

프로그래밍 기본 개념 이해하기

학습내용

1. VBA의 구성요소
2. 이벤트 프로시저

학습목표

1. VBA 프로그래밍을 위한 다양한 구성요소를 알 수 있다.
2. 특정 이벤트가 발생할 때 실행되는 프로시저를 작성할 수 있다.

01. VBA 구성요소

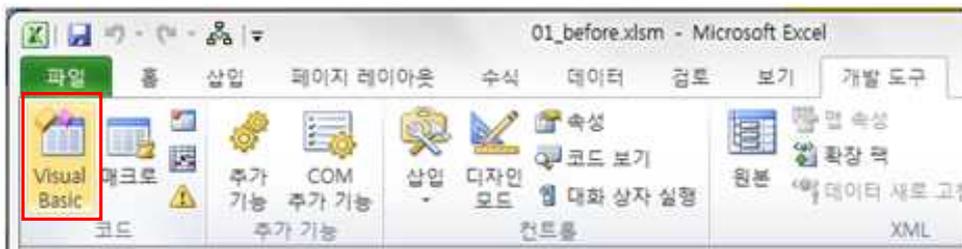
1. VBE(Visual Basic Editor)

1) VBE 란?

- Visual Basic Editor의 약자
- 엑셀을 자동화하기 위해 필요한 코드 개발 및 오류 수정 등의 작업을 진행할 수 있는 편집기

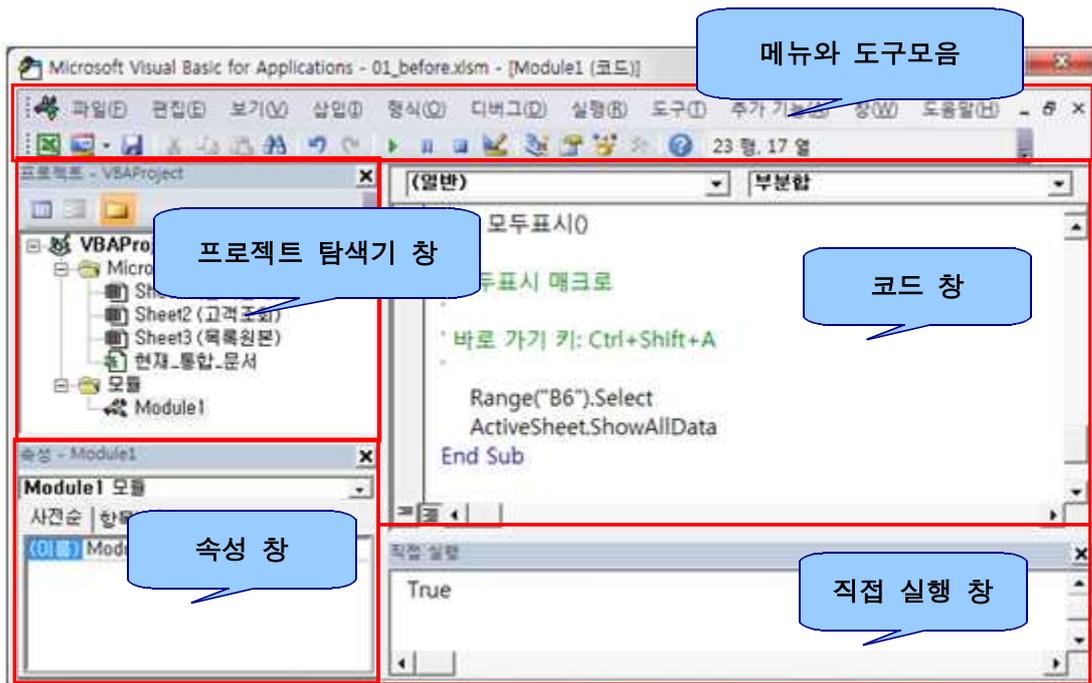
2) 실행 방법

- [개발 도구] 탭 → [코드] 그룹 → [Visual Basic] 아이콘 클릭



- 단축키 : Alt + F11

2. VBE 구성요소



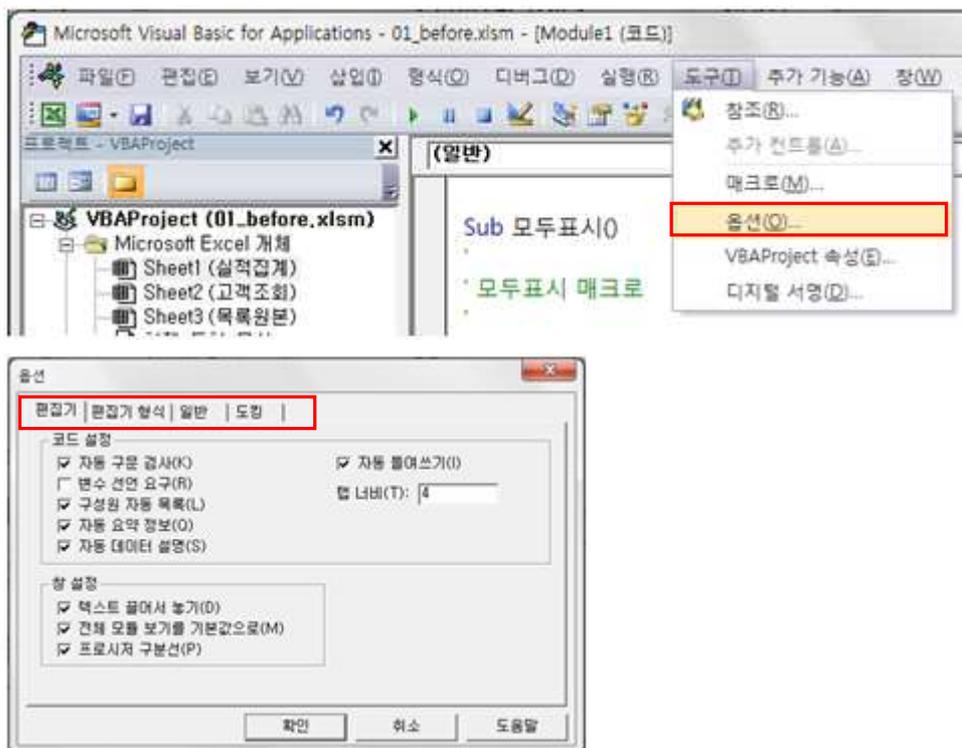
메뉴와 도구모음	- VBE에서 사용할 수 있는 명령을 메뉴와 도구 모음으로 제공하는 영역
프로젝트 탐색기	- 현재 열려있는 파일과 각 파일에 속해 있는 모든 항목(워크시트, 폼, 모듈 등)이 계층 구조로 표시되는 영역
속성 창	- [프로젝트 탐색기]에서 선택한 개체의 속성을 지정하는 곳
코드 창	- VBA 코드를 작성하는 곳
직접 실행창	- 직접 실행 창은 간단한 실행 결과를 바로 확인할 때 사용하는 창 - '?1+2' 를 입력하고 [Enter]키를 누르면 결과 3 표시

3. VBE 환경 설정

- VB 편집기 사용자는 VB 편집기의 환경을 원하는 대로 설정 가능
(코드 창의 글꼴, 크기, 색상 설정, 코드 작성 시 표시되는 목록이나 설명의 표시 설정, 열리는 창 들의 도킹 여부 등)

1) 설정 방법

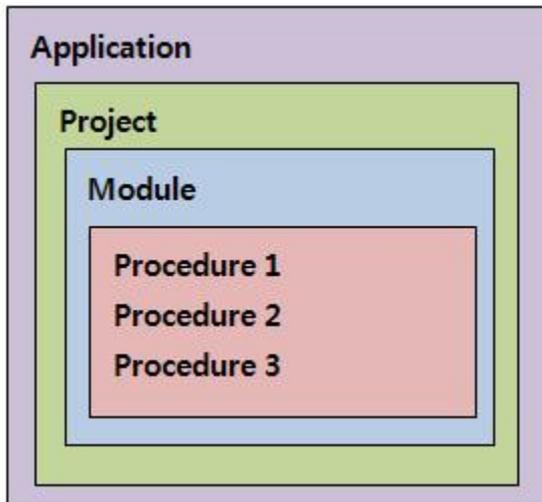
- [도구] → [옵션] 메뉴를 선택, 각 탭에서 원하는 옵션 지정 가능



탭 구분	설명
[편집기] 탭	직접 코드를 작성할 때 필요한 부가적인 정보에 대한 표시여부 설정
[편집기 형식] 탭	코드 창에 나타나는 글자의 속성 지정
[일반] 탭	일반적인 옵션 사항 설정 (기본값으로 사용하는 것이 일반적)
[도킹] 탭	각 창의 도킹 여부 설정

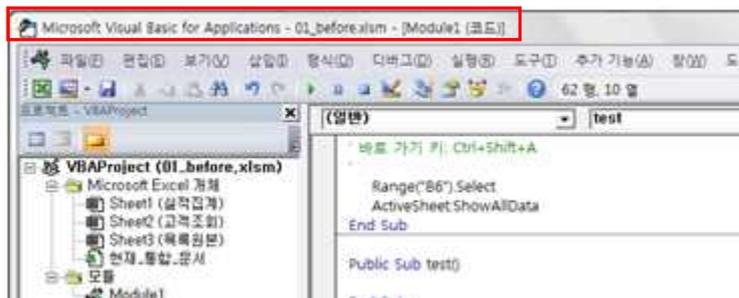
4. VBA 기본 구조

- 개체 : VBA 프로그램의 각 구성요소, 개체는 계층 구조를 가짐



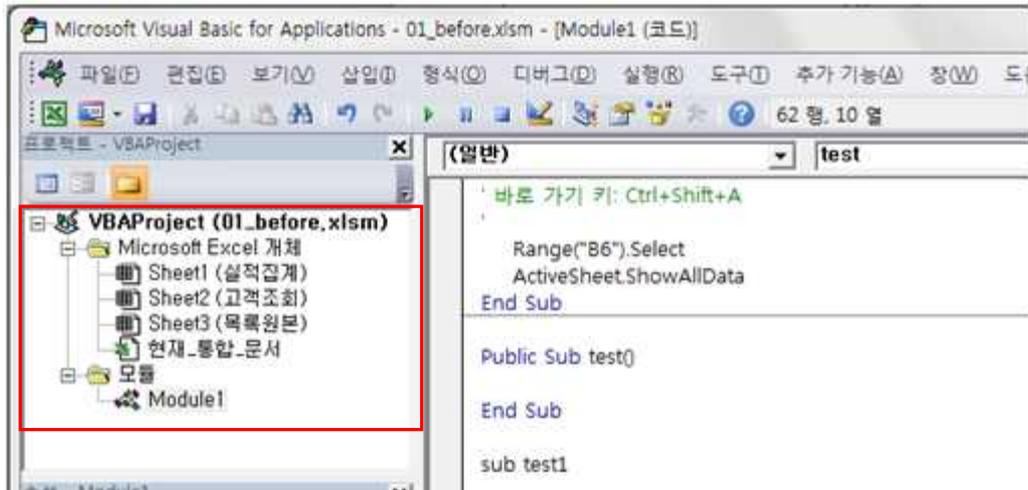
① Application (응용 프로그램)

