

**BIXOLON**

소프트웨어 매뉴얼

# **Label Artist-II**

---

**Ver. 2.10**

<http://www.bixolon.co.kr>

## 목차

저작권 .....	4
1. 운영체제(OS) 환경 .....	5
2. 지원 프린터 .....	5
3. 소프트웨어 설치 준비 .....	6
4. 설치 및 제거 .....	7
4-1 설치 .....	7
4-2 제거 .....	7
5. 사용방법 .....	8
5-1 <파일> 메뉴 .....	8
5-1-1 새로 만들기 .....	8
5-1-2 열기 .....	11
5-1-3 저장 .....	12
5-1-4 인쇄 .....	13
5-2 <편집> 메뉴 .....	15
5-2-1 복사, 잘라내기, 붙여넣기, 삭제 .....	15
5-2-2 실행취소, 다시하기 .....	15
5-2-3 순서 .....	16
5-2-4 맞춤 .....	18
5-2-5 점선/줄자에 맞추기 .....	20
5-3 <객체> 메뉴 .....	21
5-3-1 텍스트 .....	21
5-3-2 멀티라인 텍스트 .....	24
5-3-3 바코드 .....	25
5-3-4 선 .....	28
5-3-5 사각형, 원 .....	29
5-3-6 그림 .....	30
5-3-7 RFID .....	32
5-3-8 기타 .....	34

5-4 <데이터베이스> 메뉴 .....	35
5-4-1 DB 연결하기 .....	35
5-4-2 DB 연결 해제하기 .....	36
5-4-3 객체에 DB 연결하기 .....	36
5-5 템플릿 .....	37
5-5-1 템플릿 설정하기 .....	37
5-5-2 템플릿 내보내기 .....	38
5-6 JSON 파일 저장 .....	40
5-7 <보기> 메뉴 .....	40
<b>6. Label Artist-II SDK .....</b>	<b>41</b>
6-1 변수 설정 .....	42
6-2 C# 프로그래밍 .....	43
6-2-1 LabelArtist 프로퍼티 .....	43
6-2-2 LabelArtist 메소드 .....	45
6-2-3 LabelArtist 활용 예제 .....	49
6-3 C++ 프로그래밍 .....	51
6-3-1 Label Artist-II SDK 메소드 .....	52
6-3-2 Label Artist-II SDK 에러코드 .....	57
6-3-3 Label Artist-II API 활용 예제 .....	58

## 저작권

© BIXOLON Co., Ltd. 모든 권한을 소유합니다.

이 사용설명서와 제품에 사용된 저작물은 저작권법에 의해 보호되어 있습니다.

(주)빅솔론의 사전 서면 동의 없이 사용 설명서 및 제품에 사용된 저작물에 대한 일부 또는 전체를 무단으로 복제, 저장, 전송하는 것을 금합니다.

제공된 정보는 본 제품에만 해당되며 다른 제품에 대해서는 적용되지 않습니다.

또한 본 정보 사용으로 인해 발생하는 직/간접적 손해에 대해 책임지지 않습니다.

- 빅솔론 로고는 (주)빅솔론의 등록상표입니다.
- 모든 다른 상표 또는 제품 이름은 해당하는 회사 또는 조직의 상표입니다.

(주)빅솔론은 제품의 기능과 품질 향상을 위하여 지속적인 개선을 하고 있습니다.

이로 인하여 제품의 사양과 매뉴얼의 내용은 사전 통보 없이 변경될 수 있습니다.

## 사용시 주의사항

프린터와 같은 전자 제품은 정전기에 의해 쉽게 훼손될 수 있습니다.

정전기로부터 프린터를 보호하기 위해서는 프린터 후면 부에 케이블을 연결하거나 제거하기 전에 반드시 프린터 전원을 끄십시오.

만약 프린터가 정전기로부터 손상을 입었을 경우에는 가까운 구입처에 문의하십시오.

## 1. 운영체제(OS) 환경

다음 운영체제(OS)에서 사용 가능합니다.

