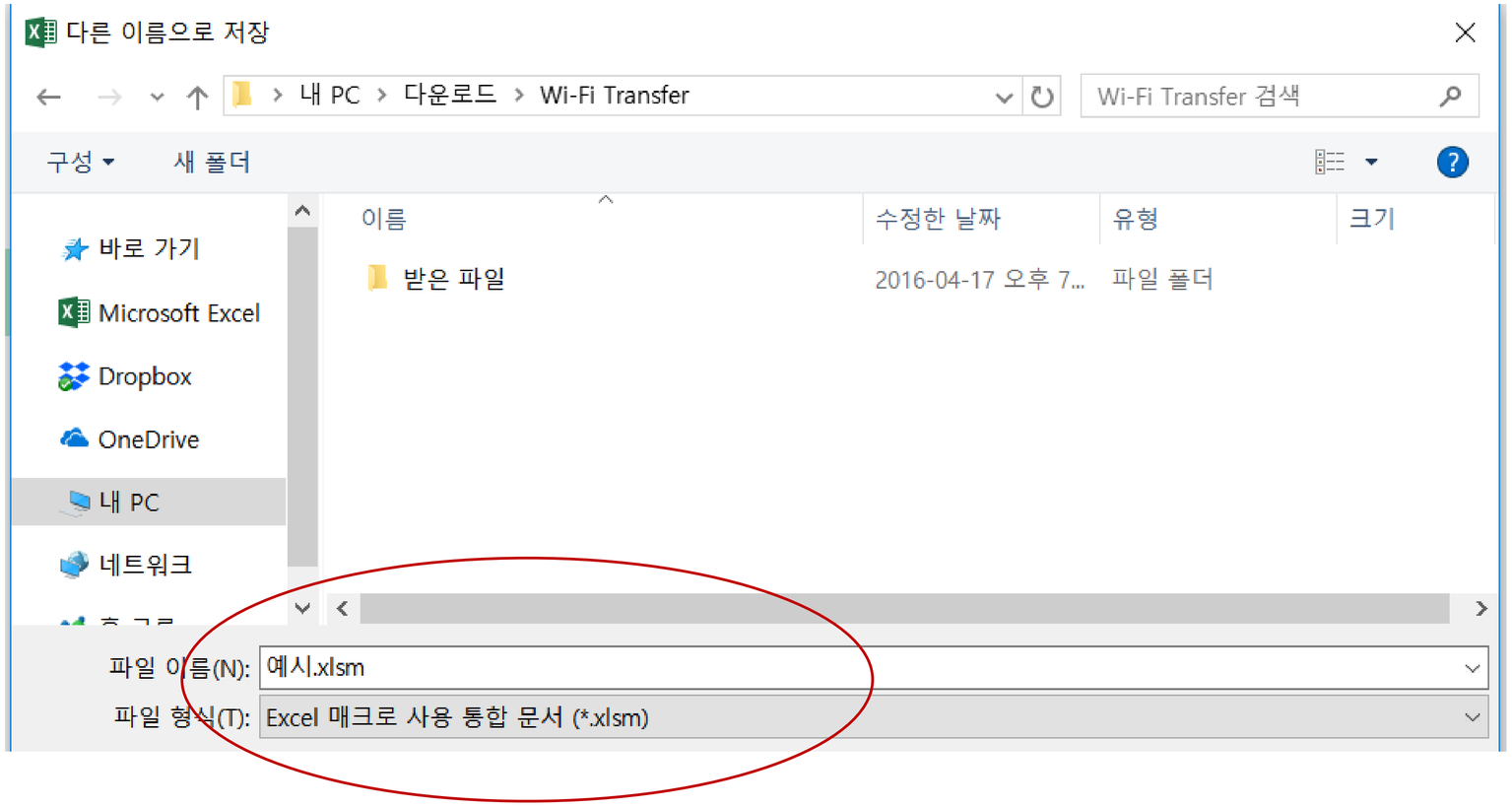
02. 코드를 작성하는 방법 소개 (VBE, Module, Procedure)

03. 무작정 간단한 코드 따라해보기 (Sub Procedure)

04. 주석으로 코드에 대한 메모하기

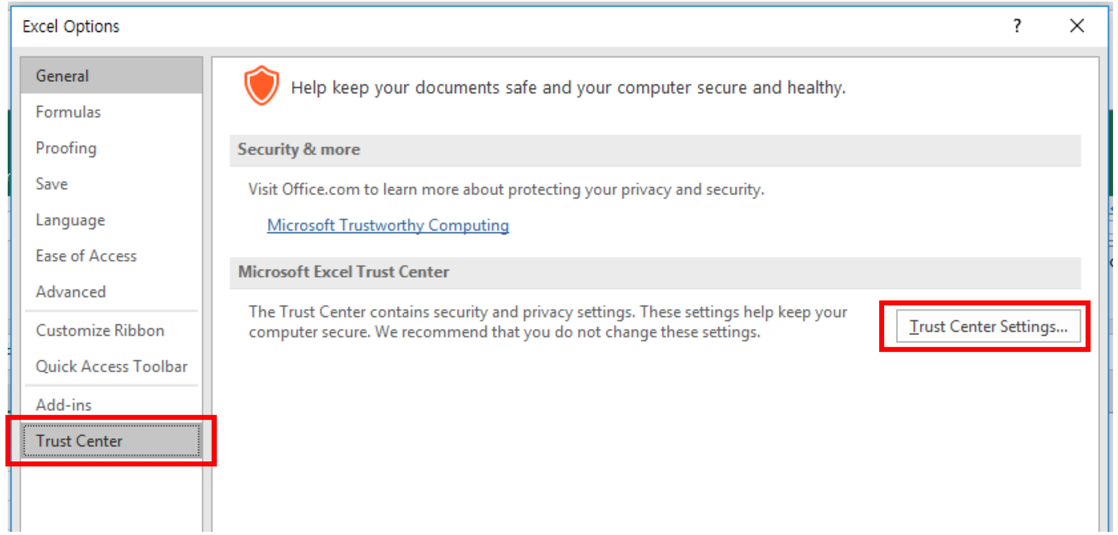
파일 저장하는 방법

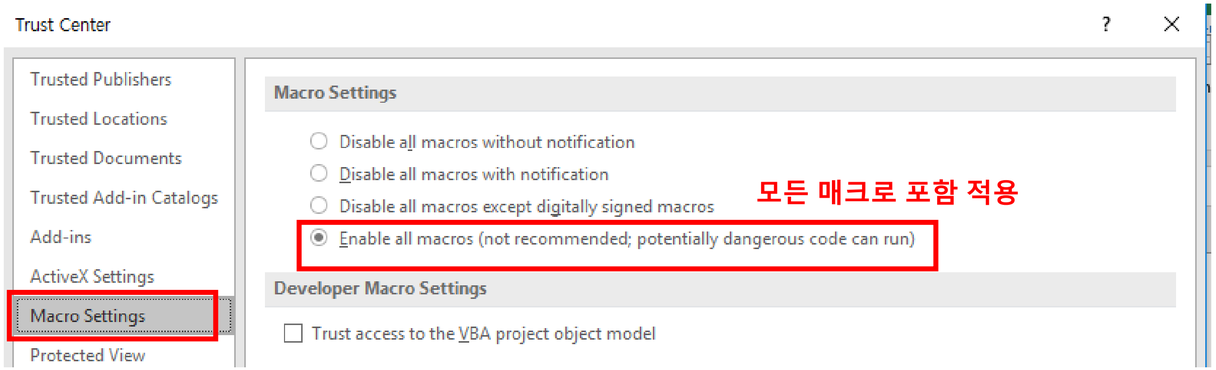
반드시 “xlsm” 확장자로 저장하여야 VBA코드가 같이 저장됩니다.



매크로 사용 전 옵션 설정! (필수)

1) 엑셀 옵션
2) 보안 센터 설정 (Trust Center)
3) 매크로 설정 → 모든 매크로 포함 적용
4) 적용하고 엑셀 저장 후 다시 엑셀 켜기

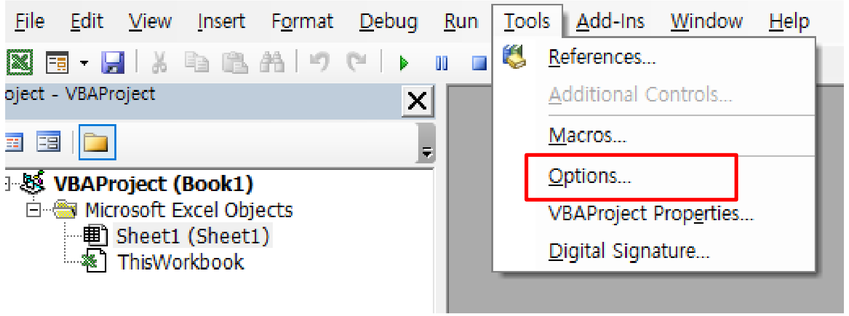




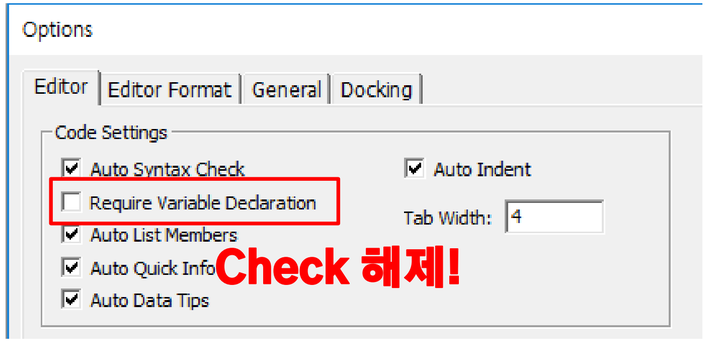
코딩하기에 적합한 셋팅

엑셀에서 ALT + F11 을 같이 눌러보세요!
VBE가 OPEN됩니다!

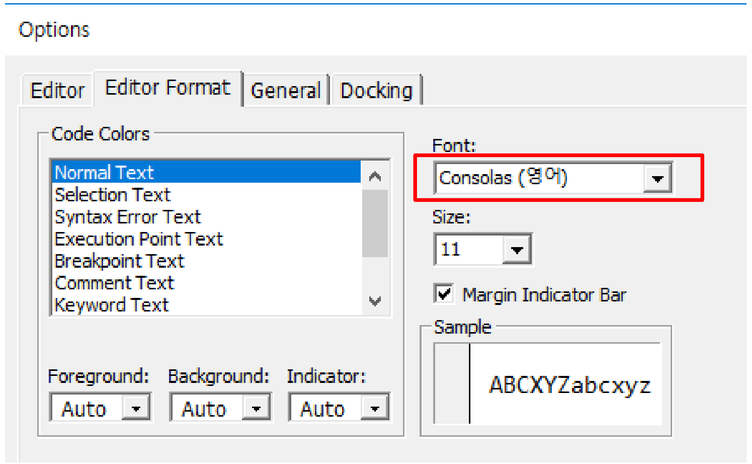
1) VBE 상단의 Tools (도구) -> Option (옵션) 클릭!