- 현재 실행 중인 프로그램인 엑셀 자체를 의미
- VBA 코드의 최상위 계층 의미



② Project

- 하나의 통합 문서에 작성되는 모든 VBA 코드 내용
- Workbook (통합문서), Worksheet, Module, 폼 등으로 구성

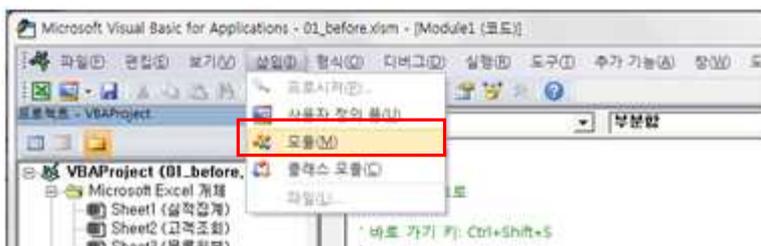


③ Module

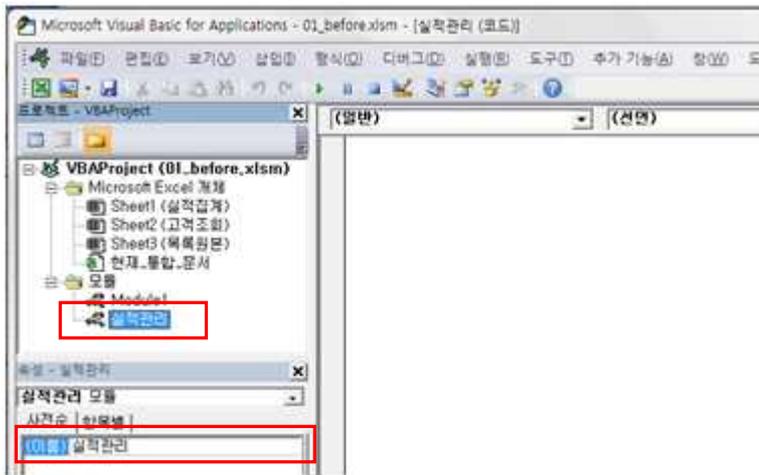
- Procedure의 집합, 표준 모듈과 폼 모듈, 클래스 모듈로 구분됨

표준 모듈	<ul style="list-style-type: none"> - 워크시트 모듈(Sheet로 표시되는 모듈), ThisWorkbook 모듈, 공용 모듈이 존재함 - 워크시트 모듈 : 워크시트마다 자동으로 각각 하나씩 만들어지고, ThisWorkbook 모듈 : 통합 문서 즉, 엑셀 파일에 하나가 만들어 짐
폼 모듈	<ul style="list-style-type: none"> - 사용자 정의 폼을 디자인하고, 사용자 정의 폼의 컨트롤에 이벤트 프로시저를 작성하는 모듈
클래스 모듈	<ul style="list-style-type: none"> - 개체를 새롭게 정의해서 사용할 수 있도록 작성하는 모듈 - 개체의 속성, 메서드, 이벤트를 정의하는 모듈

▶ 모듈 삽입 : Project에 표준 모듈을 삽입하려면 [삽입] → [모듈] 메뉴 활용



▶ **모듈 이름 정의** : 삽입된 모듈의 이름은 [속성] 창의 [이름] 속성에서 지정



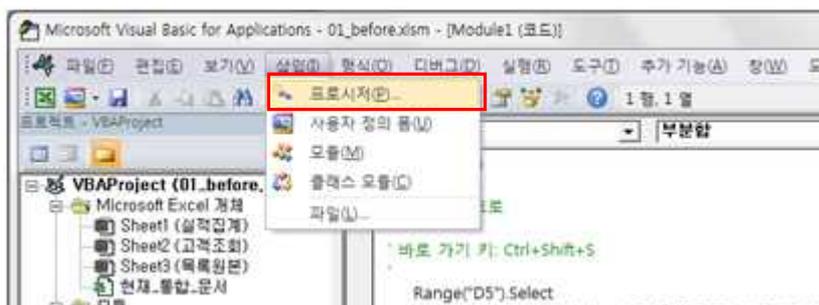
④ Procedure

- 특정 작업을 실행하기 위해 모여진 실행문의 집합
- 실행 방법에 따라 Sub Procedure, Function Procedure, Property Procedure로 구분 됨.

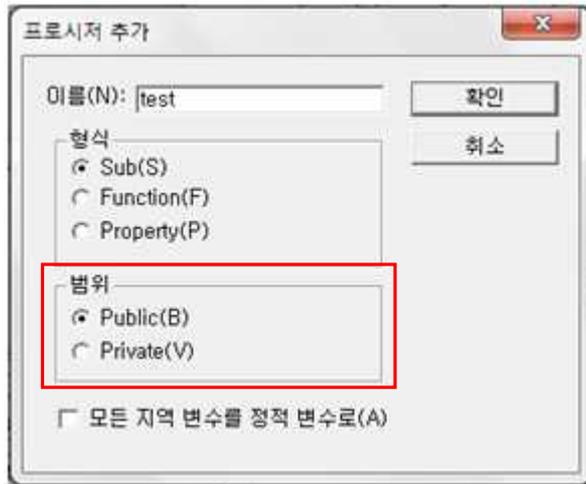
Sub Procedure	<ul style="list-style-type: none"> - 작성한 코드를 순차적으로 실행하는 Procedure - 매크로 기록기를 사용하여 기록된 엑셀 작업을 순차적으로 실행하는 매크로도 Sub Procedure로 작성됨
Function Procedure	<ul style="list-style-type: none"> - 엑셀의 함수 기능을 수행하는 Procedure - 엑셀에서 제공하지 않는 함수를 직접 Function Procedure를 작성하여 워크시트에서 함수를 사용하듯이 사용 가능 - Sub Procedure처럼 작성한 코드를 순차적으로 실행하지만, Sub Procedure와 다른 점은 실행문의 결과 값을 Function 이름에 반환

▶ Procedure 작성 방법

① 코드 창 임의의 위치에 클릭한 다음 [삽입] → [프로시저] 메뉴 선택



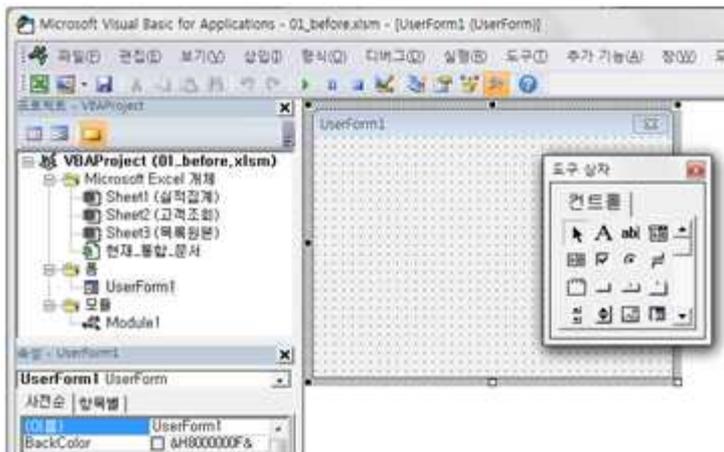
② 프로시저의 이름, 형식, 범위를 지정한 후, <확인> 단추 클릭



- Public : 다른 모듈의 프로시저에서 해당 Procedure를 호출 가능
- Private : 해당 프로시저가 속한 모듈 내의 다른 프로시저에서만 해당 프로시저를 호출 가능

⑤ User Form (사용자 정의 폼)

- 자료의 입출력을 효과적으로 하기 위한 양식(폼)



5. Procedure의 구성요소

1) 개체 (Object)

- 엑셀, 통합문서, 통합 문서를 구성하는 워크시트, 셀 등을 모두 개체로 인식
- 대표적 개체 : Application, Workbook, Worksheet, Range 등

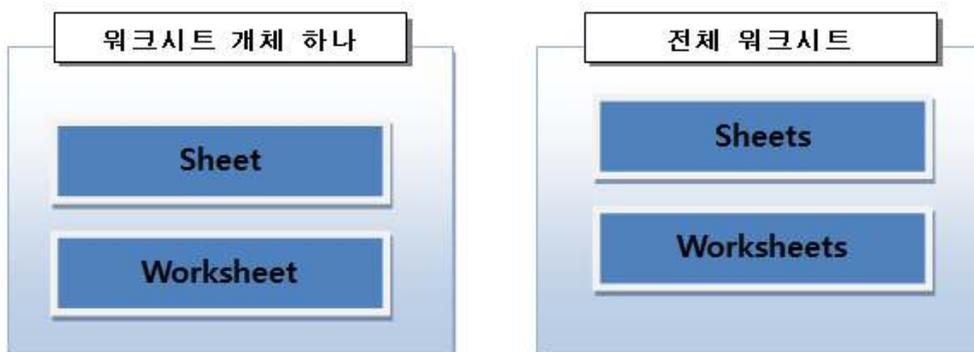
2) 컬렉션 (Collection)