Microsoft Windows Server 2008 (32bit/64bit)  
Microsoft Windows Server 2008R2 (64bit)  
Microsoft Windows Server 2012 (64bit)  
Microsoft Windows Server 2016 (64bit)  
Microsoft Windows Server 2019 (64bit)  
Microsoft Windows 7 (32bit/64bit)  
Microsoft Windows 8 (32bit/64bit)  
Microsoft Windows 10 (32bit/64bit)

## 2. 지원 프린터

Label Artist-II는 아래 프린터 모델에서 사용 가능합니다.

SLP-TX220/TX223, TX420/TX423, TX400/TX403, TX400R/TX403R  
SLP-DX220/DX223, DX420/DX423  
SLP-DL410/DL413  
SLP-T400/T403  
SLP-D220/D223, D420/D423  
SRP-770II/770III/E770III  
SPP-L3000/L310/L410  
XT5-40/43/46  
XD3-40d  
XD3-40t  
XL5-40CT/43CT  
XD5-40d/43d  
XD5-40t/43t  
XM7-20/40R  
SRP-S3000  
XT3-40/43  
BDP-T42/T43/D42/D43, BTP-42/43/46

### **3. 소프트웨어 설치 준비**

Label Artist-II의 최신버전은 당사 홈페이지에서 다운로드 받을 수 있습니다.  
([www.bixolon.co.kr](http://www.bixolon.co.kr))

## 4. 설치 및 제거

### 4-1 설치

- 1) 설치파일(Label Artist-II\_Vx.x.x.exe)을 더블 클릭하십시오.  
※ 설치파일을 실행하기 위해서는 관리자 권한 (Administrator Privilege)가 필요할 수도 있습니다.
- 2) 스크린에 보이는 지시사항에 따라서 설치를 계속 진행하십시오.



프로그램 설치 중 필수 항목인 “Microsoft Visual C++ 2008 SP1 Redistributable Package (x86)”의 설치를 요구할 수 있으며, 팝업 창이 보이면 설치 버튼을 클릭해 주십시오.


### 4-2 제거

- 1) 제어판에서 “프로그램 제거” 메뉴로 이동하십시오.
- 2) Label Artist-II을 선택하고 “제거” 버튼을 클릭하면, PC에서 설치된 프로그램을 쉽게 제거할 수 있습니다.

## 5. 사용방법

### 5-1 <파일> 메뉴

#### 5-1-1 새로 만들기

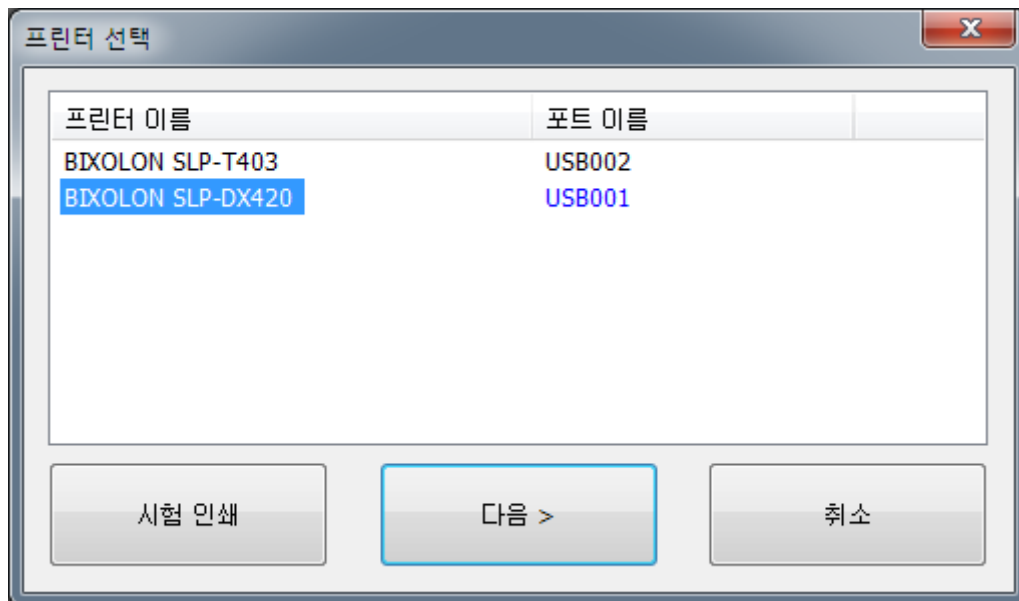
1) <파일> 메뉴에서 <새로 만들기 > 항목을 선택합니다.