2) Editor 탭 클릭! (변수선언 요구 체크 해제)



3) Editor Format 탭 클릭! (가독성 좋은 폰트로 변경)



코드를 작성하는 방법

1 VBE

코드를 편집하는 공간
Visual Basic Editor

✓ALT + F11
✓개발도구 – Visual Basic

2 모듈

코드를 작성하는 도화지
Module

✓VBE – 삽입 – 모듈 삽입

3 프로시저

명령코드의 집합체
Procedure

✓명령문들이 순서대로 진행

Sub
Procedure

명령문들이 순차적으로 실행
결과값은 반환되지 않는다.

(보통 버튼에 연결)

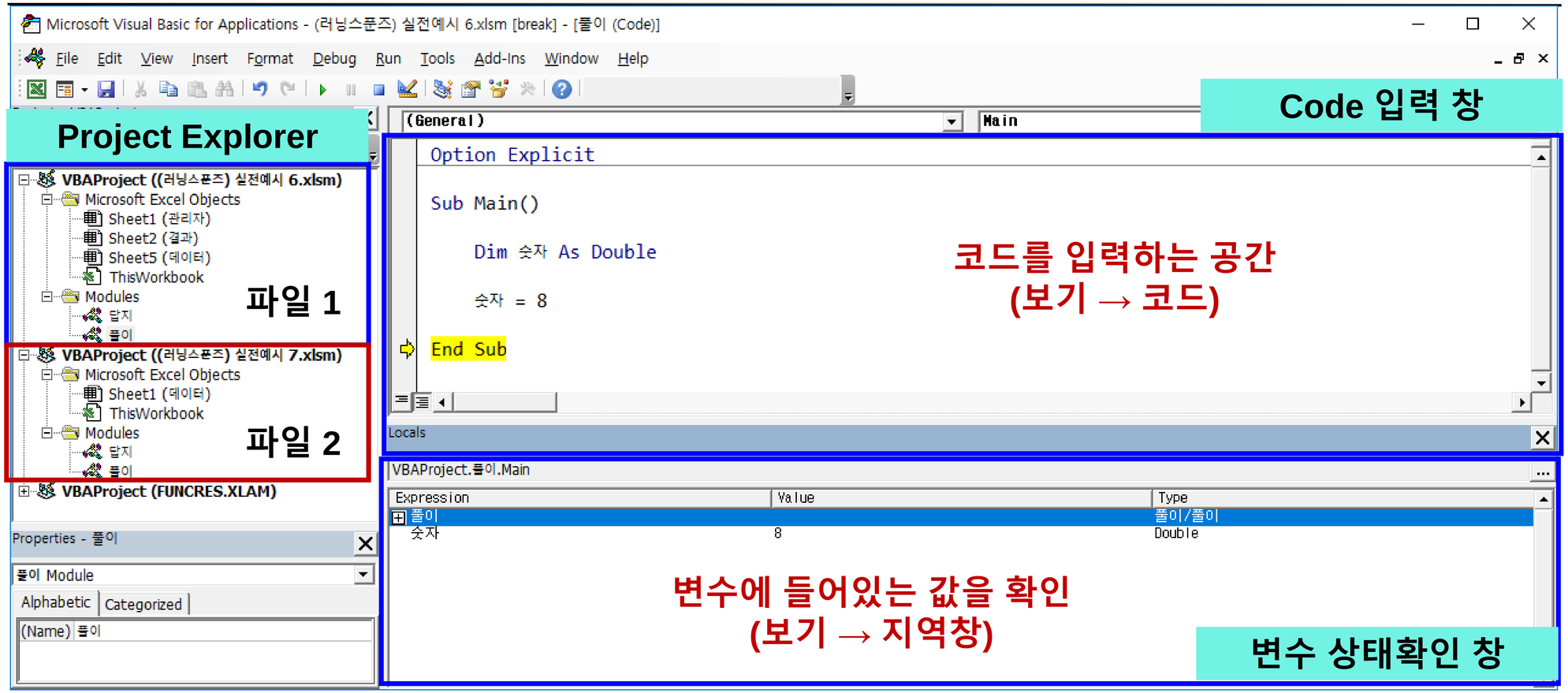
Function
Procedure

사용자 정의 함수
(=Sum, Match 등)

명령문에 의해 계산이 처리되고
결과값을 반환한다.

VBE interface

엑셀 파일이 몇개가 켜지든 하나의 VBE 창에서 컨트롤을 할 수 있습니다.



Sub 프로시저 (개념)

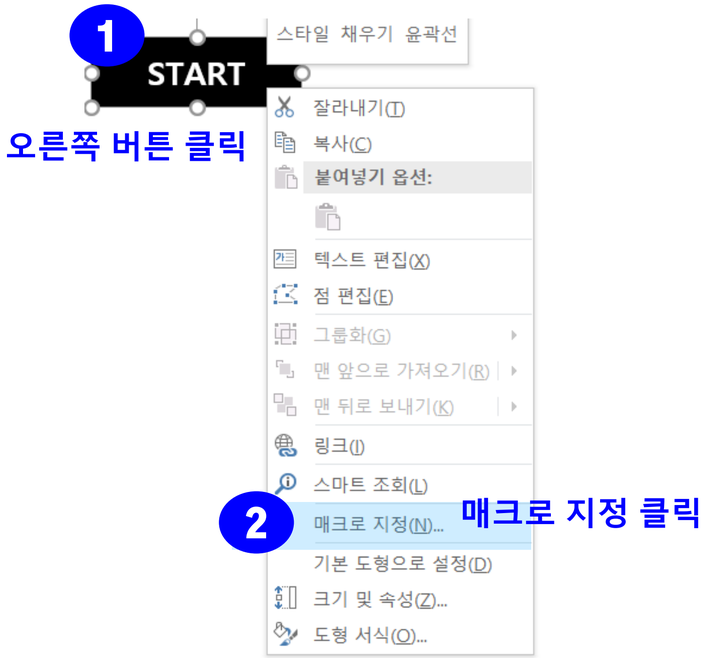
작성코드 예시	설명
Sub 예시1() 명령1 명령2 명령3 명령4 명령5 **End Sub** **프로시저는 이렇게 구성**	*** 프로시저 네이밍 규칙 :** 한글 또는 영어로 시작 / 공백 포함 X *** 실행 방법 :** 프로시저 전체 실행 (F5) or / 한줄씩 실행 (F8) / 실행 중지 *** 꿀팁 :** "Sub 예시1" 까지만 타이핑 후 Enter를 누르면 "()" 와 "End Sub"가 자동 생성!

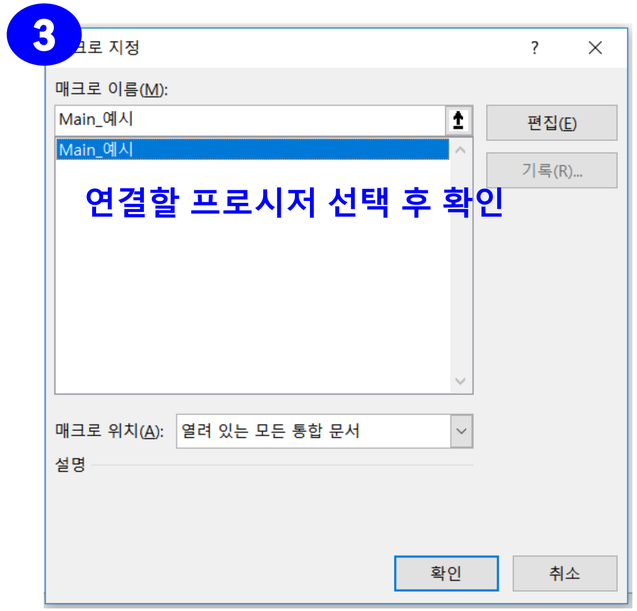
Sub 프로시저 (따라하기)

작성코드 예시	설명
Sub 예시1() Cells(1, 1).Value = 1 Cells(1, 1).Font.Size = 15 Cells(1, 1).Font.Bold = True Cells(1, 1).Interior.Color = RGB(0,0,255) Cells(1, 1).Font.Color = RGB(255,255,255) **End Sub**	① A1셀의 **값을** 1로 변경 ② A1셀의 **폰트 사이즈를** 15로 변경 ③ A1셀의 **폰트 굵게** 설정 ④ A1셀의 **배경색** Blue로 지정 ⑤ A1셀의 **폰트색** White로 지정 **＊Key Point :** 프로시저의 진행방향은 위에서 아래다! 그리고 오른쪽에 있는 값이 왼쪽으로 대입된다.

프로시저를 버튼에 연결하기

프로그램 배포 시, 버튼에 프로시저를 연결합니다. (사용자 UI 편의)





주석으로 코드에 메모하기

특정 명령문을 **Skip** 하고자 할때,

특정 명령문에 대한 **메모**를 하고자 할때,

주석을 사용합니다.

```
Sub 예시1()

    Cells(1, 1).Value = 1    'A1셀의 값을 1로 변경
    Cells(1, 1).Font.Size = 15   'A1셀의 폰트 사이즈를 15로 변경
    Cells(1, 1).Font.Bold = True     'A1셀의 폰트 굵게 설정
    Cells(1, 1).Interior.clor = RGB(0, 0, 255)  'A1셀의 배경색 Blue로 지정
    Cells(1, 1).Font.Color = RGB(255, 255, 255) 'A1셀의 폰트색 White로 지정

    'Cells(1, 1).Value = "Skip"

End Sub
```

명령문 뒤에 따옴표 ' 사용하여 메모

명령문 앞에 따옴표 ' 사용하여 Skip

셀 개체에 접근해보기

01. 엑셀을 이루고 있는 개체(Object) 소개 – 속성, 메서드

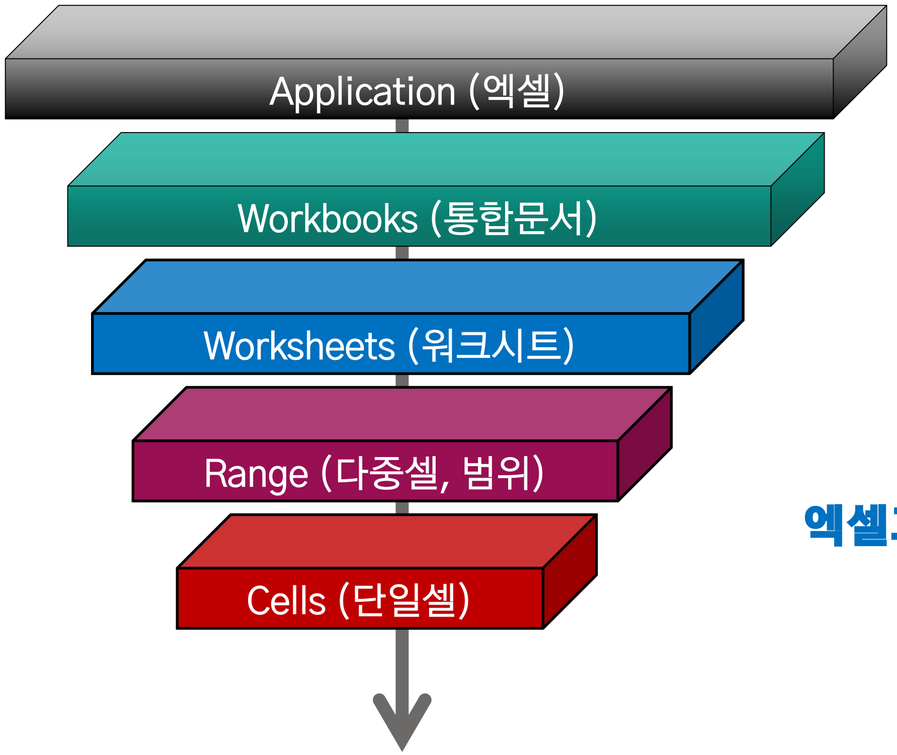
02. Cells(단일셀) 개체 이해하기

03. Range(다중셀) 개체 이해하기

04. 삭제, 복사 메서드 이해하기

05. 행전체(Rows), 열전체(Columns)를 어떻게 컨트롤 할수 있을까?