- 개체들의 집합, 일반적으로 개체 이름에 복수형 's' 를 붙여 표현

통합문서의 첫 번째 시트 지칭하기

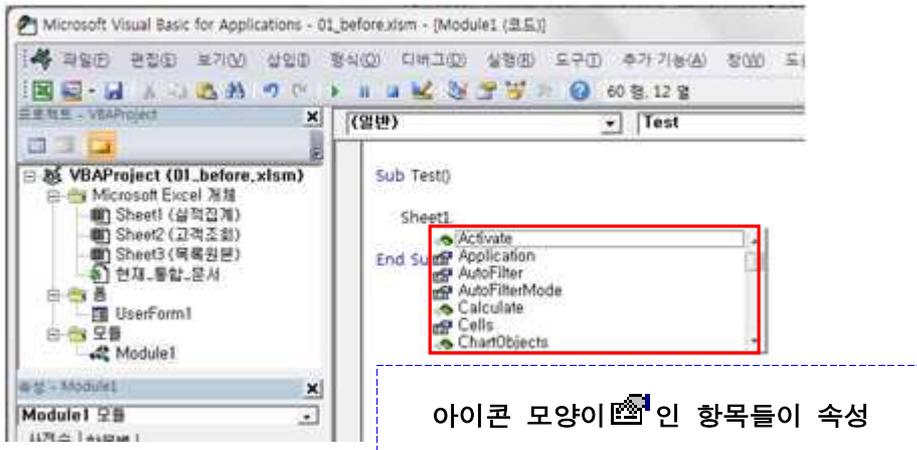
방법 1 : Sheets(1), WorkSheets(1)

방법 2 : Sheets("Sheet1"), WorkSheets("Sheet1")



3) 속성 (Property)

- 개체가 가지는 **특성**으로 셀의 주소, 글꼴, 글꼴 색, 워크시트의 이름 등이 모두 개체가 가지는 속성
- 개체와 속성 사이에는 ‘.(점)’ 을 찍어 구분
- 예) 첫 번째 워크시트의 이름을 의미하는 코드 : `Sheet1.Name`



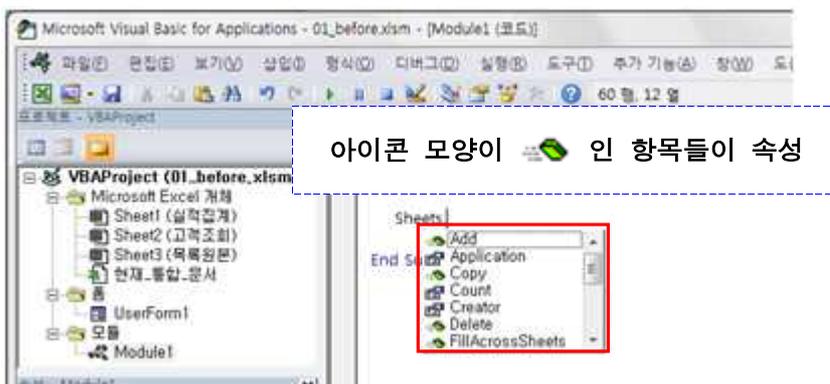
- 개체와 속성 입력 방법

개체.속성 = 속성 값

첫 번째 워크시트의 이름을 1월로 지정하는 코드 → `Sheet1.Name = "1월"`
 [A1]셀에 '10' 을 입력하는 코드 → `Range("a1").Value = 10`

4) 메서드 (Method)

- 개체가 **실행할 수 있는 동작**
- 워크시트 삽입, 삭제, 특정 셀을 선택, 내용 지우기 등의 동작이 메서드
- 개체와 메서드 사이에 ‘.(점)’ 으로 구분



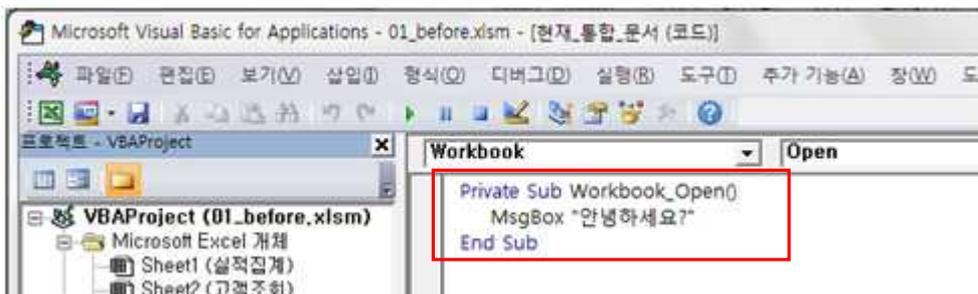
- 개체와 메서드 입력 예시

```

워크시트 삽입하는 코드 → Sheets.Add
[A1]셀 선택하는 코드 → Range( "A1" ).Select
    
```

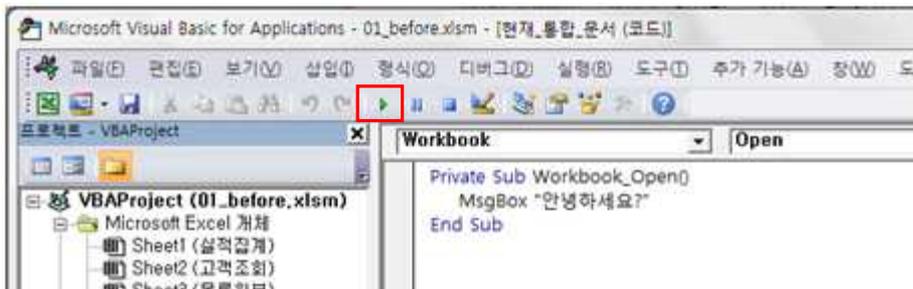
5) 이벤트 (Event)

- 개체가 일으키는 사건
- 통합문서를 열거나 닫을 때, 워크시트가 활성화 될 때, 워크시트의 특정 셀이 변경될 때



6. Sub Procedure 실행

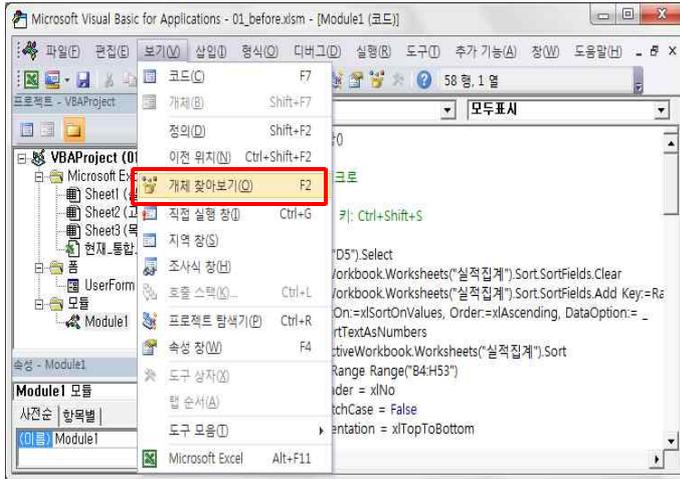
- [표준] 도구 모음의 [Sub/사용자 정의 폼 실행(F5)] 버튼을 클릭 / [F5]키 활용



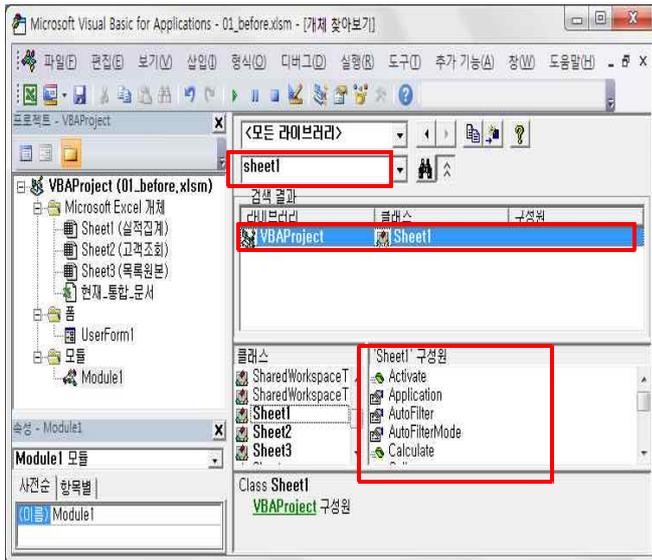
Tip >> 개체 찾아보기와 도움말 창 활용하기

1. 개체 찾아보기 : 엑셀 VBA 에서는 지원하는 다양한 개체들마다 사용 가능한 속성, 메서드, 이벤트 들이 서로 다르기 때문에 이런 모든 것들을 외워서 프로그래밍하는 것은 불가능 하다. 그러므로 개체 찾아보기 창이나 VBA 도움말 등을 참고하는 것이 효과적이다.

① VBE 창에서 [보기] → [개체 찾아보기]를 선택하거나[F2]키를 누른다.



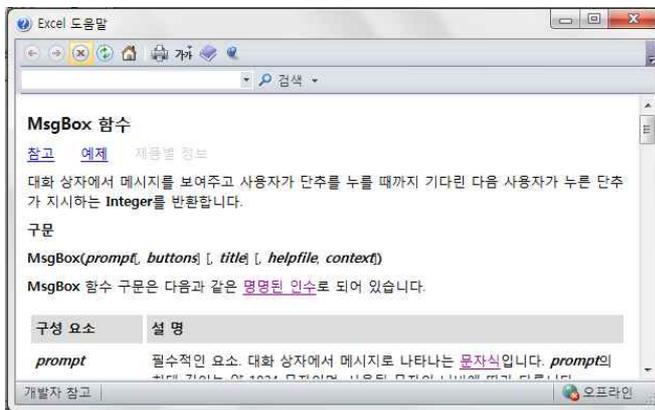
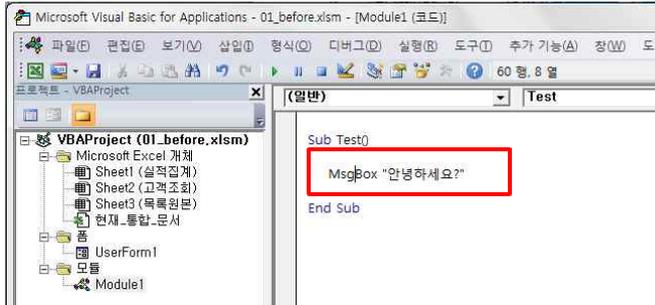
② 검색하고자 하는 개체 이름을 입력하고 [Enter]키를 누르면 해당 키워드를 포함하는 개체들이 [검색 결과]에 표시된다. [검색 결과] 중 원하는 개체를 선택하면 아래쪽에 해당 개체의 속성, 메서드 등이 표시된다.