Label Artist-II 프로그램을 사용하기 위해서는 반드시 프린터 드라이버가 설치되어 있어야 합니다.

2) 프린터를 선택할 수 있는 다이얼로그가 아래와 같이 나타납니다.

사용할 프린터를 선택하고 <다음> 버튼을 클릭합니다.



3) 아래와 같이 <인쇄 기본 설정> 화면이 나타납니다.

라벨 사이즈 등의 인쇄에 필요한 옵션을 선택하고 <확인> 버튼을 클릭합니다.

The image displays three screenshots of the '인쇄 기본 설정' (Print Basic Settings) dialog box in the Label Artist-II software.

- Top Left Screenshot:** Shows the '기본 설정' (Basic Settings) tab. Options include:
  - 인쇄 속도 (Print Speed): 5 (인치/초)
  - 인쇄 농도 (Print Density): 14
  - 종이 크기 (단위: mm) (Paper Size (unit: mm)): 너비 101.6, 높이 152.4
  - 여백 (단위: mm) (Margin (unit: mm)): 왼쪽 0.00, 오른쪽 0.00, 위 0.00, 아래 0.00
  - 인쇄 방향 (Print Direction): ☒ 회전 안함, ☐ 180도 회전
  - 여러 열(列)을 가지는 라벨 용지 (Label paper with multiple columns): 열(列) 1, 가로 간격 0.0 mm
  - 종이 절단 (Paper Cutting): 안함, 간격: 1
- Top Right Screenshot:** Shows the '레이아웃' (Layout) tab. Options include:
  - 인쇄 방법 (Print Method): ☒ 직렬 열전사, ☐ 열전사 전송
  - 종이 유형 (Paper Type): ☒ 간격, ☐ 연속
  - 위치 조정 (Position Adjustment):
    - 겹/불력마크 길이 (Overlap/Blank mark length): 3.00 mm
    - 겹/불력마크 오프셋 (Overlap/Blank mark offset): 0.00 mm
    - 절단/커터 오프셋 (Cut/Offset): 0.00 mm
  - ☒ 압축하여 데이터 전송 (Compress and transmit data), ☐ 되감기 (Rewind)
  - 코드 페이지 (Code Page): ☐ 코드 페이지 사용, CP437
- Bottom Screenshot:** Shows the '레이아웃' (Layout) tab. Options include:
  - 라벨 모양 (Label Shape): ☐ 직사각형, ☒ 둥근 직사각형, ☐ 타원형. 모서리 반경 (Corner radius): 3.18 mm
  - 색상 (Color): 배경 색 (Background color): ☐ None, ☒ 인쇄 안함 (Don't print)