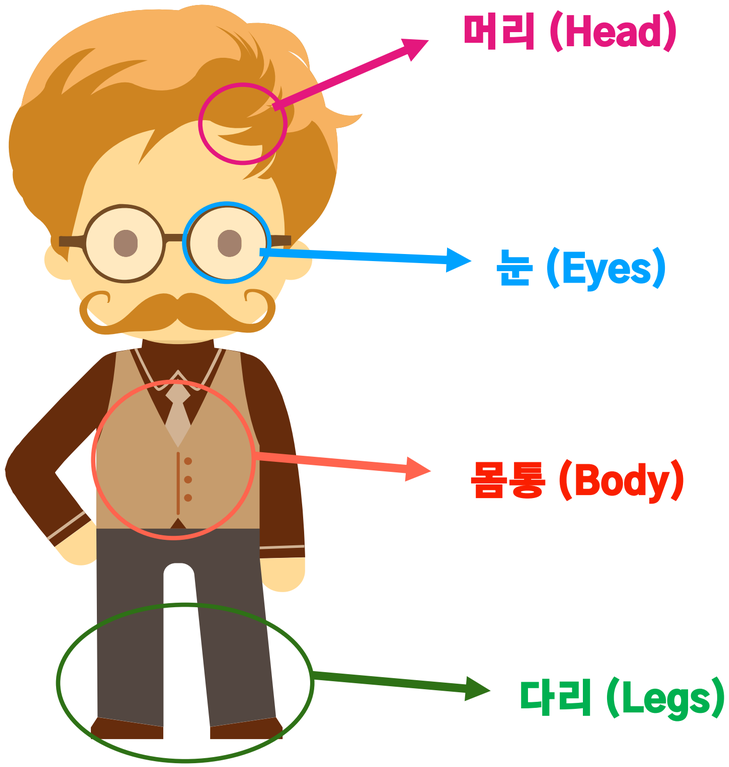
엑셀을 이루고 있는 개체 (Object)



엑셀의 개체 (Object)

=

엑셀과 사람은 개체로 이루어짐



사람의 개체 (Object)

속성과 메서드 (Properties & Method)

속성은 **개체가 가질수 있는 특징**

메서드는 **개체가 할 수 있는 행동**

개체 구분	속성 (특징)	메서드 (행동)
Workbooks	✓Path (파일 주소) ✓Name (파일명)	✓Close (통합문서 종료) ✓Open (통합문서 열기)
Worksheets	✓Name (시트명) ✓Filter Mode (필터의 유무)	✓Add (워크시트 추가) ✓Copy (워크시트 복제)
Range	✓Value (셀의 값) ✓Interior (셀의 배경) ✓Font (셀의 폰트)	✓Clear (범위 삭제) ✓Copy (범위 복사)
Cells		

속성(Properties)

속성은 개체가 가지고 있는 특징입니다.

속성은 특정 값으로 변경 할 수 있다!

개체 (Object)

① A1셀의 Value 속성을 1로 변경해줘.
② 활성화된 시트의 Name 속성을 "관리자"로 변경해줘.
③ A1셀의 Font.Size를 14로 변경해줘.

속성 변경

속성 반환

① A1셀의 Value 값이 뭐니?
② 활성화된 시트의 Name 이 뭐니?
③ A1셀의 Font.Size 가 몇이니?

속성은 반환 받을 수 있다!

메서드(Method)

메서드는 개체가 할 수 있는 행동입니다.

개체 (Object)

메서드는 명령만 가능 하다!

① A1셀을 Clear 해줘
② A1:C100의 범위에 Autofilter (자동필터) 해줘
③ 활성화된 시트를 Delete (삭제) 해줘

명령

Cells, Range 개체의 속성/메서드

개체 종류	구분	기능 설명	기능	예시
Cells / Range	속성	값	Value	Cells(1, 1).Value = 5
		배경 색상	Interior.Color	Cells(1, 1).Interior.Color = vbBlue
		글꼴 색상	Font.Color	Cells(1, 1).Font.Color =vbBlue
		글꼴 굵기	Font.Bold	Cells(1, 1).Font.Bold = True
		글꼴 이름	Font.Name	Cells(1, 1).Font.Name = “맑은 고딕”
		테두리 스타일	Borders.LineStyle	Cells(1, 1).Borders.LineStyle = True
		수평 맞춤	HorizontalAlignment	Cells(1, 1).HorzontalAlignment = xlCenter
		셀 서식	NumberFormatLocal	Cells(1, 1).NumberFormatLocal = “#,##0”
		행/열 개수 파악	End	Cells(Rows.Count, 1).End(xlUp).Row
	메서드	값 + 서식 초기화	Clear	Cells(1, 1).Clear
		값 초기화	ClearContents	Cells(1, 1).ClearContents
		서식 초기화	ClearFormats	Cells(1, 1).ClearFormats
		셀 삭제	Delete	Cells(1, 1).Delete Shift:=xlUp
		셀 복사	Copy	Cells(1, 1).Copy Destination:=Cells(2, 1)
		자동 필터	Autofilter	Range(Cells(1, 1),Cells(100, 5)).Autofilter Field:=1,Criteria1:=“사과”
		정렬	Sort	범위.Sort Key1:=기준이 될 범위, Order1:=xlAscending,Header:=xlYes

Sheets / Workbooks 개체의 속성/메서드

개체 종류	구분	기능 설명	기능	예시
Sheets	속성	시트 개수	Count	Sheets.Count
		시트 명	Name	Activesheet.Name = “관리자”
		시트 숨기기 유무	Visible	Activesheet.Visible = xlSheetVisible
	메서드	시트 추가	Add	Sheets.Add (After:=Sheets(Sheets.Count)).Name = “새로운 시트”
		시트 복사	Copy	Sheets(“관리자”).Copy After:=Sheets(Sheets.Count)
		시트 삭제	Delete	Sheets(“관리자”).Delete
		시트 활성화	Activate	Sheets(“관리자”).Activate
		시트 프린트	PrintOut	Sheets(“관리자”).PrintOut

개체 종류	구분	기능 설명	기능	예시
Workbooks	속성	파일 주소	Path	Thisworkbook.Path
	메서드	문서 열기	Open	Workbooks.Open FileName:=“파일주소”
		문서 닫기	Close	Activeworkbook.Close SaveChanges:=False
		문서 추가	Add	Sheets.Add
		문서 저장	Save	Activeworkbook.Save
		문서 다른이름 저장	SaveAs	Activeworkbook.SaveAs FileName:=“파일주소”

Cells 개체 (개념)

작성코드 예시	설명
★ **Cells 개체 표현 :** Cells(행 번호, 열번호)	* 행 번호 : 2 / 열 번호 : A B C D E A는 1, B는 2 ~~~~~
Sub 예시2() Cells(1, 1).Value = 1 Cells(2, 6).Value = "안녕하세요" Cells(3, 3).Value = 5 **End Sub**	① A1셀의 **값을** 1로 변경 ② F2셀의 **값을 "안녕하세요"**로 변경 ③ C3셀의 **값을** 5로 변경 *** Key Point :** 문자를 넣을때는 쌍따움표("텍스트") 안에 넣어서 전달해야합니다. 단일 셀을 사용할 때는 Cells 개체를 사용합니다.

Value 속성 (개념)

특정 셀에 값을 지정할 때 사용하는 속성입니다.

사용 방법

기준셀.Value = 값 (Value 는 생략 가능)

예시

- Cells(1, 1).Value = 5
- Cells(5, 2).Value = “VBA 열심히!”
- Cells(1, 2) = 3
- Cells(1, 3).Value = Cells(1, 1).Value

Font.Color 속성 (개념)

특정 셀의 폰트 색상을 지정할 때 사용하는 속성입니다.

사용 방법

기준셀.Font.Color = 색상값 (색상값은 2종류로 표현 가능)

RGB 방식

RGB(숫자, 숫자, 숫자)

내장함수 방식

vBWhite
vBBlue
vBYellow

예시

- Cells(5, 2).Font.Color = RGB(255, 0, 0)
- Cells(1, 1).Font.Color = vbBlue

Font.Bold 속성 (개념)

특정 셀의 폰트 굵기를 지정할 때 사용하는 속성입니다.

사용 방법

기준셀.Font.Bold = True / False (True는 굵게 / False는 기본값)

예시

- Cells(5, 2).Font.Bold = True
- Cells(1, 1).Font.Bold = False

Font.Name 속성 (개념)

특정 셀의 폰트 이름를 지정할 때 사용하는 속성입니다.

사용 방법

기준셀.Font.Name = “폰트 이름” (실제 있는 폰트만 가능!)

예시

- Cells(5, 2).Font.Name = “맑은 고딕”
- Cells(1, 1).Font.Name = “맑은 고딕”

Cells 개체 (실습)

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

	A	B	C	D
1	VBA			
2		열심히		
3			**2018**	
4		배워봐요		
5				
6	재밌는 수업			

Cells 개체 (실습)

작성코드 예시	설명
Sub 예시3()	
Cells(1, 1).Value = “VBA”	① A1셀의 **값을 “VBA”** 로 변경
Cells(1, 1).Font.Color = RGB(255, 0, 0)	② A1셀의 **폰트색 Red** 로 변경
Cells(6, 1).Value = “재밌는 수업 ”	③ A6셀의 **값을 “재밌는 수업 ”** 로 설정
Cells(2, 2).Value = “열심히”	④ B2셀의 **값을 “열심히 ”** 로 설정
Cells(4, 2).Value = “배워봐요”	⑤ B4셀의 **값을 “배워봐요 ”** 로 설정
Cells(4, 2).Font.Color = RGB(255, 0, 0)	⑥ B4셀의 **폰트색 Red** 로 변경
Cells(3, 3).Value = 2018	⑦ C3셀의 **값을 “2018 ”** 로 설정
Cells(3, 3).Font.Bold = True	⑧ C3셀의 **폰트 굵게** 변경
End Sub	

Range 개체 (개념)

작성코드 예시	설명
★ **Range 개체 표현 :** Range(시작셀, 마지막셀)	* 예시 : Range(Cells(1,1),Cells(5,1)) ▶ A1:A5 범위 셀
Sub 예시3()	
Range(Cells(1, 1),Cells(5, 1)).Value = 1	① A1:A5 범위 셀의 **값을 1**로 변경
Range(Cells(1, 1),Cells(5, 1)).Font.Bold = True	① A1:A5 범위 셀의 **폰트을 굵게**로 변경
Range(Cells(1, 1),Cells(5, 1)).Font.Color = vbRed	① A1:A5 범위 셀의 **폰트의 색상을 빨강으로** 변경
Range(Cells(1, 1),Cells(5, 1)).Borders.LineStyle = True	① A1:A5 범위 셀의 **테두리 생성**
End Sub	***Key Point :** 보통 Range 개체는 특정 범위에 대한 셀 서식을 수정 할때 사용합니다. 해당 범위에 대해서, 색상 / 가운데 정렬 / 셀서식 등등 수행

Interior.Color 속성 (개념)

특정 셀의 배경 색상을 지정할 때 사용하는 속성입니다.