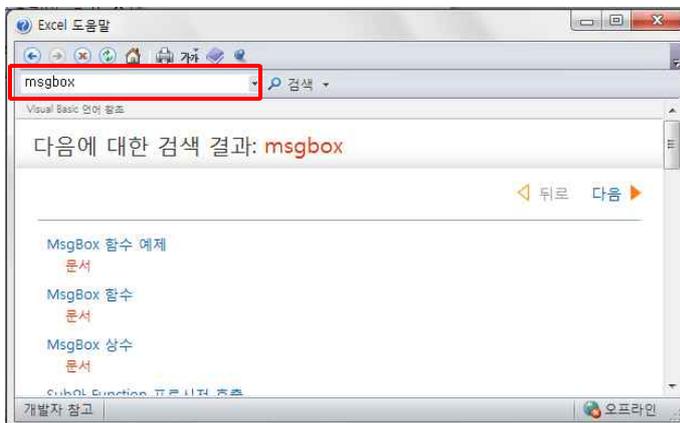
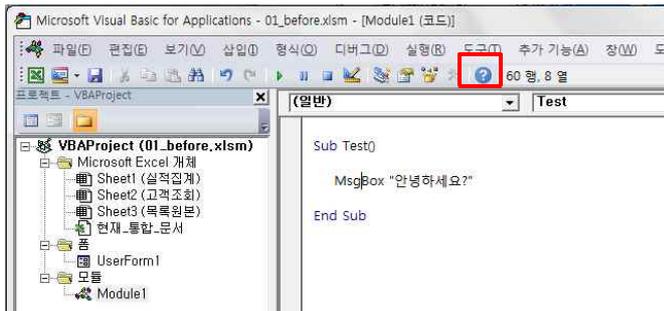
2. 도움말 사용하기

방법 1 : 작성한 VBA 코드에 커서를 놓고 [F1]키를 누른다.

화면에 표시되는 도움말과 상단의 [참고], [예제] 등을 통해 코드에 대한 도움을 얻을 수 있다.



방법2 : 도움말 창을 실행한 후, 검색어를 입력하는 방법으로도 가능하다.



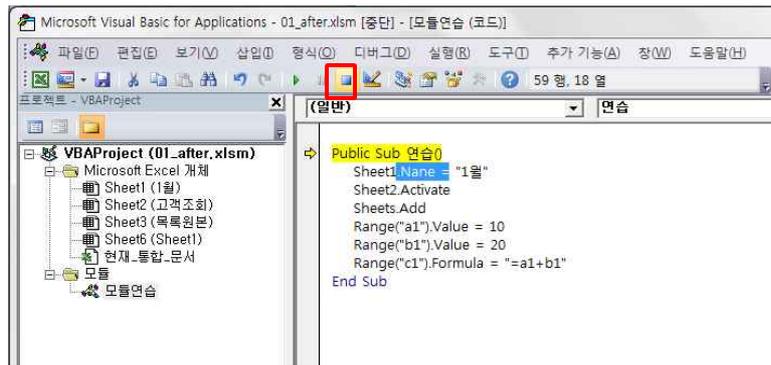
Tip >> VBA 오류

1. 컴파일 오류

컴파일(Visual Basic 언어를 컴퓨터가 이해할 수 있는 언어로 변환) 작업 시 발생한 오류로 다양한 원인이 있으나 개체나 구성원 이름의 철자가 틀린 경우에 주로 발생한다.

<대처법>

[표준] 도구 모음의 [재설정] 버튼을 클릭하고 오류의 원인을 찾아 수정

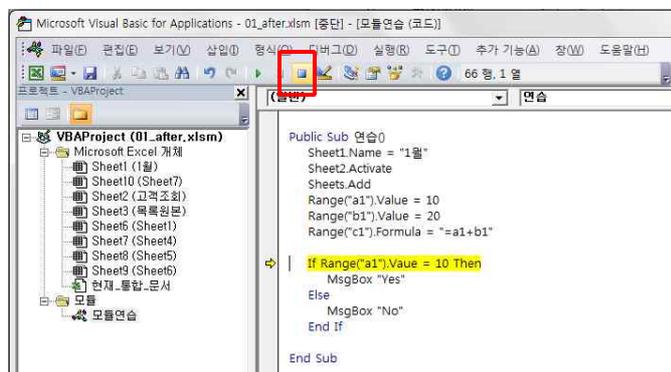
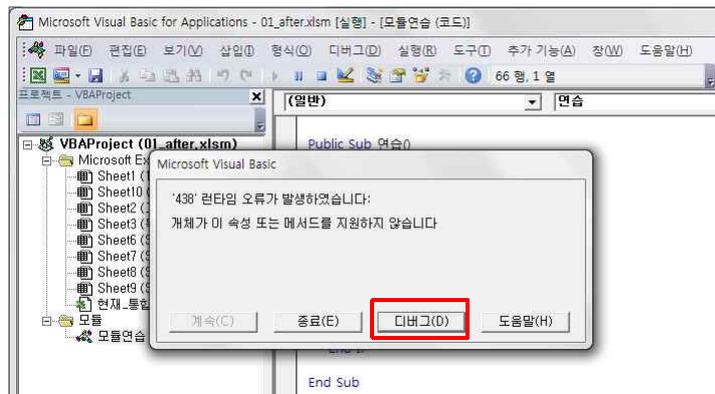


2. 런타임 오류

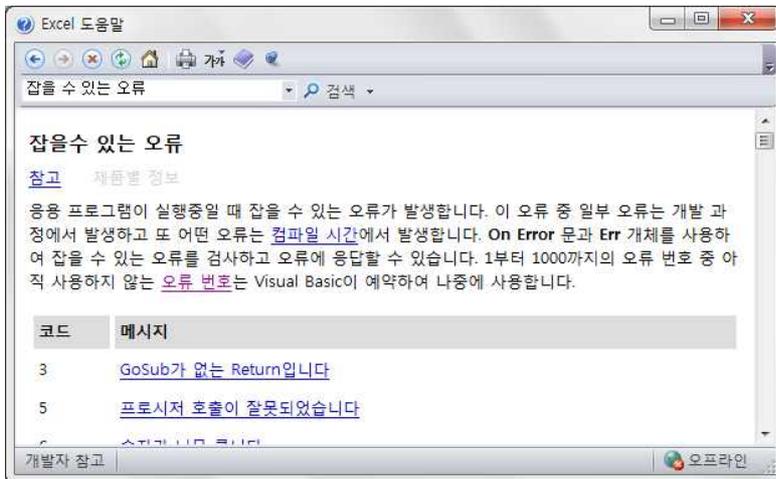
코드가 실행되는 동안 발생하는 오류를 의미한다. 런타임 오류의 경우 오류 메시지를 표시하는 대화상자에 표시되는 오류 코드를 통해 오류의 원인을 파악 가능하다.

<대처법>

[디버그] 버튼을 클릭한 후, [표준] 도구 모음의 [재설정] 버튼을 클릭하고 오류의 원인을 찾아 수정



3. 도움말을 활용하여 오류의 원인 확인 가능



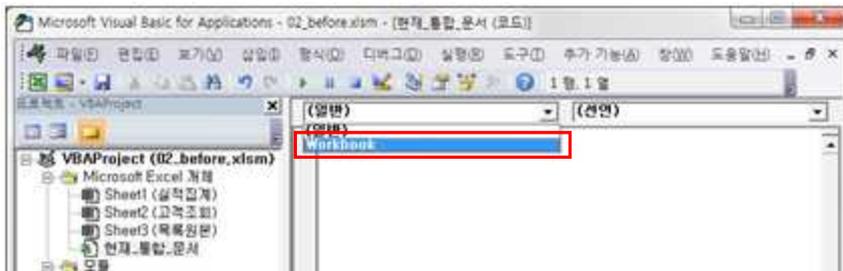
02. 이벤트 프로시저

1. 이벤트 프로시저란?

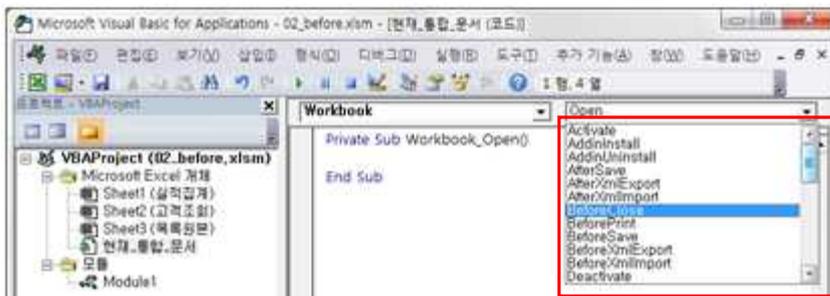
- 개체에 특정 이벤트가 발생할 때 실행되는 프로시저
- 예) 통합 문서가 열릴 때 특정 시트를 자동 활성화
특정 시트가 선택되면, 시트의 내용을 수정하지 못하도록 시트 보호

2. 작성 방법

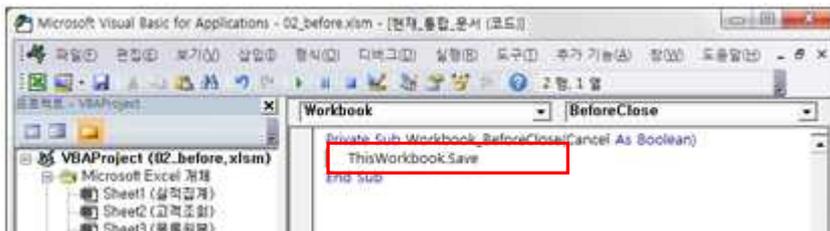
- ① 이벤트를 실행하는 개체 선택하고, 해당 개체가 발생시키는 이벤트 선택



- ② [프로시저 목록]에서 원하는 이벤트 선택



- 삽입된 이벤트 프로시저에 해당 이벤트가 발생할 때 실행할 실행문 작성



3. 개체 별 주요 이벤트

- 통합 문서 주요 이벤트

이벤트	설명
Activate	통합문서가 활성화될 때 발생
BeforeClose	통합문서를 닫기 전에 발생
BeforePrint	통합문서를 인쇄하기(미리보기 포함) 전에 발생
BeforeSave	통합문서를 저장하기 전에 발생
DeActivate	통합문서가 비 활성화될 때 발생
NewSheet	통합문서에 새 워크시트를 삽입할 때 발생
Open	통합 문서를 열 때 발생

