



오빠두엑셀 기초강좌 E-BOOK

1

Free Tutorial

2

Business Excel

3

Basic to Expert

한국어

#엑셀 VBA1-6

>> VBA 개체

Property

(속성)

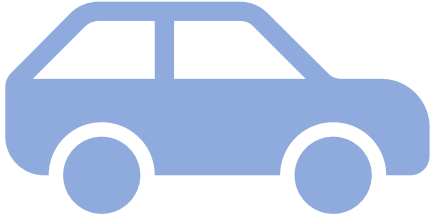


1. VBA 에서 Property(속성) 이란 무엇인가?

이전강의에서 VBA 에서 개체(Object)를 자동차에 비유하였습니다. 이번에도 개체를 자동차에 비유하여 Property(속성)에 대해 알아보겠습니다.

여기 자동차 1 대가 놓여있습니다.

이 자동차의 속성은 무엇이 있을까요? 정말 다양하겠지만 아래 몇 가지만 나열하겠습니다.



- | | |
|--------|--------------|
| 1. 제조사 | (Brand) |
| 2. 모델 | (Model) |
| 3. 크기 | (Size) |
| 4. 연비 | (Efficiency) |
| 5. 엔진 | (Engine) ... |

보시는 바와 같이, 자동차의 속성은 해당 자동차의 특징을 나타내는 각 항목으로 구성되어 있으며 VBA 에서 사용하는 Property 도 이와 같습니다.

1-A. VBA 의 Property 는 Object 의 계층에 따라 작성됩니다.

아래의 코드를 보겠습니다.

```
MyCar.Model = "520d"
```

'// 자동차를 잘 아는 사람이라면 이 모델이 'BMW' 브랜드 제품이라는 것을 알겠지만, 코드를 작성할 때에는 각 속성을 정확히 명시하기 위해 아래와 같이 작성하게 될 것입니다.

```
MyCar.Brand("BMW").Model = "520d"
```

'// 이렇게 작성을 하게되면, 하위그룹인 모델을 지정하기 위해 상위그룹인 제조사부터 정하고 내려오면서 더욱 명확하게 속성을 이해 할 수 있게 됩니다.

1-B. Property 는 세부항목을 가질 수 있습니다.

아래의 코드를 보겠습니다.

```
MyCar.Color = vbRed
```

'// 너무 광범위한 속성입니다. 차의 외부색상인지, 내부색상인지, 전체를 빨간색으로 할 것인지, 무늬만 빨간색으로 할 것인지 명시되지 않았습니니다. 따라서 실제 코드에서는 아래와 같이 작성하게 될 것입니다.

```
MyCar.Outerior.Color = vbRed
```

```
MyCar.Interior.Color = vbBlack
```

'// 외부색상은 검정색, 내부색상은 빨간색으로 이전보다 더욱 명확하게 속성을 지정하였습니다. 실제 코드를 작성할 때에는 이보다 더 세분화된 내용으로 작성하게 될 것입니다.

2. Property 는 또 다른 Object 를 반환할 수 있습니다.

몇몇 Property 는 아래처럼 간단한 값들을 출력합니다.

Range("A1").Address	: 셀주소를 반환합니다.
Range("A1").Value	: 셀 안에 입력된 값을 반환합니다.
Range("A1").Column	: 셀의 열번호를 반환합니다.
Range("A1").RowHeight	: 셀의 행높이를 반환합니다.

반면에 대다수의 많은 Property 는 또 다른 Object 를 반환합니다.

ThisWorkbook.Worksheets("Sheet1")	
.Range("A1").Interior.Pattern	: 셀의 무늬(패턴)
.Range("A1").Font.Color	: 글꼴 색
.Range("A1").Comment(1)	: 셀의 메모
.Range("A1").EntireRow	: 해당 셀의 전체 행

Object 를 반환하면 무엇이 다른가요?