사용 방법

기준셀.Interior.Color = 색상값 (색상값은 2종류로 표현 가능)

RGB 방식

RGB(숫자, 숫자, 숫자)

내장함수 방식

vBWhite
vBBlue
vBYellow

예시

- Cells(5, 2).Interior.Color = RGB(255, 0, 0)
- Cells(1, 1).Interior.Color = vbBlue

HorizontalAlignment 속성 (개념)

셀 맞춤을 지정할 때 사용하는 속성입니다.

사용 방법

기준셀. HorizontalAlignment = 매개변수 (매개변수 리스트)

- xlLeft (왼쪽 맞춤)
- xlCenter (가운데 맞춤)
- xlRight (오른쪽 맞춤)

예시

```
- Range(Cells(1, 1), Cells(100, 5)).HorizontalAlignment = xlCenter
```

NumberformatLocal 속성 (개념)

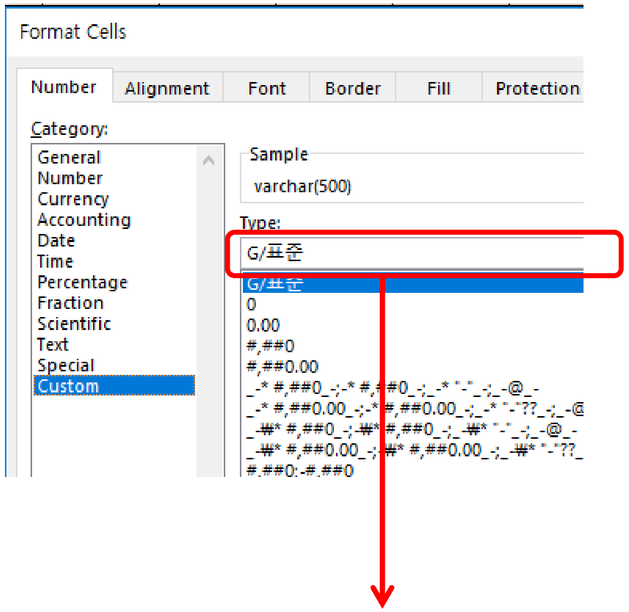
셀 사용자 지정 표시 형식을 지정할 때 사용하는 속성입니다.

사용 방법

기준셀. NumberformatLocal = “셀 표시 형식”

예시

- Range(Cells(1, 1), Cells(100, 5)).NumberformatLocal = “##,##0”
- Range(Cells(1, 6), Cells(100, 10)).NumberformatLocal = “YYYY-MM-DD”



입력 된 값을 그대로 사용!

Borders.LineStyle 속성 (개념)

셀 테두리를 지정할 때 사용하는 속성입니다.

사용 방법

기준셀.Borders.LineStyle = True / False

예시

- Range(Cells(1, 1), Cells(100, 5)).Borders.LineStyle = True
- Range(Cells(1, 1), Cells(100, 5)).Borders.LineStyle = False

Range 개체 (실습)

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

	A	B	C	D	E
1	안녕하세요	10,000	20,000	20,000	20,000
2	안녕하세요	10,000	20,000	20,000	20,000
3	안녕하세요	10,000	20,000	20,000	20,000
4	안녕하세요	10,000	20,000	20,000	20,000
5	안녕하세요	10,000	20,000	20,000	20,000

Value 속성

NumberformatLocal 속성

HorizontalAlignment 속성

Range 개체 (실습)

작성코드 예시	설명
Sub 예시3()	
Range(Cells(1, 1), Cells(6, 1)).Value = “안녕하세요”	① A1:A6 범위 셀의 **값을 “안녕하세요”** 로 변경
Range(Cells(1, 2), Cells(6, 2)).Value = 10000	② B1:B6 범위 셀의 **값을 “10000”** 로 변경
Range(Cells(1, 3), Cells(6, 5)).Value = 20000	③ C1:E6 범위 셀의 **값을 “20000 ”** 로 설정
Range(Cells(1, 3), Cells(6, 5)).NumberformatLocal = “##,##0”	④ C1:E6 범위 셀의 **표시 형식을 “##,##0 ”** 로 설정
Range(Cells(1, 1), Cells(6,5)).HorizontalAlignment = xlCenter	⑤ C1:E6 범위 셀의 **맞춤을 “xlCenter ”** 로 설정
End Sub	

Clear 메서드 (개념)

셀의 내용을 삭제할 때 사용하는 메서드입니다.

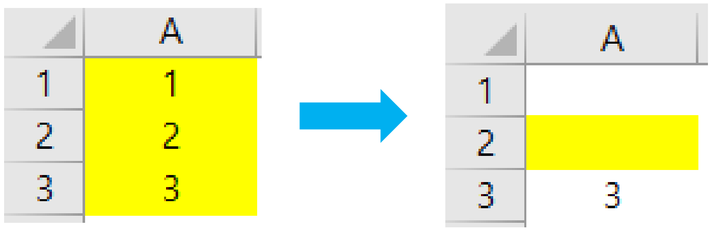
사용 방법

기준셀. Clear

(Clear 종류)
- Clear (내용+서식 삭제)
- ClearContents (내용 삭제)
- ClearFormats (서식 삭제)

예시

- Cells(1, 1).Clear
- Cells(2, 1).ClearContents
- Cells(3, 1).ClearFormats



Delete 메서드 (개념)

셀 자체를 삭제할 때 사용하는 메서드입니다.

사용 방법

기준셀. Delete Shift:=매개변수

생략 가능

기준셀 매개항목

```
Range(Cells(1, 1), Cells(5, 1)).Delete shift:=xlup
Delete([Shift])
```

매개변수

xlUp (위로 삭제)

xlToLeft (왼쪽으로 삭제)

예시

- Range(Cells(1, 1),Cells(5, 1)).Delete xlUp
- Cells(1, 5).Delete Shift:=xlToLeft

Clear / Delete 메서드의 차이

Clear 메서드는 셀의 위치적 변화가 없습니다. (기존 형태를 유지하면서 데이터를 삭제 할때 사용)

Delete 메서드는 셀 자체를 삭제하기 때문에, 삭제 셀에 의해 다른 셀에 위치적 변화가 생깁니다. (새로운 형태로 정리하고자 할때 사용)

	A	B
1	Clear	Delete
2	Clear	Delete
3	Clear	Delete
4	Clear	Delete
5	Clear	Delete

```
Cells(1, 1).Clear
Cells(1, 2).Delete shift:=xlUp
```

	A	B
1		Delete
2	Clear	Delete
3	Clear	Delete
4	Clear	Delete
5	Clear	

한칸씩 위로 밀림

Copy 메서드 (개념)

셀을 복사-붙여넣기할 때 사용하는 메서드입니다.

사용 방법

기준셀. Copy Destination:=매개변수

생략 가능 보낼 위치(셀)

예시

- Range(Cells(1, 1),Cells(5, 1)).Copy Cells(6, 1)
- Cells(1, 5).Copy Destination:=Cells(1, 1)

변수를 알아보자

01. 변수는 무엇이고 왜 사용할까?

02. 변수의 데이터 형식 종류 (자료형, 개체형)

03. 변수를 선언하는 방법 + 활용하는 방법

04. 변수를 사용하여 곱하기만 되는 계산기를 만들어보기

수학에서의 X와 동일하다

X = 3으로 가정을 할때,

Y = 2X + 3 이라는 수식이 있다.

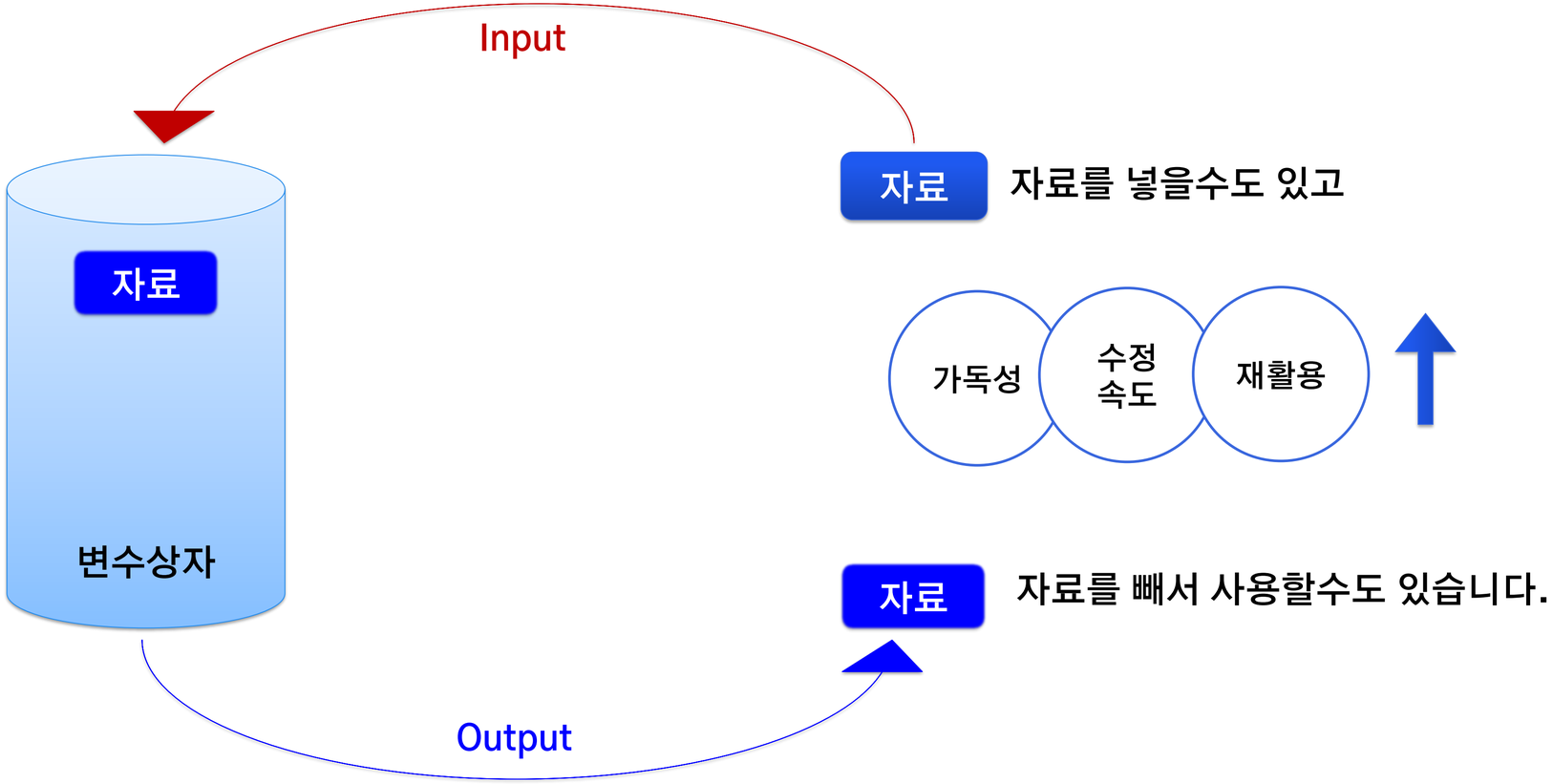
Y의 값은 ?