- 통합 문서 주요 이벤트 중 모든 워크시트에 공통으로 적용되는 이벤트

: 모든 워크시트에 공통으로 적용되는 이벤트를 작성하려면 통합 문서 이벤트 중 Sheet로 시작하는 이벤트를 사용

이벤트	설명
SheetActivate	통합문서 내 워크시트를 선택할 때 발생
SheetBeforeDoubleClick	통합문서 내 워크시트에서 더블클릭 할 때 발생
SheetBeforeRightClick	통합문서 내 워크시트에서 마우스 오른쪽 버튼을 클릭할 때 발생
SheetCalculate	통합문서 내 워크시트에서 계산이 되는 경우에 발생
SheetChange	통합문서 내 워크시트에서 값이 수정될 때 발생
SheetDeActivate	통합문서 내 워크시트가 비활성화 될 때 발생
SheetSelectionChange	통합문서 내 워크시트 내의 셀을 선택할 때 발생

- 워크시트 주요 이벤트

이벤트	설명
Activate	워크시트가 활성화될 때 발생
BeforeDoubleClick	워크시트에서 더블클릭할 때 발생
BeforeRightClick	워크시트에서 마우스 오른쪽 버튼을 클릭할 때 발생
Calculate	워크시트에서 계산이 되는 경우 발생
Change	워크시트의 값이 수정될 때 발생
DeActivate	워크시트가 비활성화될 때 발생
SelectionChange	워크시트 내의 셀을 선택할 때 발생

의견 나누기

➤ 문제 제시

통합 문서가 닫히기 전에 ‘수고하셨습니다!’ 메시지 박스를 표시한 후, 통합 문서에 변경 사항이 있다면 무조건 저장하는 이벤트 프로시저를 작성하려고 합니다. 어떻게 작성해야 할까요?

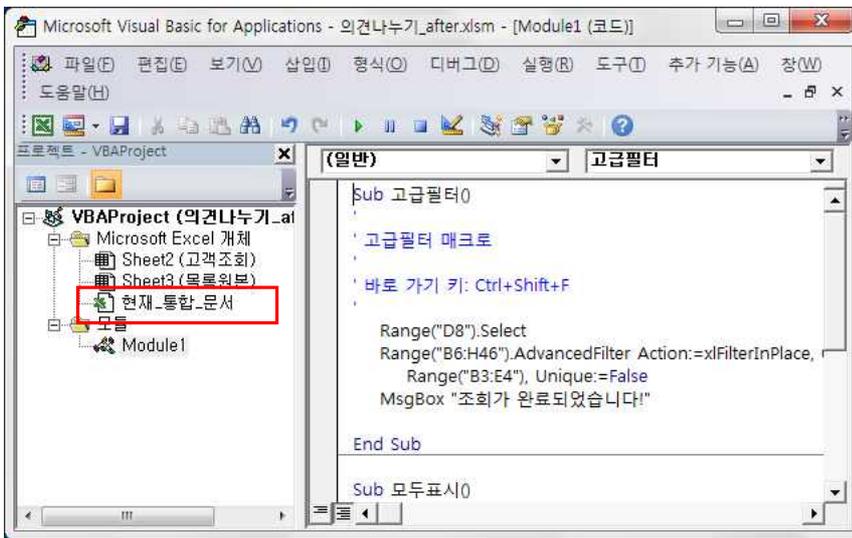
일자	일자	지점	분류	조회	모두표시
>=2011-4-1	<=2011-12-31	서울	컴퓨터		

일자	지점	단가	수량	판매액		
2011-01-06	서울	1,500,000	50	75,000,000		
2011-01-08	서울	150,000	45	6,750,000		
2011-01-15	대전	350,000	20	7,000,000		
2011-01-29	대구	2,000,000	25	50,000,000		
2011-01-29	대전	2,500,000	5	12,500,000		
2011-02-02	서울	2,000,000	20	40,000,000		
2011-02-05	부산	컴퓨터	모니터	300,000	7	2,100,000
2011-02-18	대구	가전	세탁기	700,000	12	8,400,000
2011-03-01	대전	디지털	캠코더	1,000,000	15	15,000,000
2011-03-03	대전	디지털	MP3	150,000	12	1,800,000

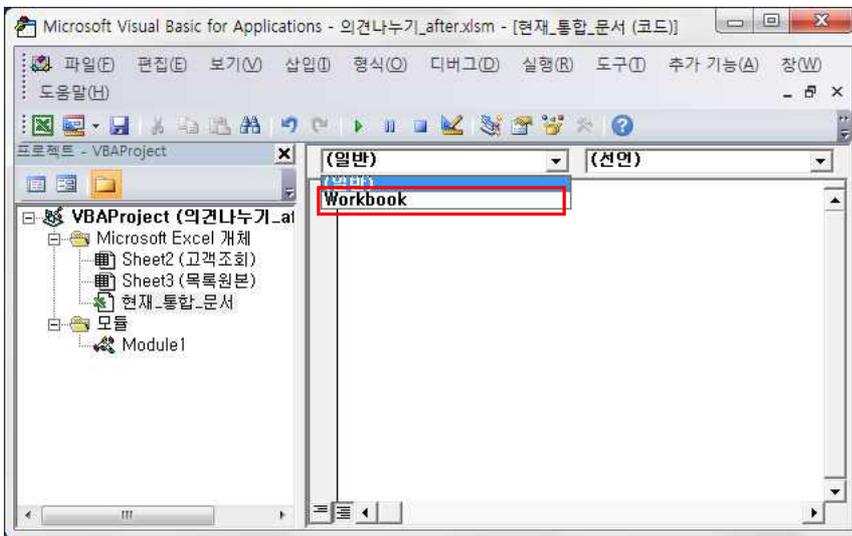
[전문가 의견]

- 먼저 통합 문서가 닫히기 전이므로 Workbook_BeforeClose 이벤트 프로시저를 작성합니다.
- 메시지 박스를 표시하는 실행문은 'MsgBox "수고하셨습니다!"'입니다.
- 또한, 자동으로 저장되도록 설정하기 위한 실행문은 'ThisWorkbook.Save' 코드입니다.

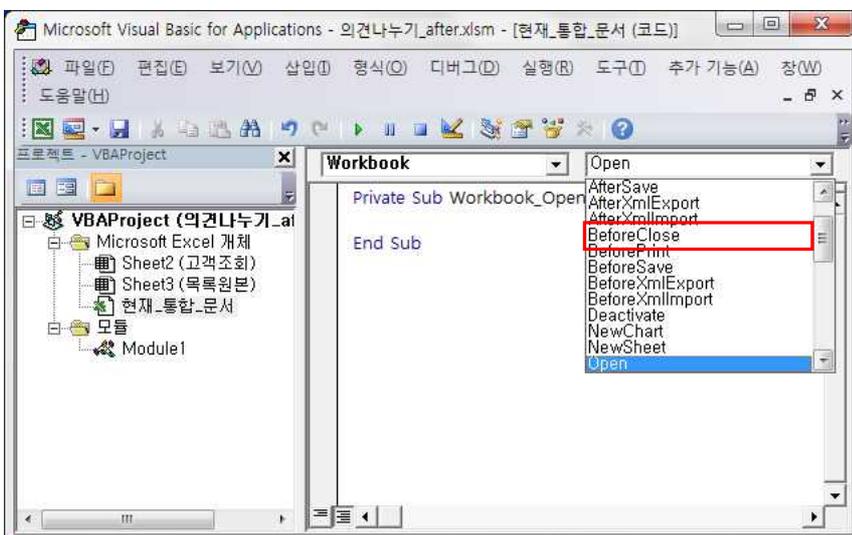
① VB 편집기 창이 표시되면 화면 왼쪽 [프로젝트 탐색기]에서 [현재_통합_문서]를 더블클릭합니다.



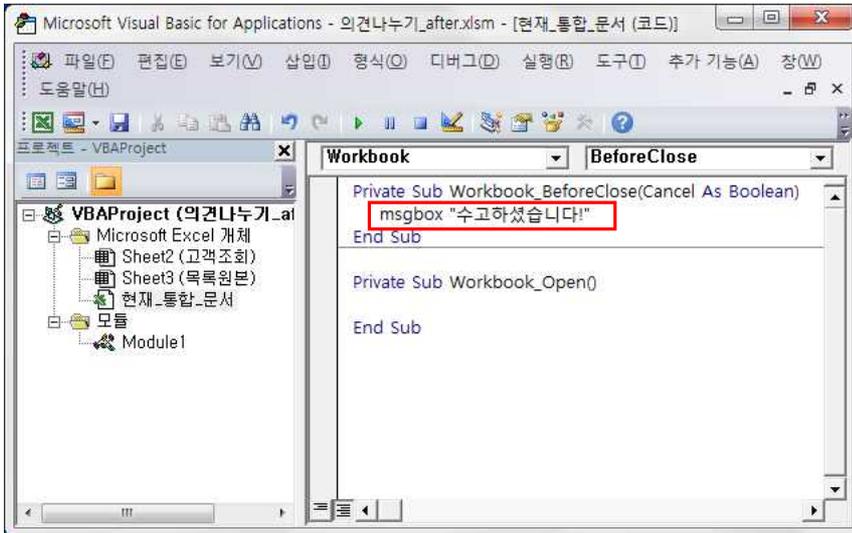
② 현재 통합 문서 코드 창이 표시되면 [개체 목록]의 목록 단추 클릭한 후 [Workbook]을 선택합니다.