아래의 간단한 예를 보겠습니다.

```
Dim Rng as Range

Set Rng = Worksheets("Sheet1").Range("A1")
'// Rng 는 Object 이므로 Set 을 통해서 변수를 지정해줍니다.

Comment = Rng.Comment(1)
'// Comment 가 Object 로 반환되므로, Set 없이 지정하게 되면 VBA 에서는 오류를 출력합니다.
'// 따라서 각 Property 에서 반환되는 값을 잘 이해하고 있으면 더욱 빠르게 코드를 작성 할 수 있습니다.
```

3. 읽기전용(Read-Only) Property

몇몇 Property 는 읽기만 가능합니다. 즉 해당 VBA Property(속성)에 다른 값을 덮어씌우는 것이 불가능합니다. 대표적으로 아래 항목들이 있습니다.

Range.Address : 범위의 셀 주소입니다. 셀의 고유값이므로 읽기전용입니다.
 Application.ActiveWorkbook.Path : 워크북의 경로입니다. 이 또한 엑셀이 실행중인 한 변경 불가능한 고유값이므로 읽기전용입니다.
 Application.ActiveWorkbook.FullName : 워크북의 경로를 포함한 이름입니다. 경로와 마찬가지로 읽기전용으로 사용됩니다.

하지만 대부분의 Property 는 읽기와 쓰기 모두 가능합니다.

Range("A1").Value = ThisWorkbook.Name
 '// A1 셀의 값을 해당 워크북의 파일명으로 덮어씌웁니다.

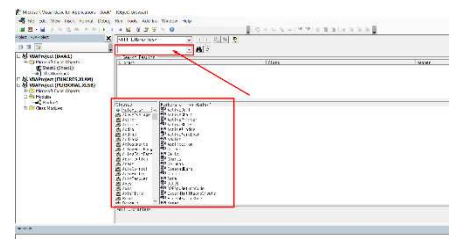
Range("A1").Font.Color = vbBlue
 '// A1 셀의 글꼴색을 파란색으로 변경합니다.

Range("A1").RowHeight = Range("A1").ColumnWidth
 '// A1 셀의 행높이를 A1 셀의 열넓이 값으로 변경합니다.

4. 각 Object(개체)의 Property(속성)를 쉽게 확인하는 방법

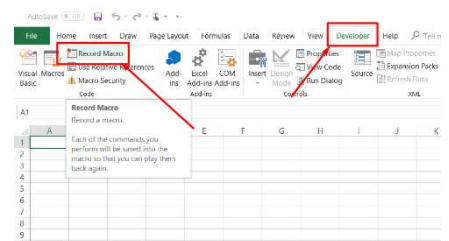
다양한 방법이 있겠지만 저는 주로 3 가지 방법을 사용합니다.

1. VBA 편집기에서 키보드 F2 키를 눌러 라이브러리 창을 불러옵니다. 원하는 개체 또는 클래스를 검색하면 해당 항목에 대한 모든 Property 및 Method 그리고 Event 에 대해 확인할 수 있습니다.



2. 엑셀에서 Macro Recording (매크로 녹화기능)을 활용합니다. 매크로 녹화기능을 통해 코드를 작성하게 되면 모든 행동 하나하나가 상당히 자세하게 기록됩니다. 따라서 매크로 녹화기능을 통해 작성된 코드를 분석하면 원하는 행동에 대한 Key Property 또는 Method 를 쉽게 확인할 수 있습니다.

개발자틀(Developer)를 활성화하는 방법은 [여기](#)를 참고하세요.



3. StackOverFlow 와 같은 프로그래밍 언어 커뮤니티를 잘 활용하면, 때로는 생각했던 것 이상의 결과물을 얻을 수도 있습니다.

5. 예제 코드 전문

'// 삽입(Insert) -> 모듈(Module) 에서 모듈을 추가한 뒤 아래 코드를 삽입하세요.

```
Sub Test()
```

'// 변수 할당

```
Dim WB As Workbook
```

```
Dim WS As Worksheet
```

```
Dim Rng As Range
```

'// 변수 지정

```
Set WB = ThisWorkbook
```

```
Set WS = WB.Worksheets("Sheet1")
```

```
Set Rng = WS.Range("A1")
```

'// 개체 속성 변경

```
Rng.Value = WB.Name
```

```
Rng.Font.Color = vbBlue
```

```
Rng.RowHeight = Rng.ColumnWidth
```

```
End Sub
```