우리는 위의 수식을 풀때, X의 값을 3이라고 생각을 하고 풀죠?

프로그래밍에서의 변수도 X와 동일하다고 생각하시면 됩니다!

변수의 이해 (개념)

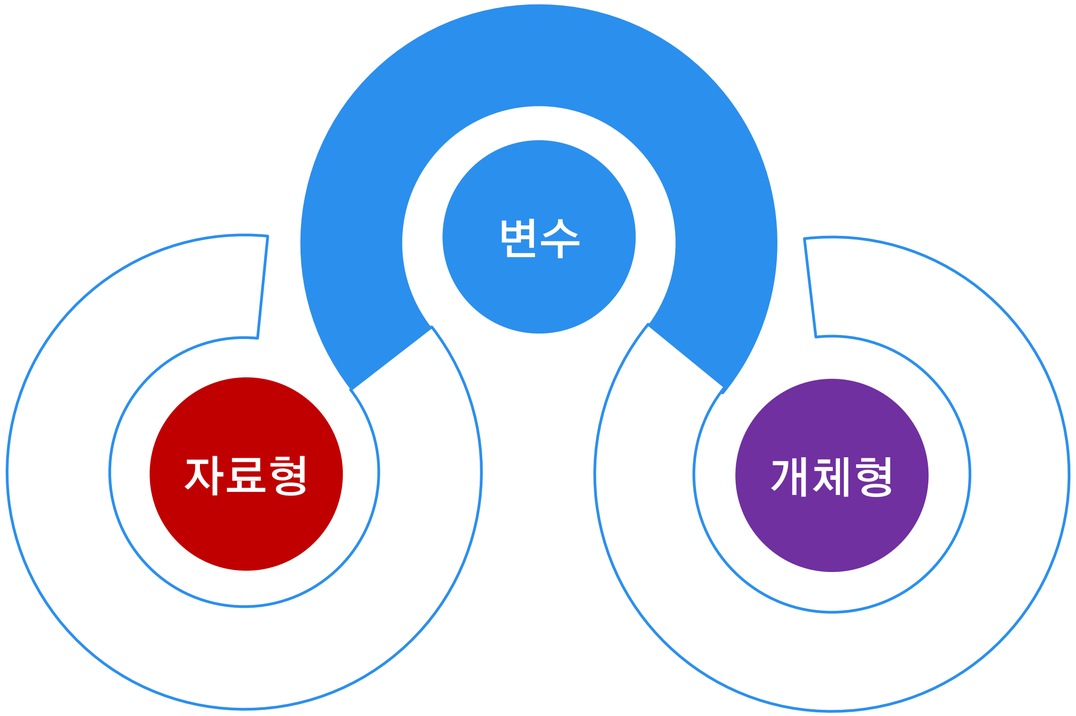
변수는 자료/개체를 저장 할 수 있는 가상의 데이터 그릇 입니다.



변수를 왜 사용할까?

작성코드 예시	설명
Sub 변수_미사용() Cells(1, 1).Value = 300 Cells(1, 1).Value = 300 * 3 Cells(1, 1).Value = 300 * 5 **End Sub**	**★ 변수 미사용코드의 문제점** ① 만약 300이 아닌, 200으로 변경하고 싶다면? 일일히 변경 필요 ② 개발 한달 뒤에 300이라는 숫자가 의미하는 바를 바로 알수있을까? 없음
Sub 변수_사용() Dim 학생점수 as Double 학생점수 = 300 Cells(1 ,1).Value = 학생점수 Cells(1, 1).Value = 학생점수 * 3 Cells(1, 1).Value = 학생점수 * 5 **End Sub**	**★ 변수 사용코드의 장점** ① 만약 300이 아닌, 200으로 변경하고 싶다면? 변수의 값만 변경 ② 학생점수라는 이름으로 선언했기때문에! 가독성 상승 ③ 프로그램이 끝날때까지 변수를 언제든지 사용가능! 재활용 가능

변수의 종류 (자료형 / 개체형)



자료형 변수

숫자, 텍스트 등
다양한 데이터 형식을 지정할 수 있습니다.

개체형 변수

자료값 이외의 엑셀 개체도 (Range, Sheet 등)
변수에 저장해서 사용할 수 있습니다.

왜 사용할까?

✓누적 되는 값을 처리해야 할 때
✓특정 자료를 나만의 별명으로 쉽게 관리 가능

변수의 종류 (자료형 / 개체형)

변수 구분	구분	데이터 형식	메모리 크기	값 범위	접두어	
자료형	숫자	Byte	1 byte	0 ~ 255	b	bt
		Integer	2 byte	-32,768 ~ 32,767	i	int
		Long	4 byte	-2,147,483,648 ~ 2,147,483,647	l	ilng
		Single	4 byte	약 -3.4E38 ~ 3.4E38	s	sng
		Double	8 byte	-1.8E308 ~ 4.9E324	d	dbl
		Currency	8 byte	-922,337,203,685,477.5808 ~ 922,337,203,685,477,5807	c	cur
		Decimal	14 byte	+-79,228,162,514,264,337,593,543,950,335	d	dec
	논리값	Boolean	2 byte	True, False	b	bl
	날짜/시간	Date	8 byte	100년 1월 1일 ~ 9999년 12월 31일	d	dt
	텍스트	String	10 byte 이상	가변 길이는 약 2조, 고정 길이는 65,400	s	str
	공통	Variant	16 byte 이상	숫자는 Double과 동일, 텍스트는 String과 동일	v	var
개체형	Cells / Range	Range	4 byte	-	rng	
	Worksheet	Worksheet	4 byte	-	ws	
	Workbook	Workbook	4 byte	-	wb	
	Chart	Chart	4 byte	-	c	cht

변수 선언 방법 (개념)

변수를 사용하려면 선언을 무조건 해야합니다.

선언 방법

Dim 변수이름 as 데이터 형식

예시

- Dim 학생점수 as Double → 학생점수라는 이름을 가진 Double형 그릇
- Dim 정수 as integer → 점수라는 이름을 가진 integer형 그릇
- Dim 학생이름 as string → 학생이름이라는 이름을 가진 String형 그릇

자료형 변수 (개념)

자료형 = 숫자 or 문자

작성코드 예시	설명
★ **변수 선언 방식 :** Dim 변수이름 As 데이터 형식	★ **변수의 특징**
Sub 변수선언_연습()	① 변수 선언은 프로시저 시작 직후 모두 선언!
Dim 정수 as integer	② 숫자 형식의 초기값(변수가 만들어진 직후)은 0 입니다.
Dim 합산 as Double, 실수 as Double	문자 형식의 초기값은 공백(“”) 입니다.
정수 = Cells(1 ,1).Value	③ A1셀의 값을 정수 변수에 할당한다.
실수 = 0.1323	④ 0.1323 값을 실수 변수에 할당한다.
합산 = 정수 + 실수 * 2	⑤ 정수 변수 + 실수 변수 * 2의 값을 합산 변수에 할당한다.
	※ 여기서 변수간 연산이 자유로움
Cells(2, 1).Value = 합산	⑥ A2셀의 값을 변수의 값으로 변경한다.
End Sub	

자료형 변수 (실습)

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

아래 문제를 풀어보세요!
문제 - 수학점수라는 이름을 가진 Double 형 변수에 80.5 할당 - 영어점수라는 이름을 가진 Double 형 변수에 70.4 할당 - F2셀에 수학점수 값 넣기 - F3셀에 영어점수 값 넣기 - F4셀에 수학점수 + 영어점수 합산 값 넣기 - F5셀에 수학점수, 영어점수의 평균 값 넣기 ※ 보통, 값들은 변수에 담아 사용! (코드가 길어질수록 변수 사용시 효과적인 코드 가능)

자료형 변수 (실습)

작성코드 예시	설명
★ **변수 선언 방식 :** Dim 변수이름 As 데이터 형식	
Sub 변수선언_실습()	
Dim 수학점수 as Double, 영어점수 as Double	① 수학점수, 영어점수 이름을 가진 Double 형 변수 선언
수학점수 = 80.5	② 수학점수 변수에 80.5 값 할당
영어점수 = 70.4	③ 영어점수 변수에 70.4 값 할당
Cells(2, 6).Value = 수학점수	④ F2셀의 값을 수학점수 변수의 값으로 변경
Cells(2, 6).Value = 영어점수	⑤ F3셀의 값을 영어점수 변수의 값으로 변경
Cells(2, 6).Value = 수학점수 + 영어점수	⑥ F4셀의 값을 수학점수 + 영어점수 변수의 합산으로 변경
End Sub	

개체형 변수 (개념)

작성코드 예시	설명
★ 개체형 할당 방식 : Set 변수이름 = 개체 (Range, Cells, sheets 등)	**★ 개체형 변수의 장점**
Sub 개체형_연습()	- 동일한 유형의 개체를 일일히 적을 필요 없이,
Dim 표1 as Range, 표2 as Range	- 할당한 변수로 간단히 표현 할 수 있으며,
	- 빠른 속도로 수정을 할 수 있습니다.
Set 표1 = Range(Cells(1, 1), Cells(100, 3))	① 표1 변수에 A1:C100 범위를 할당한다.
Set 표2 = Cells(1, 5)	② 표2 변수에 E1셀을 할당한다.
표1.Font.Color = RGB(255, 255, 0)	③ 표1 범위의 폰트 색상을 노랑색으로 변경한다.
표1.Value = “테스트”	④ 표1 범위의 값을 “테스트 ” 로 변경한다.
표2.Value = “테스트 중입니다 ”	⑤ 표2 범위의 값을 “테스트 중입니다 ” 로 변경한다.
End Sub	

개체형 변수 (실습)

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

아래 문제를 풀어보세요!
문제 - 범위라는 이름을 가진 Range 형 변수에 B1:F6 범위 할당 - 해당 범위에 **폰트크기를 15로** 변경 (Font.Size 속성) - 해당 범위에 **폰트를 굵게** 변경 (Font.Bold 속성) - 해당 범위에 **배경색을 파랑색으로** 변경 (Interior.Color 속성) - 해당 범위에 **값을 1로** 변경 (Value 속성)

개체형 변수 (실습)

작성코드 예시	설명
★ **개체형 할당 방식 :** Set 변수이름 = 개체	
Sub 개체형_실습()	
Dim 범위 as Range	① 범위 이름을 가진 Range 형 변수 선언
Set 범위 = Range(Cells(1, 2),Cells(6, 6))	② 범위 변수에 B1:F6 범위 할당
범위.Font.Size = 15	③ 해당 범위에 폰트크기를 15로 변경
범위.Font.Bold = True	④ 해당 범위에 폰트를 굵게 변경
범위.Interior.Color = vbBlue	⑤ 해당 범위에 배경색을 파랑색으로 변경
범위.Value = 1	⑥ 해당 범위에 값을 1로 변경
End Sub	

조건문을 알아보자

01. If ~ End If 조건문 (가장 Basic)

02. If ~ Else ~ End If 조건문 (예외 상황이 있을때)

03. If ~ Else If ~ Else ~ End If 조건문 (다중 조건이 있을때)

04. And, Or 문 활용하기

If ~ End If 조건문 (개념)

작성코드 예시	설명
★ **IF ~ End If (단일 조건문)** → 가장 Basic 한 조건문	

Sub If문_단일조건()
 	
If 조건 Then
[True Code]
End If
 	
If ~ End If는 한쌍
 	
※ 조건은?
대상 vs 대상 : 등호/부동호로 비교 되는것!
Ex : Cells(1, 1).Value = 5
End Sub
	① 해당 조건이 참이면,
	② True Code에 입력 되어있는 명령문 실행
	③ If문은 End If로 필수적으로 마무리 해줘야 합니다.