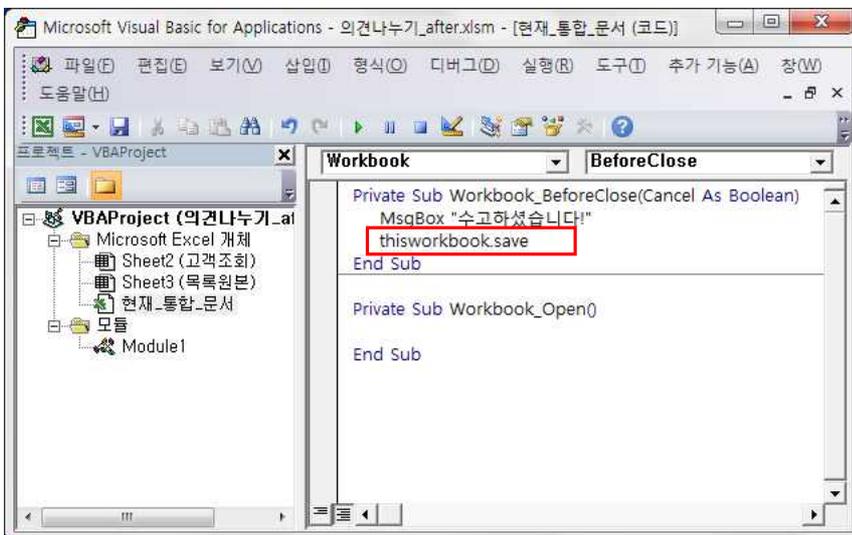
③ [프로시저 목록]의 목록 단추를 클릭한 다음 [BeforeClose] 이벤트를 선택합니다.



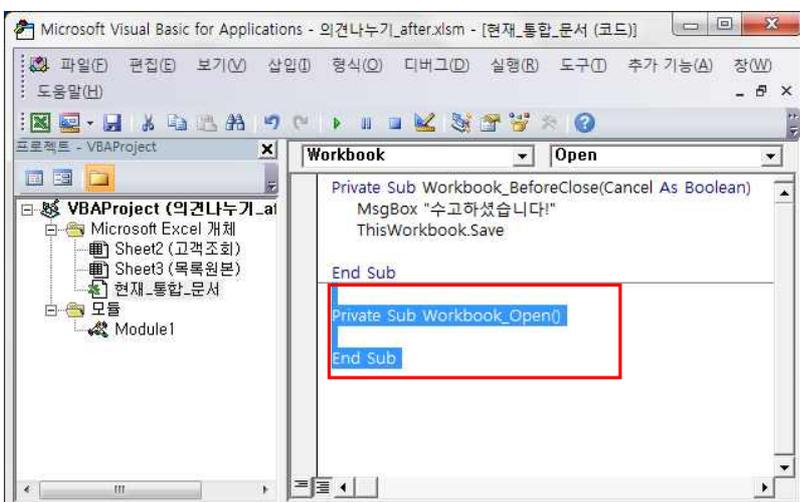
④ 메시지 박스를 표시하기 위해서 삽입된 BeforeClose 이벤트 프로시저에 'msgbox "수고하셨습니다!"' 를 입력하고 [Enter]키를 누릅니다.



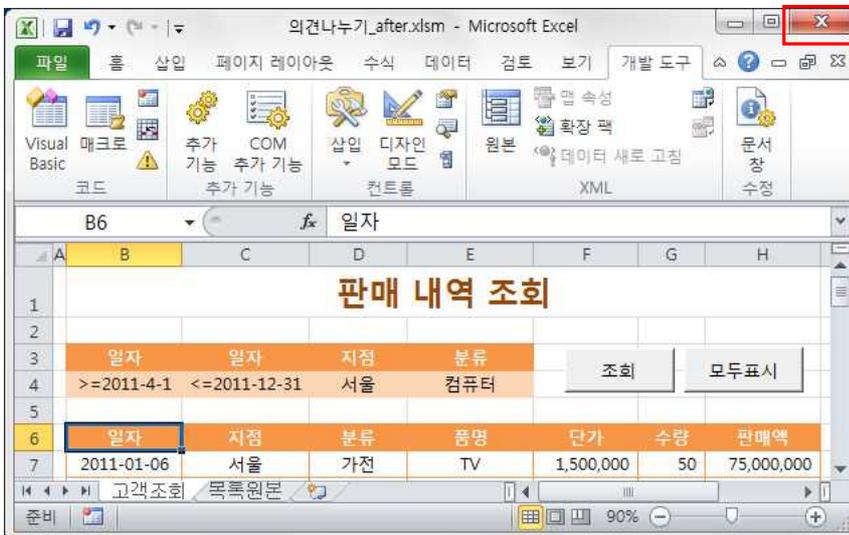
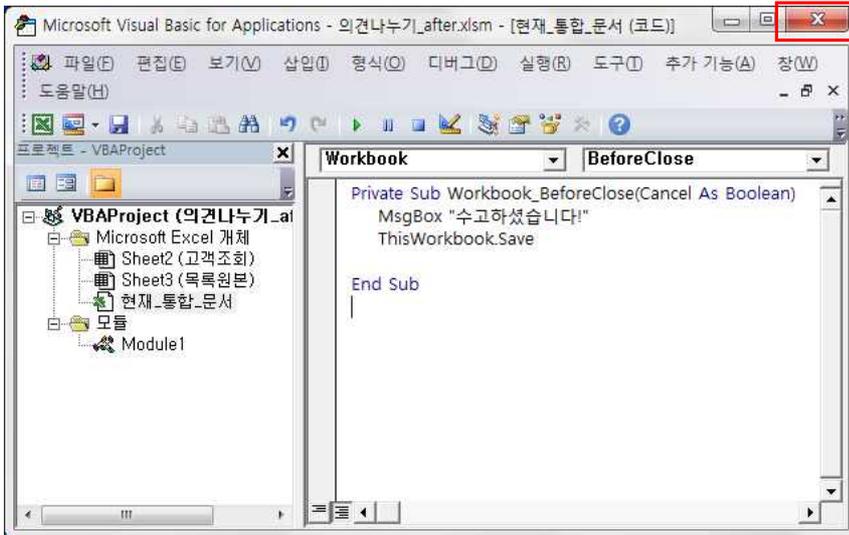
⑤ 문서 자동 저장을 위해 'thisworkbook.save' 를 입력한 후, [Enter]키를 누릅니다.



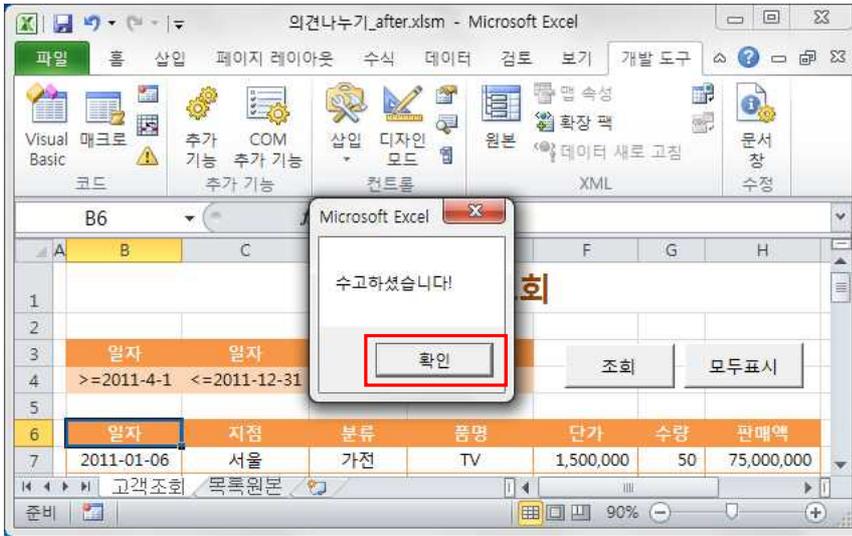
⑥ 자동으로 삽입된 Workbook_Open 이벤트 프로시저는 사용하지 않을 것이므로 삭제합니다.



⑦ VB 편집기 창의 [닫기] 버튼을 클릭하여 창을 닫고, 워크시트로 이동합니다.엑셀 창의 [닫기] 버튼을 클릭하여 통합 문서를 닫습니다.



⑧ '수고하셨습니다!' 메시지 박스가 표시되고 <확인> 단추를 클릭하면 문서의 변경 내용이 자동으로 저장되고 문서가 닫힙니다.



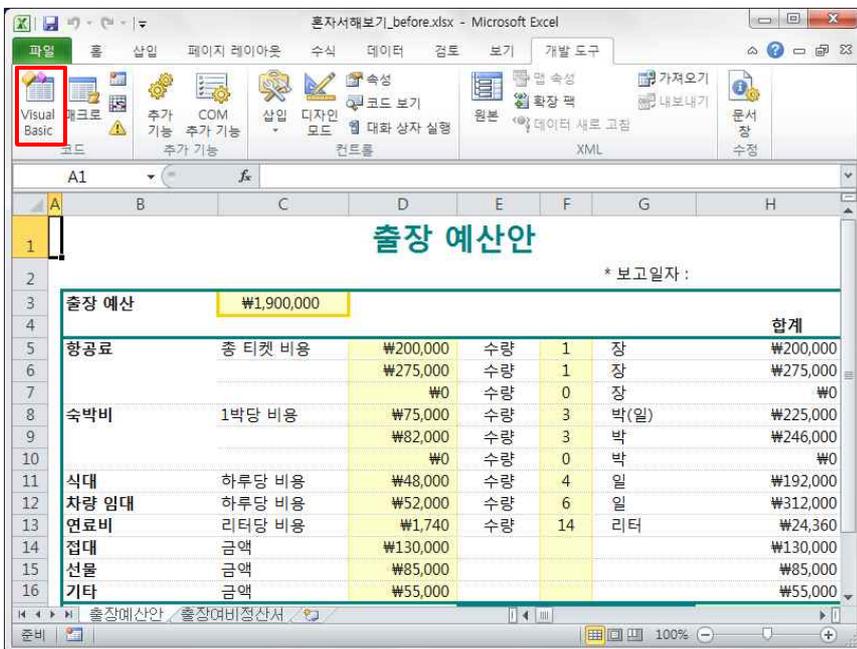
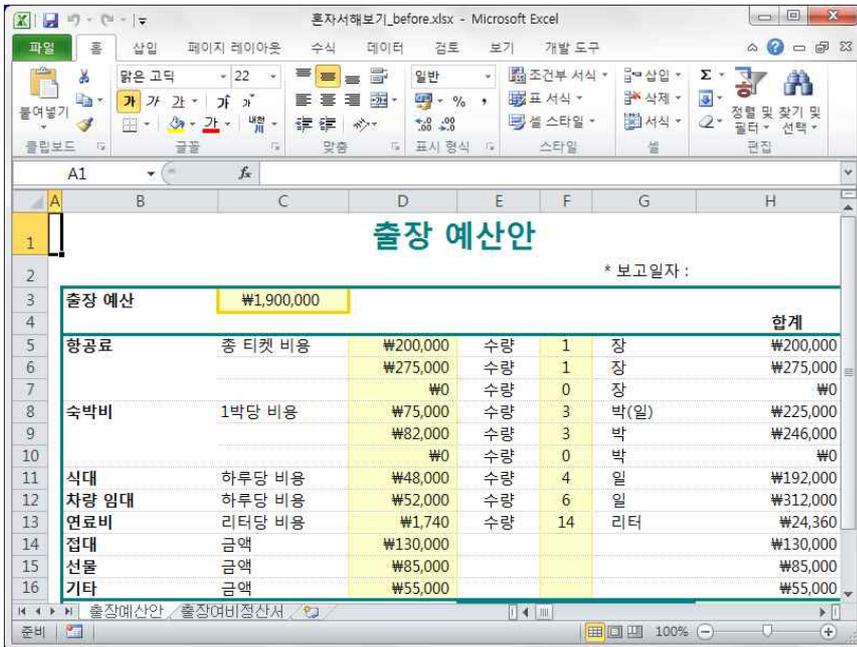
혼자서 해보기

➤ 다음 조건을 차례대로 수행하시오.