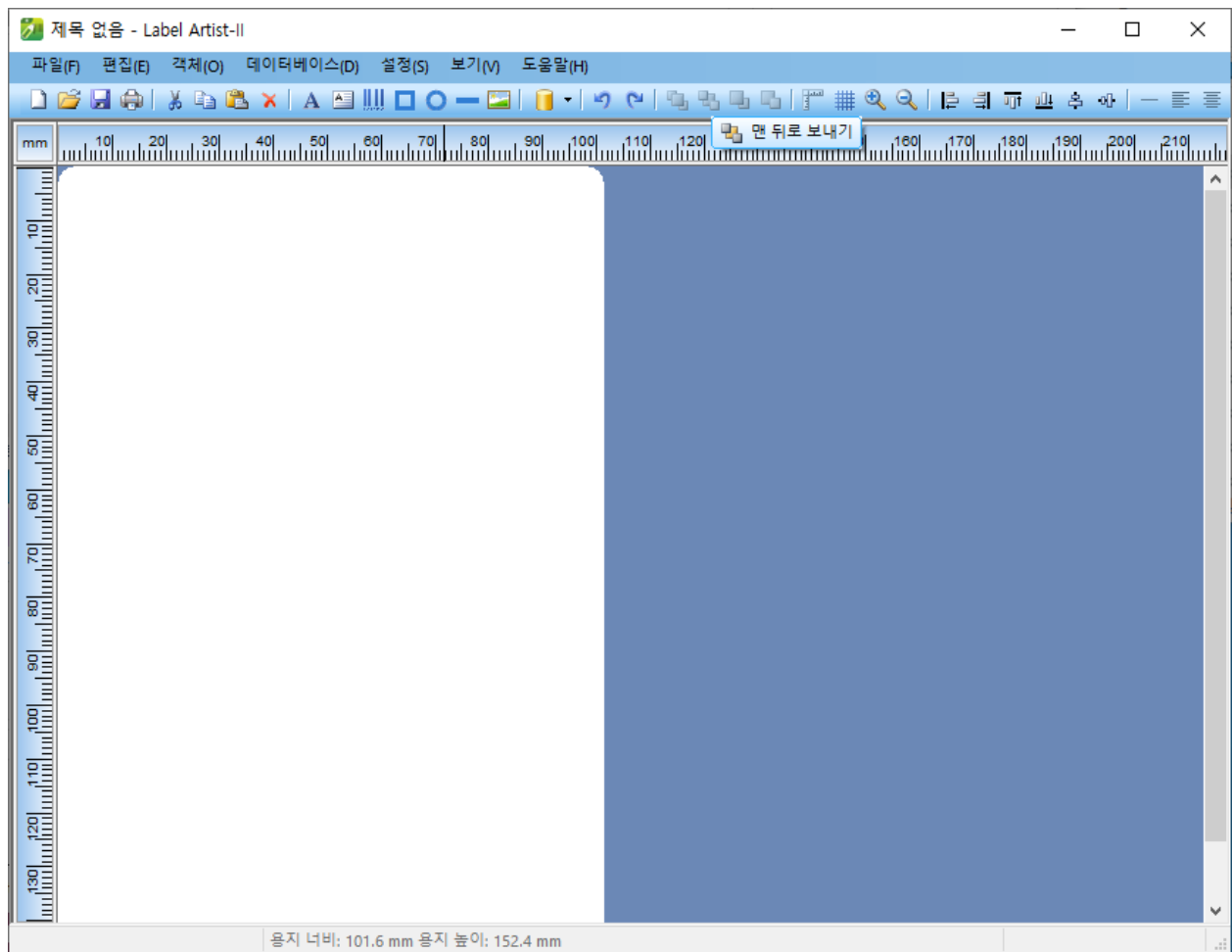
## 코드 페이지

코드 페이지는 프린터 폰트 중 ASCII 폰트 인쇄 시 적용됩니다.




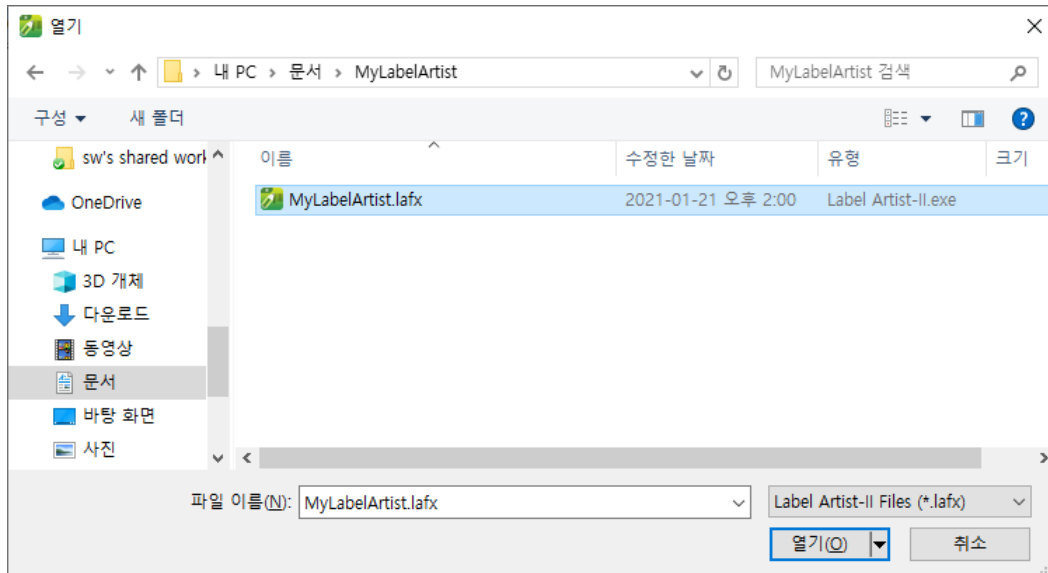
The image shows a screenshot of the '글꼴' (Font) dialog box. The '프린터 폰트' (Printer Font) radio button is selected. In the '종류' (Type) dropdown menu, 'ASCII' is selected and highlighted with a red box.