If ~ End If 조건문 (실습)

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

아래 문제를 풀어보세요!
문제 - 조건 : A1셀의 값이 5 이상 (>=)이라면, - 참 : A2셀의 값을 "참입니다 " 로 변경 A2셀의 폰트의 색상을 파랑색으로 변경

If ~ End If 조건문 (실습)

작성코드 예시	설명

Sub 조건문_단일()
 	
If Cells(1, 1).Value >= 5 Then
Cells(2, 1).Value = "참입니다"
Cells(2, 1).Font.Color = vbBlue
End If
 	
End Sub
	① A1셀의 값이 5이상이면,
	② A2셀의 값을 "참입니다 " 로 변경
	③ A2셀의 배경색을 파랑색으로 변경

If ~ Else ~ End If 조건문 (개념)

작성코드 예시	설명
★ IF ~ Else ~ End If → 한 조건으로 2가지 판단을 하고자 할때 사용	
Sub If문_단일조건()	

If 조건 Then
[True Code]
Else → Else는 선택적으로 사용 가능!
[False Code]
End If
	① 해당 조건이 참이면,
	② True Code에 입력 되어있는 명령문 실행
	③ 해당 조건이 거짓이면,
	④ False Code에 입력 되어있는 명령문 실행

End Sub	

If ~ Else ~ End If 조건문 (실습)

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

아래 문제를 풀어보세요!

문제

	A	B	C	D
1	1	3		
2				
3				
4				
5				

- 조건 : A1셀의 값 + B1셀의 값이 4이상 (>=) 이라면,
- 참 : C1셀의 **값**을 "참입니다" 로 변경
  C1셀의 **배경색**을 "파랑색" 으로 변경
- 거짓 : C1셀의 **값**을 "거짓입니다 " 로 변경
  C1셀의 **배경색**을 "빨강색 " 으로 변경 -> Cells(1, 3).Interior.Color = vbRed

If ~ Else ~ End If 조건문 (실습)

작성코드 예시	설명

Sub 조건문_다중()
 	
If Cells(1 ,1).Value + Cells(1, 2).Value >= 5 Then
Cells(1, 3).Value = “참입니다”
Cells(1, 3).Interior.Color = RGB(0, 0, 255)
Else
Cells(1, 3).Value = “거짓입니다”
Cells(1, 3).Interior.Color = RGB(255, 0, 0)
End IF
End Sub
	① A1셀의 값 + B1셀의 값이 5이상이면,
	② C1셀의 값을 “참입니다 ” 로 변경
	③ C1셀의 배경색을 파랑색 으로 변경
	그게 아니라면 ----------
	④ C1셀의 값을 “거짓입니다 ” 로 변경
	⑤ C1셀의 배경색을 빨강색 으로 변경

If ~ ElseIf ~ End If 조건문 (개념)

작성코드 예시	설명
★ 2개 이상의 조건에 따라 명령을 제어 할 때 사용	
Sub If문_다중조건()	
If 조건1 Then	- Elseif와 Else는 선택적으로 사용 가능합니다.
[조건1이 참일 경우 명령문 실행]	- Elseif는 개수의 제한이 없습니다 / Else는 1개만 사용이 가능합니다.
Elseif 조건2 Then	- Else는 If문의 가장 아래에 있어야 합니다. (예외 상황일때 사용)
[조건2가 참일 경우 명령문 실행]	
Else	
[위 조건이 모두 거짓일 경우 명령문 실행]	★ If / Elseif는 조건과 Then이 필수적으로 붙는다.
End If	Else는 조건이 없고 Then이 없다.
End Sub	

If ~ ElseIf ~ End If 조건문 (실습)

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

아래 문제를 풀어보세요!

문제

조건	명령문
A1셀의 값 = 1	① B1:B10 값 = “초록사과“ ② B1:B10 배경색 = “초록색 ”
A1셀의 값 = 2	① B1:B10 값 = “딸기“ ② B1:B10 배경색 = “빨강색 ”
그 외 (Else)	① B1:B10 값 = “““ ② B1:B10 배경색 = “흰색 ”

기준셀.Interior.Color = RGB(?, ?, ?)

If ~ ElseIf ~ End If 조건문 (실습)

작성코드 예시	설명
Sub 조건문_다중()	
If Cells(1 ,1).Value = 1 Then	① A1셀의 값이 1이면,
Range(Cells(1, 2),Cells(10, 2)).Value = “초록사과”	- B1:B10 범위의 값을 “초록사과 ” 로 변경
Range(Cells(1, 2),Cells(10, 2)).Interior.Color = vbGreen	- B1:B10 범위의 배경색을 초록색 으로 변경
ElseIf Cells(1 ,1).Value = 2 Then	② A1셀의 값이 2이면,
Range(Cells(1, 2),Cells(10, 2)).Value = “딸기”	- B1:B10 범위의 값을 “딸기 ” 로 변경
Range(Cells(1, 2),Cells(10, 2)).Interior.Color = vbRed	- B1:B10 범위의 배경색을 빨강색 으로 변경
Else	③ 그외라면,
Range(Cells(1, 2),Cells(10, 2)).Value = “”	- B1:B10 범위의 값을 “ ” 로 변경
Range(Cells(1, 2),Cells(10, 2)).Interior.Color = vbWhite	- B1:B10 범위의 배경색을 흰색 으로 변경
End If	
End Sub	

And / Or 문 (개념)

작성코드 예시	설명
★ And / Or → 조건을 한번에 여러가지 Control 할수 있음	
Sub If문_다중조건()	
If 조건1 And 조건2 Then	- 조건은? 비교대상 vs 비교대상
[조건1과 조건2가 참일 경우 명령문 실행]	- And와 Or의 개수 제한은 없습니다.
Elseif 조건2 Or 조건3 Then	
[조건2과 조건3중 하나라도 참일 경우 명령문 실행]	아래의 코드는 오류입니다.
End If	~~If Cells(1, 1).Value = 1 Or 2 Then~~
	아래의 코드가 정답입니다.
	If Cells(1, 1).Value = 1 Or Cells(1, 1).Value = 2 Then
End Sub	

숫자인지? 에러인지? (개념)

조건문에서는 숫자인지, 에러인지 찾아주는 함수를 사용할 수 있습니다.

숫자인지? IsNumeric

```
If IsNumeric(Cells(1, 1).Value) = True Then
```

날짜인지? IsDate

```
If IsDate(Cells(1, 1).Value) = False Then
```

에러인지? IsError

```
If IsError(Cells(1, 1).Value) = False Then
```

반복문을 알아보자

01. 반복문을 왜 사용할까?

02. For Next 반복문

03. For Each Next 반복문

04. 내가 가지고 있는 행의 개수 카운트하기

반복문을 왜 사용할까?

규칙적인 명령어를 1000줄 이상 써야한다면? → 코딩 노가다로 이어집니다.

노가다 코딩

```
Cells(1, 1).Value = "시리얼1"
Cells(2, 1).Value = "시리얼2"
Cells(3, 1).Value = "시리얼3"
Cells(4, 1).Value = "시리얼4"
Cells(5, 1).Value = "시리얼5"
⋮
Cells(1000, 1).Value = "시리얼1000"
```

효율적 코딩

```
For 변수 = 1 to 1000
    Cells(변수, 1).Value = "시리얼" & 변수
Next 변수
```

For Next 반복문 (개념)

작성코드 예시	설명
★ 반복문 형태 반드시 기억할것!	★ 반복문의 특징
Sub 반복문_개념() Dim 변수 As Double For 변수 = 1 To 1000 Step 1 [실행할 명령문들] Next 변수 1000번 반복 ※ 보통 반복문에 사용하는 변수는 i, i2, i3 등 사용 (의미 없는 변수기 때문에 짧은게 좋음) End Sub	① 반복문에 사용하는 변수는 무조건 "숫자형" 변수로 선언! ② 카운터 변수가 1~1000까지 1씩 증가할때마다, ③ 명령문들이 한번씩 실행됩니다. ④ Next는 카운터 변수에 Step 값 만큼 증가시켜 위로 올려보냅니다. ※ Step은 생략 가능하며, 생략시 Default값은 1씩 증가 ※ 변수를 사용하여 원하고자 하는 로직을 구성하여야 합니다.

For Next 반복문 (실습 ①)

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

아래 문제를 풀어보세요!
문제 - A1:A1000 범위 셀에 순차적으로 1~1000까지의 값을 순차적으로 입력해라 - 예시 참고 A1 셀의 값 = 1 A2 셀의 값 = 2 A3 셀의 값 = 3 A4 셀의 값 = 4 : A1000 셀의 값 = 1000

For Next 반복문 (실습 ①)

작성코드 예시	설명
Sub 반복문_실습1()	
Dim 변수 as Double	① 반복문에 필요한 "숫자형 " 변수 선언
For 변수 = 1 to 1000 ~~Step 1~~	② 변수의 값이 1부터 1000까지 1씩 증가
Cells(변수, 1).Value = 변수	③ A1셀부터 순차적으로 변수의 값을 입력
Next 변수	④ 변수의 값이 1000이 될때까지 ②번 항목 반복
※ 보통 Step은 생략 함 (Default : 1씩 증가)	
End Sub	

For Next 반복문 (실습 ②)

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

아래 문제를 풀어보세요!

문제

– 아래의 데이터 베이스에서 제시한 조건 별로 데이터를 처리해라.

	A	B
1	0	
2	1	
3	5	
4	23	
5	15	
6	23	
7	13	
8	1	
9	5	
10	161	
11	0	
12	34	
13	235	
14	1	
15	23	

A열의 값이?	B열의 값을
0 일때	“품절”
1 이상 5 미만	“품절 임박”
5 이상	” 재고 ”

For Next 반복문 (실습 ②)

작성코드 예시	설명

Sub 반복문_실습2()
Dim i as Double
For i = 1 to 15
If Cells(i, 1).Value = 0 Then
Cells(i, 2).Value = “품절”
Elseif Cells(i, 1).Value >= 1 And Cells(i, 1).Value < 5 Then
Cells(i, 2).Value = “품절임박”
Else
Cells(i, 2).Value = “재고”
End If
Next i
End Sub
	① 반복문에 필요한 “숫자형 ” 변수 선언
	② 변수의 값이 1부터 15까지 1씩 증가
	③ A열의 값이 0이면
	→ B열에 “품절 ” 입력
	④ A열의 값이 1이상 / 5 미만이면
	→ B열에 “품절임박 ” 입력
	⑤ A열의 값이 5이상 이면
	→ B열에 “품절 ” 입력

For Next 반복문 – 감소문 (개념)

작성코드 예시	설명
★ 반복문 형태 반드시 기억할것!	★ 감소문을 사용하는 곳
Sub 반복문_개념() Dim 변수 As Double For 변수 = 1000 To 1 Step -1 [실행할 명령문들] Next 변수 ※ Step은 무조건 -1로 두어야 합니다. End Sub	반복문 + Delete 메서드를 사용해서 조건에 맞는 **범위/시트**를 삭제 할 때, → Delete 문의 특성상 아래에서 위로 반복해야 합니다.