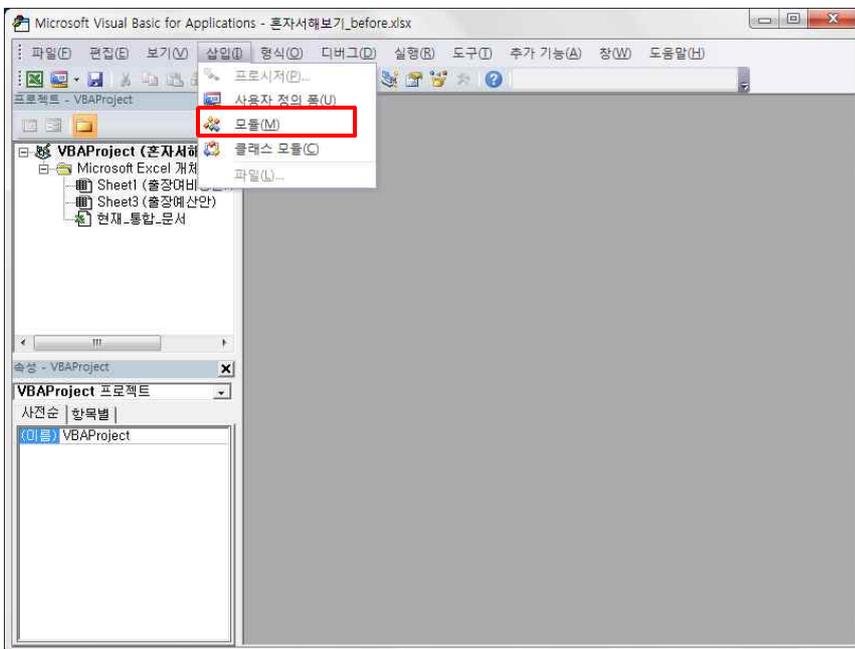
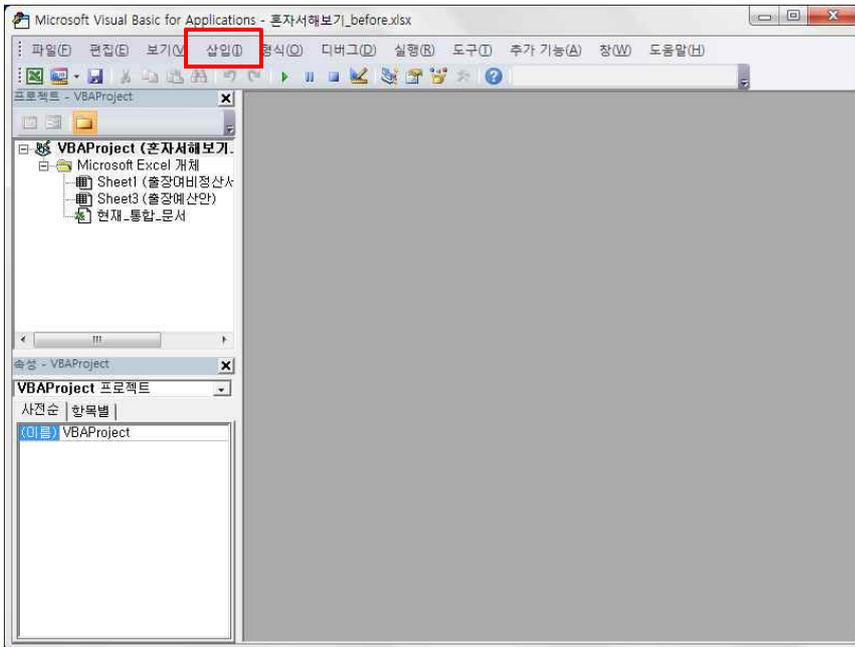
- ① VB 편집기 창으로 이동한 후 새 모듈을 삽입하고 이름을 ‘매출관리’ 로 지정하시오.
- ② ‘연습’ 이라는 이름의 Sub Procedure를 삽입하시오.
- ③ 연습Sub Procedure 에 [출장예산안] 시트를 활성화 시킨 후 [H3]셀에 오늘 날짜를 입력하는 실행문을 작성하시오. (VBA에서 오늘 날짜를 반환하는 함수는 Date 임)
- ④ 연습Sub Procedure를 실행한 후 워크시트에서 결과를 확인하시오.

출장 예산안							
* 보고일자						2011-10-29	
출장 예산		₩1,900,000				합계	
합공료	출 티켓 비용	₩200,000	수량	1	장	₩200,000	
		₩275,000	수량	1	장	₩275,000	
		₩0	수량	0	장	₩0	
숙박비	1박당 비용	₩75,000	수량	3	박(일)	₩225,000	
		₩82,000	수량	3	박	₩246,000	
		₩0	수량	0	박	₩0	
식대	하루당 비용	₩48,000	수량	4	일	₩192,000	
차량 임대	하루당 비용	₩52,000	수량	6	일	₩312,000	
연료비	리터당 비용	₩1,740	수량	14	리터	₩24,360	
접대	금액	₩130,000				₩130,000	
선물	금액	₩85,000				₩85,000	
기타	금액	₩55,000				₩55,000	
총 출장 비용						₩1,744,360	
예산 미달(초과) 금액						₩155,640	

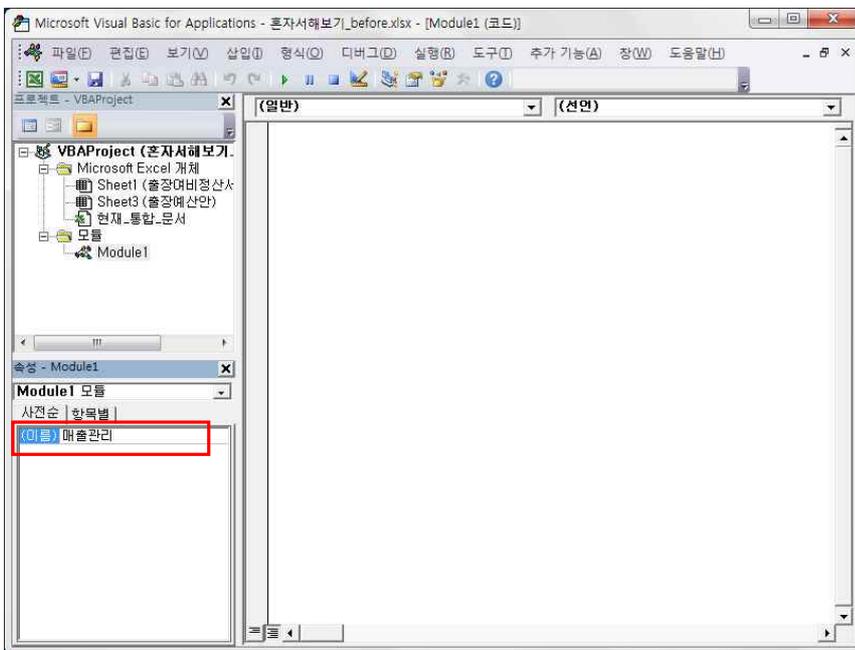
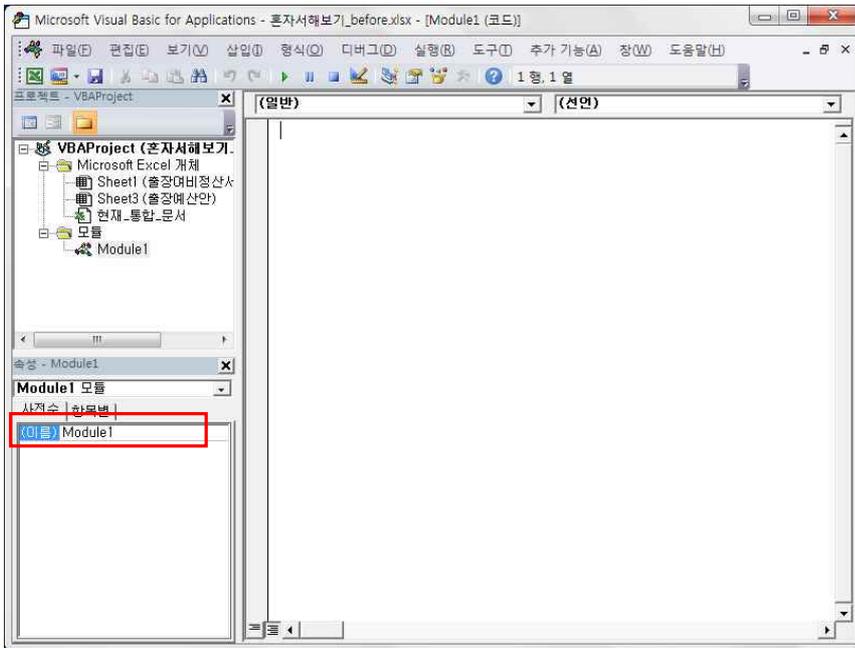
① VB 편집기 창으로 이동하기 위해 [개발 도구] 탭의 [코드] 그룹에서 [Visual Basic]을 클릭합니다.