4) 설정이 완료되면 아래와 같이 새로운 라벨이 생성되어 보여집니다.

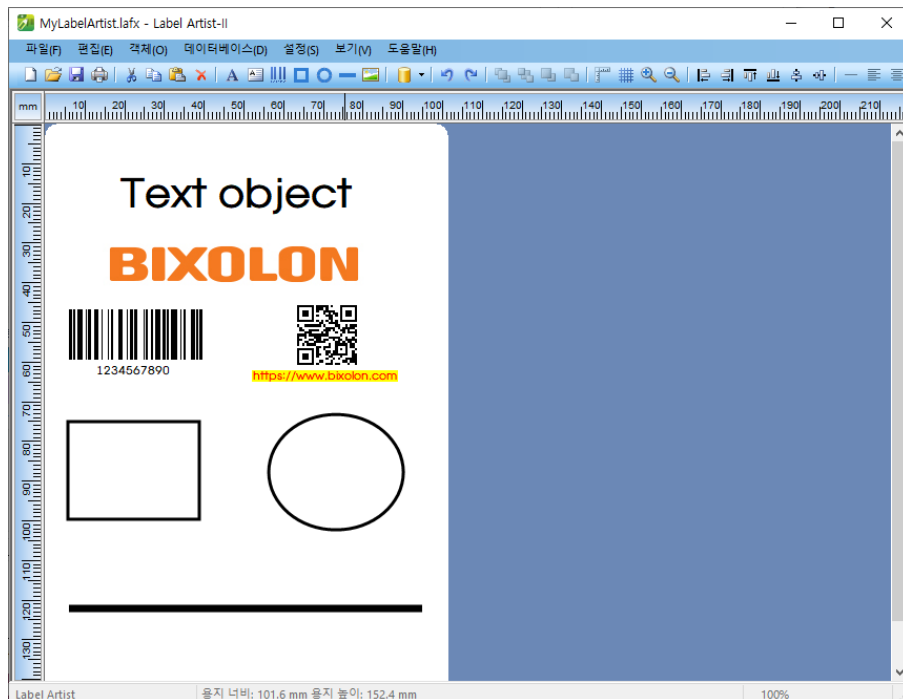


## 5-1-2 열기


- 1) <파일> 메뉴에서 <열기  > 항목을 선택합니다.
- 2) “열기” 다이얼로그가 아래와 같이 보여지면 이전에 저장했던 파일(\*.lafx)중 하나를 선택하고 [열기]버튼을 클릭합니다.