End 속성 (개념)

데이터의 행/열을 카운트할 때 사용하는 속성입니다.

사용 방법

기준셀. End(매개변수).Row

기준셀. End(매개변수).Column

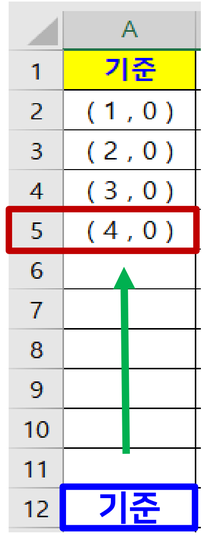
xlToLeft (왼쪽으로)

xlUp (위로)

xlToRight (오른쪽으로)

xlDown (아래로)

예시



5라는 행번호를 어떻게 알 수 있을까?

Cells(12, 1).End(xlUp).Row

기준에서

Ctrl + Up

행 번호

실전예시 ①

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

문제	품목 별 가격대를 아래의 조건에 맞게 입력하세요.
풀이 조건	조건1) 당일 조사가격이 - 100,000 이상이면 ☞ 가격대는 "상" - 50,000 이상이면 ☞ 가격대는 "중" - 50,000 미만이면 ☞ 가격대는 "하" 조건2) H5 셀의 품목과 일치하는 데이터의 행 (A열~E열)을 H5 셀의 배경색, 글꼴색과 동일하게 변경하세요.
결과	

	A	B	C	D	E
1	조사일자	표준품목명	당일조사가격	표준시장명	가격대
2	2018-03-01	한우	10200	일로가축시장	
3	2018-03-01	한우	9900	영광가축시장	
4	2018-03-01	한우	3030000	일로가축시장	
5	2018-03-01	한우	3450000	일로가축시장	

시작 버튼

품목명	한우

실전예시 ②

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

문제	데이터 가공 문제

풀이 조건	
	조건1) 수량이 2개 이상이면 해당 행은 삭제 (Delete) 조건2) 판매가 – (판매가*할인율) ☞ 할인금액 (E열에) 넣기 조건3) 판매가가 100,000 이상이면서 / 쇼핑몰명이 “GS shop” 인 경우 ☞ 해당 행의 배경색을 파란색으로, 폰트색을 흰색으로 변경

결과

	A	B	C	D	E	F	G
1	상품명	수량	판매가	할인율 (%)	할인금액	쇼핑몰명	주문구분
2	[삼성]원형접시 15cm 2P 일리아나 세트	1	66,000	15%		CJOshopping (신)	주문(진행)
3	삼성 자이언트 카푸치노 머그 2P 세트	1	80,000	15%		CJOshopping (신)	주문(진행)
4	삼성 플라워찬합-색상선택	1	99,000	20%		CJOshopping (신)	주문(진행)

H I J

시작 버튼

실전예시 ③

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

문제	선택한 범위의 거래량만 제외하고 모두 삭제하기

풀이 조건	1) 거래량 설정 범위를 “제외하고” 모두 삭제 (Delete 메서드 사용) 2) 프로그램 코드 가장 상단에 “O1:P247” 범위에 있는 데이터를 “A1”셀에 붙여넣기

결과

	A	B
1	년/월/일	거래량(주)
2	2016/12/29	150,329
3	2016/12/28	133,258
4	2016/12/27	93,069
5	2016/12/26	96,472
6	2016/12/23	166,697
7	2016/12/22	127,092
8	2016/12/21	131,042
9	2016/12/20	152,304

거래량	
0 이상	999,999 이하

Run Macro

위에서 설정한 범위를 제외한 나머지를 삭제하세요.

1) O1:P247 범위를 Copy해서 A1셀에 붙여넣기
2) 선택한 범위를 제외한 "나머지"를 Delete 하기

시트 개체에 접근해보기

01. Sheets 개체에 대해서 알아보기

02. Sheets 개체에 자주 사용하는 속성/메서드 알아보기

03. Sheet간 작업을 처리하는 팁!

04. 시트간 Vlookup과 같은 기능을 VBA로 구현해보기

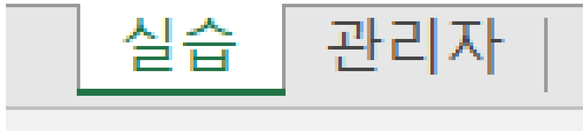
Sheets 개체 (개념)

Sheets 개체는 이렇게 사용합니다.

표현 방법

실제 시트이름 사용	Sheets("시트명")
페이지 번호 사용	Sheets(1)
활성화 시트 사용	ActiveSheet

페이지 번호란?
가장 좌측이 1번



1 Page 2 Page

사용 방법

Sheets("시트명").메서드 / Sheets("시트명").속성 = 변경 값

Sheets 개체 생략

Cells, Range 개체 앞에는 Sheets 개체를 쓸 수도 있고, 생략도 가능합니다.

생략O / 현재 활성화 된 시트 기준

```
Cells(1, 1).Value = 5
Range(Cells(1 ,1), Cells(100, 3).Value = 5
```

장점 코드가 단순해 가독성이 ▲

단점

생략X / 특정 시트 기준

```
Sheets("결과").Cells(1, 1).Value = 5
Sheets("관리자").Cells(1, 5).Font.Size = 15
```

장점 화면 전환 없이 속성 변경 가능

단점 코드가 길어져 가독성 ▼

Activate 메서드 (개념)

개체를 활성화할 때 사용하는 메서드입니다. (보통 Worksheet 개체에서 많이 사용합니다 – 시트간 이동)

사용 방법

개체.Activate (모든 개체에서 사용 가능)

- Cells / Range
- Worksheets / Workbooks

예시

- Sheets("관리자").Activate
- Sheets(2).Activate
- Cells(1, 1).Activate

Copy 메서드 (개념)

시트를 복사할 때 사용하는 메서드입니다.

사용 방법

Sheets("견적서 양식").Copy After:=Sheets(1)

견적서 양식시트를 복사해서 첫 번째 시트 뒤로 이동

※ After를 생략하면, 통합문서 형태로 복사

Add 메서드 (개념)

시트를 추가할 때 사용하는 메서드입니다.

사용 방법

Sheets.Add (After:=Sheets(1)).Name = “변경 이름 ”

시트를 추가해서 (첫번째 시트 뒤에 생성)

이름을 바로 변경

Delete 메서드 (개념)

시트를 삭제할 때 사용하는 메서드입니다.

사용 방법

```
Application.DisplayAlerts = False
Sheets("관리자").Delete
```

관리자 시트를 삭제

Name 속성 (개념)

시트의 이름을 지정 할 때 사용하는 메서드입니다.

사용 방법

Sheets("관리자").Name = "회원 " (관리자 시트의 이름을 회원으로 변경)

예시

- Sheets(1).Name = "Main"
- Sheets("Sub").Name = "변환_Sub"

Sheets 개체 (실습)

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

아래 문제를 풀어보세요!
문제 ① 관리자, 결과 시트를 만든다. 관리자 결과 ② 관리자, 결과 시트에 동일하게 아래의 명령이 실행되도록 한다. - A1:A10 범위의 값에 순차적으로 1부터 10넣기 - 만약 넣을 값이 5이상이면 해당 셀의 배경색을 빨강색으로 변경

Sheets 개체 (실습)

Case1

```
Sub Case1()

    Dim i As Double

    For i = 1 To 10
        Sheets("관리자").Cells(i, 1).Value = i
        Sheets("결과").Cells(i, 1).Interior.Color = i

        If i >= 5 Then
            Sheets("관리자").Cells(i, 1).Interior.Color = vbRed
            Sheets("결과").Cells(i, 1).Interior.Color = vbRed
        End If
    Next i

End Sub
```

속도 : 1위 / 효율 : 2위

실 업무에서는 같은 명령을
여러시트에 적용하는 경우는 많이 없음!

★ Case 2 ★

```
Sub Case2()

    Dim i As Double

    Sheets("관리자").Activate
    For i = 1 To 10
        Cells(i, 1).Value = i

        If i >= 5 Then
            Cells(i, 1).Interior.Color = vbRed
        End If
    Next i

    Sheets("결과").Activate
    For i = 1 To 10
        Cells(i, 1).Value = i

        If i >= 5 Then
            Cells(i, 1).Interior.Color = vbRed
        End If
    Next i

End Sub
```

속도 : 2위 / 효율 : 1위

시트간 구분이 명확하기 때문에,
보통 위와 같이 코딩을 하는것이 가장 Best

Case 3

```
Sub Case3()

    Dim i As Double


    For i = 1 To 10
        Sheets("관리자").Activate
        Cells(i, 1).Value = i

        If i >= 5 Then
            Cells(i, 1).Interior.Color = vbRed
        End If

        Sheets("결과").Activate
        Cells(i, 1).Value = i

        If i >= 5 Then
            Cells(i, 1).Interior.Color = vbRed
        End If
    Next i

End Sub
```

속도 : 3위 / 효율 : 3위

메서드는 속성을 변경하는것보다 시간이 많이 소요
따라서, 위와 같은 코드는 가장 Worst

With문 (개체 생략)

With 문에서는 개체를 .으로 생략 할 수 있다.

사용 방법

```
With Cells(1, 1)

    .Value = 5
    .Font.Size = 15
    .Font.Bold = True
    .Interior.Color = vbBlue

End With
```

보통 시트간 구분을 할 때 사용!

```
Sub Case2()

    Dim i As Double

    With Sheets("관리자")
        .Activate

        For i = 1 To 10
            Cells(i, 1).Value
        Next i
    End With
---------------------------------------------------------
    With Sheets("결과")
        .Activate

        For i = 21 To 30
            Cells(i, 1).Value
        Next i
    End With

End Sub
```

실전예시 ④

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

문제	설정한 업체에 해당하는 정보를 가져와라 (Vlookup 구현)

풀이 조건	조건) 설정 옵션에 해당하는 업체명을 “업체정보” 시트에서 찾아, 아래의 빈칸에 채워 넣어라.

결과

설정 옵션	
업체명	삼성전자

정보 가져오기

업체명	담당자	업태	종목	전화번호	비고

실전예시 ⑤

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

문제	설정한 업체에 해당하는 정보를 가져와라 (Vlookup 구현)
풀이 조건	조건) 리스트에 있는 업체명을 “업체정보” 시트에서 찾아, 아래의 빈칸에 채워 넣어라.
결과	

업체명	담당자	업태	종목	전화번호	비고
삼성전자					
LG전자					
넥슨					
네이버					
현대자동차					
러닝스푼즈					
유성전자					
카카오					
조아애드					

프로그램을 구성하는 Tip !

ⓐ 시트 구성 = 관리자 시트 (설정 값) + 입력 시트 (Raw) + 결과 시트

Ⓑ 로직 구성 (시트간 구분은 With문으로!)

① 관리자 시트로 이동 (이동을 하고, 설정 값을 변수에 담는다)

② 기존 결과값이 있는 시트로 이동 (이동을 하고, 기존의 결과값을 삭제한다)

③ 설정 값과 일치하는 데이터를 결과 시트로 이동

(a) 누적되는 리스트 형태 (규칙적으로 누적)

(b) 정해진 위치에 넣는 형태 (정해진 위치에 넣기)

④ 결과 시트로 이동 (결과물을 바로 볼 수 있도록!)