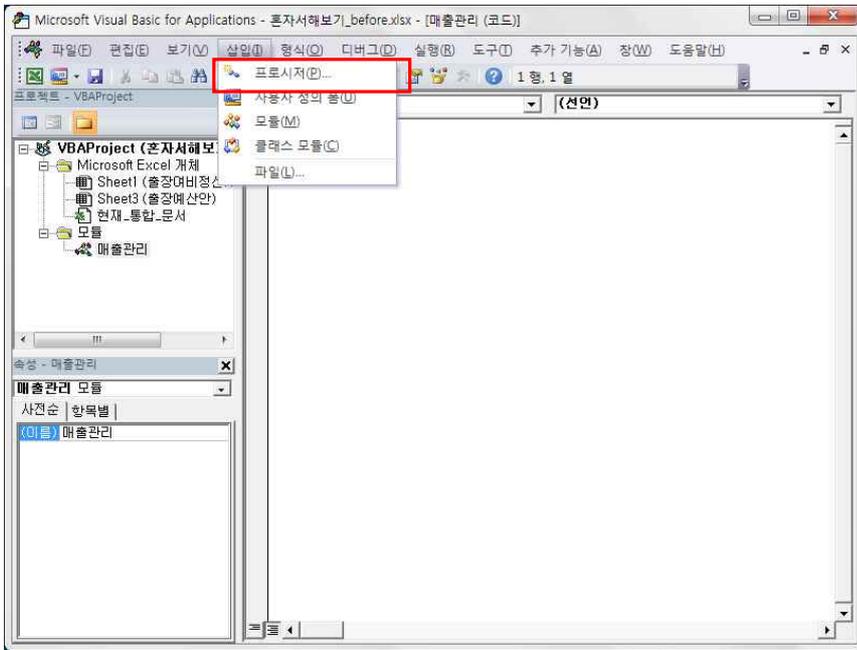
② 새 모듈을 삽입하기 위해 [삽입]의 [모듈] 메뉴를 선택합니다.



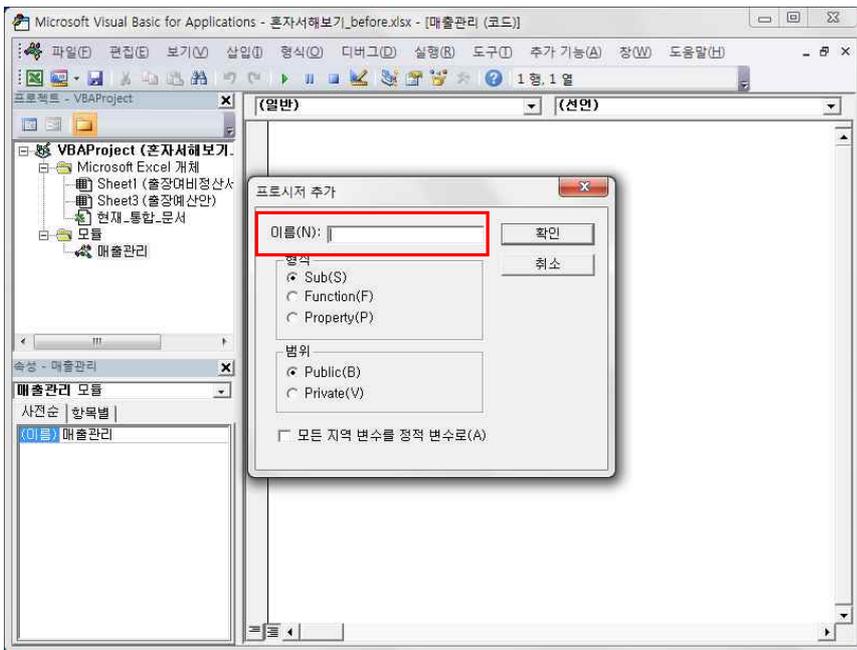
③ 새 모듈이 삽입되고 코드 창이 표시되는 것을 확인한 후 모듈 이름을 변경하기 위해 [속성] 창, [이름] 속성의 'Module1' 을 드래그하여 선택한 후, '매출관리' 를 입력합니다.

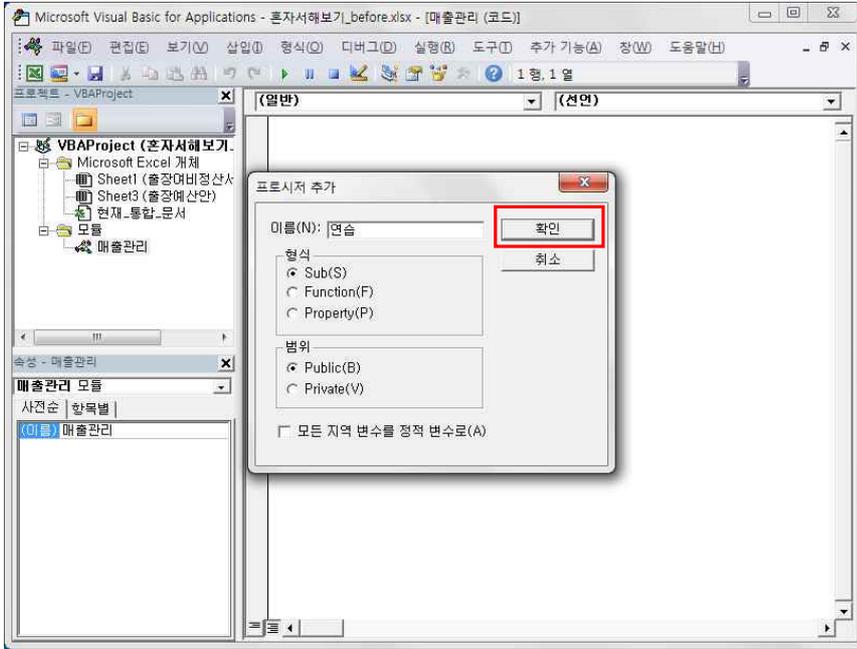


④ 이번엔 Sub Procedure를 삽입하기 위해 [삽입]의 [프로시저] 메뉴를 선택합니다.

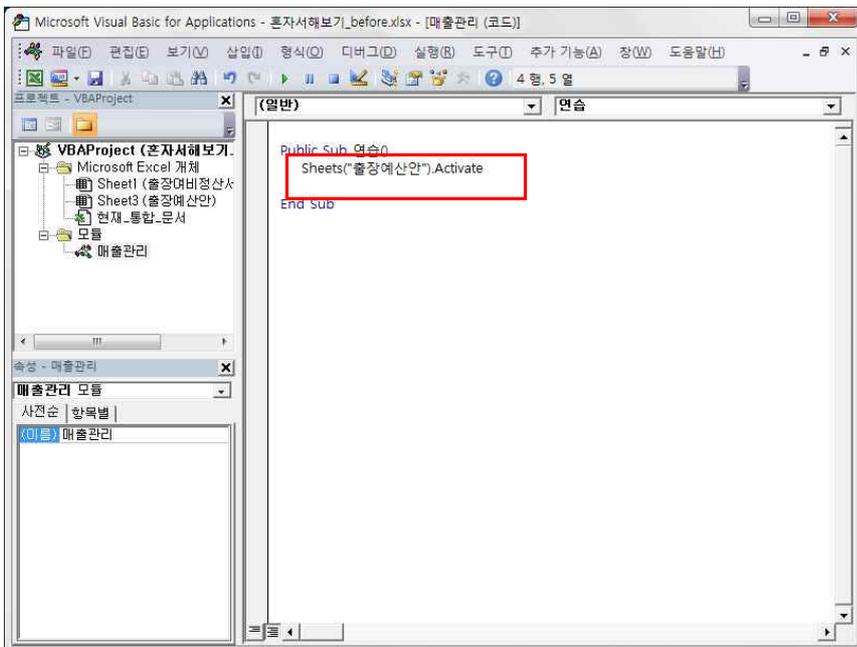


⑤ [프로시저 추가] 창이 표시되면 [이름]에 '연습'을 입력하고 <확인> 단추를 클릭합니다.

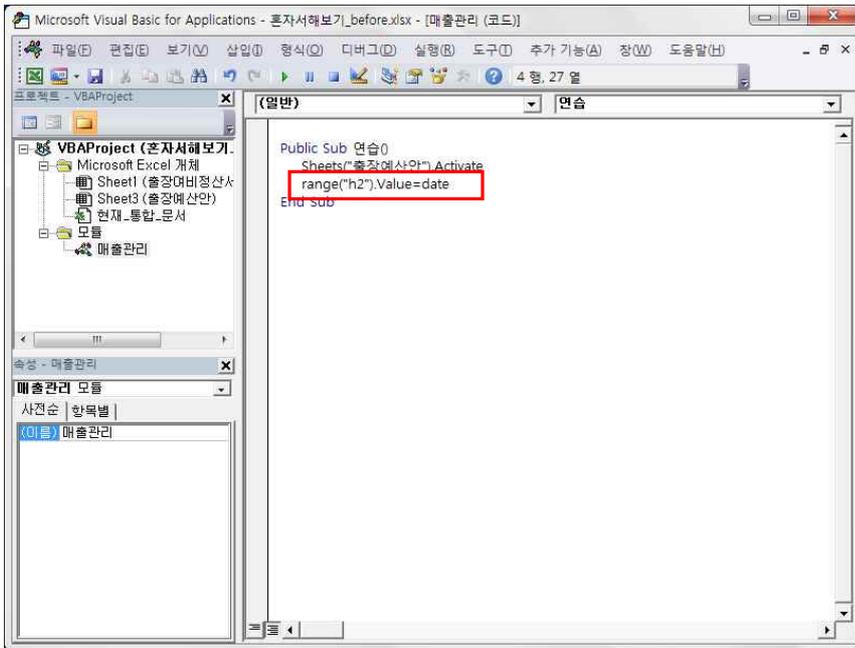




⑥ 삽입된 Sub Procedure에 [출장예산안] 시트를 활성화 시키는 코드 ‘sheets(“출장예산안”). activate’ 를 입력하고, [Enter]키를 누릅니다.



⑦ [H2]셀에 오늘 날짜를 입력하는 코드 'range("h2").value=date' 를 입력하고, [Enter]키를 누릅니다.



⑧ VB 편집기 창의 [닫기] 버튼을 클릭한 후, [출장예산안] 시트가 활성화되어 있고, [H2] 셀에 오늘 날짜가 입력되어 있는 것을 확인합니다.