총 정리 ①

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

문제	로우데이터를 활용하여 업체별 견적서 양식을 만들어보자.
풀이 조건	1. 반복문, 조건문 사용 2. 선택한 날짜/업체명을 이용하여 해당 설정값과 동일한 데이터를 견적서에 리스팅해라.
결과	

실전예시 ⑥

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

문제	데이터 전 처리 과정 실습 (나만의 양식으로 변경)
풀이 조건	"데이터" 시트에서 "결과" 시트로 Raw데이터 형식으로 변환시키기 (데이터 전 처리 과정으로, 실제 가공로직을 하기 전 해주면 더 수월하게 로직 작성 가능)

결과

No.	상품코드	상품명/옵션/BARCODE	매입유형/면세여부	물류센터	발주수량	업체납품가능수량	입고수량	팔렛트수량	공급가		
									매입가	공급가액	부가세
1	8385746	플라워 하드케이스 핑크 그라데이션 갤럭시S5 하드케이스	직매입	인천4	2	2	0	1	4500	4091	409
		R001880260001	과세								

	A	B	C	D		F	G	H
1	공통항목					공급가		
2	발주날짜	발주번호	상품코드	상품명	발주수량	매입가	공급가액	부가세
3	2018-08-01	4273356	8385746	플라워 하드케이:	2	4500	4091	409
4	2018-08-01	4273356	8385747	플라워 하드케이:	2	4500	4091	409
5	2018-08-01	4273356	8385748	플라워 하드케이:	2	4500	4091	409
6	2018-08-01	4273356	8385749	플라워 하드케이:	2	4500	4091	409
7	2018-08-01	4273356	8385750	플라워 하드케이:	2	4500	4091	409

모든 시트를 순환하기 (코드)

모든 Sheets를 순환하는 코드는 이렇게 사용합니다.

표현 방법

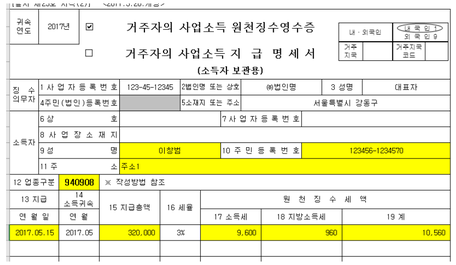
```
Sub Sheet_Loop()

    Dim i As Double

    For i = 1 To Sheets.Count
        Sheets(i).Activate
    Next i

End Sub
```

전체 시트의 갯수 만큼 반복

페이지 번호를 사용하여 표현

총 정리 ②

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

문제	로우데이터를 활용하여 사람별 사업조서 양식을 만들어보자.
풀이 조건	1. 로우데이터를 활용하여서 프린트 유무가 “O” 인것만 시트 만들기 2. 시트의 이름은 “(결과) 유영현_20181117” 과 같은 형태로 만들기
결과	 사람별 사업조서 양식 생성 (결과) 서정숙_20181117 | (결과) 이창범_20181117 시트 이름의 규칙은 위와 같음 “Date” 함수 활용하여서 오늘 날짜 표현

거주자의 사업소득 원천징수영수증
거주자의 사업소득 지 급 명 세 서
(소득자 보관용)

귀속연도	2017년	1 사 업 자 등 록 번 호	123-45-12345	2법인명 또는 상호	(주)법인명	3 성명	대표자
징수의무자		4주민(법인)등록번호		5소재지 또는 주소	서울특별시 강동구		
소득자		6 상 호		7 사 업 자 등 록 번 호			
		8 사 업 장 소 재 지					
		9 성 명	이창범	10 주 민 등 록 번 호	123456-1234570		
		11 주 소	주소1				
12 업종구분	940908	× 작성방법 참조					
13 지급 연월일	14 소득귀속 연월	15 지급총액	16 세율	원천징수세액 17 소득세	18 지방소득세	19 계	
2017.05.15	2017.05	320,000	3%	9,600	960	10,560	

(결과) 서정숙_20181117 | (결과) 이창범_20181117

Instr 함수 (개념)

문자열에서 찾고자하는 키워드의 포함의 유무를 알고자 할때 사용하는 함수입니다.

사용 방법

해석 : If 4 > 0 Then

```
Sub Keyword_Search()

    If InStr("안녕하세요", "세") > 0 Then
        MsgBox "세 라는 키워드는 포함되어 있습니다"
    Else
        MsgBox "세 라는 키워드는 포함되어 있지 않습니다"
    End If

End Sub
```

★ Instr("안녕하세요","세") → 4를 반환

안	녕	하	세	요

전체 문자 중 찾는 문자의 위치 반환

보통, 반환 값이 1이상이면 포함한다라고 생각

실전예시 ⑦

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

문제	Instr 함수를 이용하여 키워드가 포함된 셀을 Sorting 해라
풀이 조건	1) 데이터를 있는 만큼 반복해라 2) 키워드가 포함이면 -> 배경색을 “빨강 ” 미포함이면 -> 배경색을 “흰색”
결과	키워드 | 세부여행사 Run Macro → 키워드가 포함이면 배경색을 “빨강 ” 으로 변경 포함되지 않으면 배경색을 “흰색 ” 으로 변경 키워드 입력

총 정리 ③

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

문제	로우데이터를 활용하여 “키워드” 가 포함된 리스트를 만들고, 차트를 동적으로 그려보자
풀이 조건	1. 키워드가 **포함**된 데이터를 리스팅해라 2. 이름관리자를 사용하여 차트의 데이터계열을 “Dynamic Range” 로 구현을 해라 (차트동적 그리기)

결과

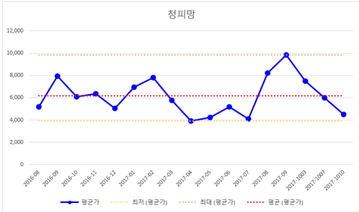
키워드	청피망

Run Macro

옵션 설정 후 버튼 클릭

날짜	품목명	반입금액	평균가	최저 (평균가)	최대 (평균가)	평균 (평균가)
2016-08	청피망	242,500	5,160	3,900	9,866	6,158
2016-09	청피망	350,100	7,957	3,900	9,866	6,158
2016-10	청피망	249,500	6,085	3,900	9,866	6,158
2016-11	청피망	248,600	6,374	3,900	9,866	6,158
2016-12	청피망	258,000	5,059	3,900	9,866	6,158
2017-01	청피망	270,600	6,938	3,900	9,866	6,158
2017-02	청피망	305,100	7,823	3,900	9,866	6,158
2017-03	청피망	208,500	5,792	3,900	9,866	6,158
2017-04	청피망	175,500	3,900	3,900	9,866	6,158
2017-05	청피망	204,000	4,250	3,900	9,866	6,158
2017-06	청피망	264,000	5,176	3,900	9,866	6,158
2017-07	청피망	212,500	4,087	3,900	9,866	6,158
2017-08	청피망	460,000	8,214	3,900	9,866	6,158
2017-09	청피망	493,300	9,866	3,900	9,866	6,158
2017-1003	청피망	37,500	7,500	3,900	9,866	6,158
2017-1007	청피망	30,000	6,000	3,900	9,866	6,158
2017-1010	청피망	9,000	4,500	3,900	9,866	6,158

결과물 시트에 리스팅



차트동적 그리기

Unmerge 메서드 (개념)

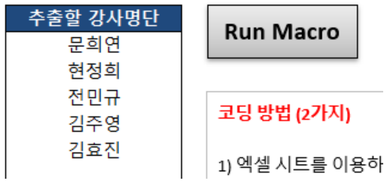
셀의 병합을 해제할 때 사용하는 메서드입니다.

사용 방법

Cells.Unmerge 모든 셀의 병합을 해제

총 정리 ④

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

문제	로우데이터를 활용하여 강사별 급여 정산서를 만드는 프로그램을 구현해라
풀이 조건	1. 추출할 강사명단에 **이름** 을 리스팅해라 2. 강사별 급여 정산서를 만들어라 (수동으로 강사명단 리스팅, 컬렉션을 이용해서 명단 리스팅)
결과	 추출할 강사명단 문회연 현정희 전민규 김주영 김효진 Run Macro 코딩 방법 (2가지) 1) 엑셀 시트를 이용하 강사 리스팅 후 버튼 클릭 강사별 양식 생성

AdvancedFilter 메서드 (개념)

데이터 셋을 편하게 필터할 때 사용하는 메서드입니다. (고급필터와 동일)

사용 방법

데이터 셋의 범위.AdvancedFilter Action:=xLFilterCopy, CriteriaRange:= 조건 범위, CopyToRange:=복사할 위치의 제목 범위

거 래 처 별 거 래 원 장

CriteriaRange

거래처	거래일자	거래일자	연락처	공급가액	세액	입고금합계	지급액합계	미지급금합계
돌핀종합	>=43374	<=43404						
돌핀종합	2018-10-01	2018-10-31	031-973-3072	90000	8181.818182	90,000	0	0

돌핀종합 (2018년 10월 1일부터 2018년 10월 31일까지) 거래내역 검색하기

CopyToRange

거래일자	거래내역	단가	수량	공급가액	세액	입고액	지급액	미지급금
2018-10-05	세제	33,000	1.0	33,000	3,000	33,000		
2018-10-05	린스	57,000	1.0	57,000	5,182	57,000		

실전예시 ⑧

지금까지 배운 내용으로 아래의 결과물이 나올 수 있도록 해봅니다.

문제	간단한 조회프로그램 만들기

풀이 조건	1) 고급필터 (AdvancedFilter)를 이용하여 원하는 데이터셋을 가져올수 있도록한다. ※ 조건의 개수가 유동적으로 변해야하는 프로그램 같은 경우에는 고급필터를 사용하면 쉽게 가능

결과

거 래 처 별 거 래 원 장

거래처	거래일자	거래일자	연락처	공급가액	세액	입고금합계	지급액합계	미지급금합계
돌핀종합	>=43374	<=43404						
돌핀종합	2018-10-01	2018-10-31	031-973-3072	90000	8181.818182	90,000	0	0

돌핀종합 (2018년 10월 1일부터 2018년 10월 31일까지) 거래내역 검색하기

거래일자	거래내역	단가	수량	공급가액	세액	입고액	지급액	미지급금
2018-10-05	세제	33,000	1.0	33,000	3,000	33,000		
2018-10-05	린스	57,000	1.0	57,000	5,182	57,000		

원하는 조건 입력시에,

조건에 맞는 데이터 셋이 검색 되도록

워크북 개체에 접근해보기

01. Workbooks 개체에 대해서 알아보기

02. Workbooks 개체에 자주 사용하는 속성/메서드 알아보기

03. 동일한 유형의 여러가지 파일을 통합하는 방법

04. 팁 공유!

Workbooks 개체 (개념)

Workbooks 개체는 이렇게 사용합니다.

표현 방법

매크로 보유 파일	ThisWorkbook
활성화 파일	ActiveWorkbook

사용 방법

Thisworkbook.메서드

Open 메서드 (개념)

문서를 오픈할 때 사용하는 메서드입니다.

사용 방법

Workbooks.Open "파일주소"

파일주소란? 전체 디렉토리를 뜻함
Ex) C:\Users\skycoms\Desktop\온라인 실습파일\Part 1\Test.xlsx

Close 메서드 (개념)

문서를 닫을때 사용하는 메서드입니다.

사용 방법

Activeworkbook.Close Savechanges:=False

위 인수를 지정해줘야, 통합문서를 닫을때 경고메세지(저장 유무) 가 뜨지 않음