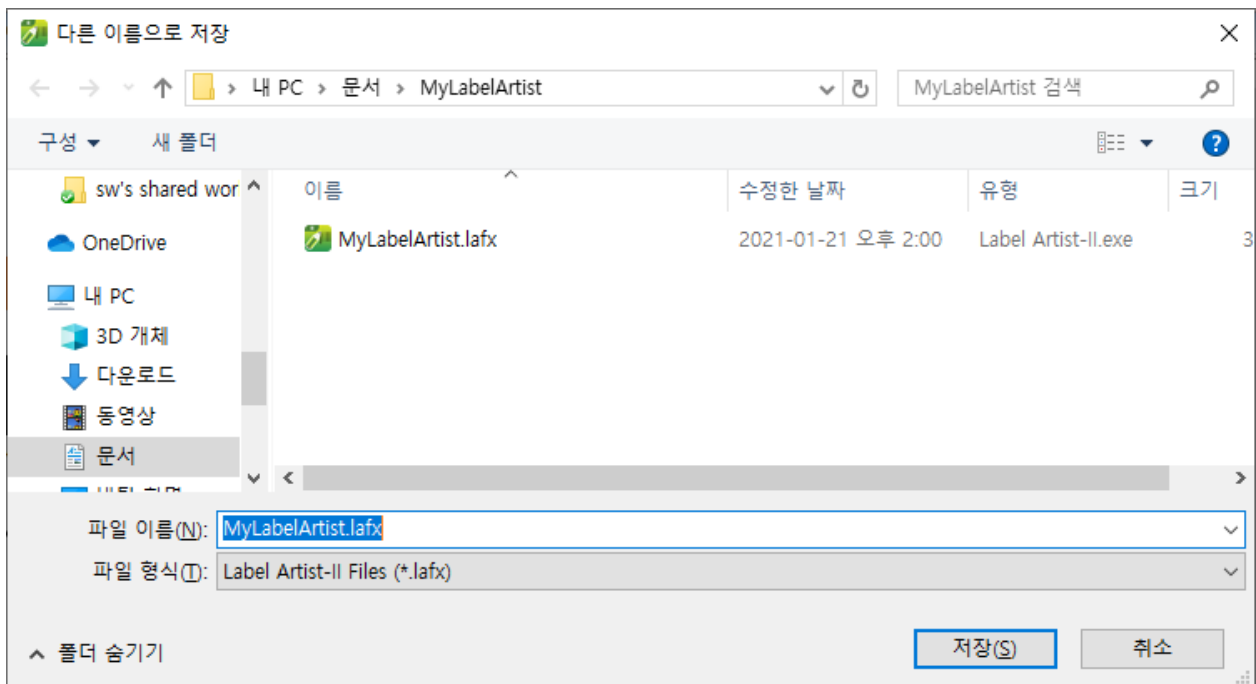


- 3) 선택된 파일에 저장되어 있던 라벨이 아래와 같이 표시됩니다.




### 5-1-3 저장

- 1) <파일> 메뉴에서 <저장 > 항목을 선택하면 현재까지 작업한 라벨의 정보가 저장됩니다.
- 2) <다른 이름으로 저장> 항목을 선택하면 아래와 같이 다이얼로그가 나타납니다.  
저장할 위치와 파일 이름을 입력하고 [저장] 버튼을 누르면 현재까지 작업한 라벨의 정보가 새 이름으로 저장됩니다.



## 5-1-4 인쇄

- 1) <파일> 메뉴에서 <인쇄 > 항목을 선택합니다.
- 2) 인쇄 설정 화면이 아래와 같이 나타나면, 인쇄 매수를 입력하고 [라벨 인쇄] 버튼을 클릭하면 인쇄를 시작합니다.



- 사본: 동일한 페이지의 인쇄 매수를 설정합니다.
- 데이터베이스 필드 사용: 데이터베이스에 연결된 객체 인쇄 시 사용할 수 있으며, 인쇄 매수를 선택한 필드 기준으로 설정합니다.
- 시리얼 넘버: 오토카운터 사용시 인쇄 범위를 설정합니다.

### 한 부씩 인쇄 옵션



한 부씩 인쇄는 복사본 인쇄 시 사용하는 옵션으로 라벨 전체를 모아서 인쇄하는 방식입니다. 한 부씩 인쇄 옵션을 해제하면 라벨 단위로 복사본이 인쇄됩니다.

- 3) 데이터베이스에 연결된 객체가 있다면 레코드 셋을 설정하여 데이터베이스의 내용을 인쇄할 수 있습니다.
- 모두 인쇄: 모든 행(Row)에 대하여 인쇄합니다.
  - 선택한 것만 인쇄: 선택된 행(Row)에 대하여 인쇄합니다.

### 5-2 <편집> 메뉴

#### 5-2-1 복사, 잘라내기, 붙여넣기, 삭제

##### 1) 복사 (Ctrl + C)

선택된 객체(텍스트, 바코드, 라인 등)을 복사합니다.

##### 2) 잘라내기 (Ctrl + X)

선택된 객체(텍스트, 바코드, 라인 등)을 잘라내기 합니다.

##### 3) 붙여넣기 (Ctrl + V)

복사 또는 잘라내기를 실행한 객체를 붙여 넣습니다.

##### 4) 삭제 (Delete)

선택된 객체를 삭제합니다.

#### 5-2-2 실행취소, 다시하기

##### 1) 실행취소 (Ctrl + Z)

마지막에 라벨에 디자인했던 내용을 취소합니다.

##### 2) 다시하기 (Ctrl + Y)

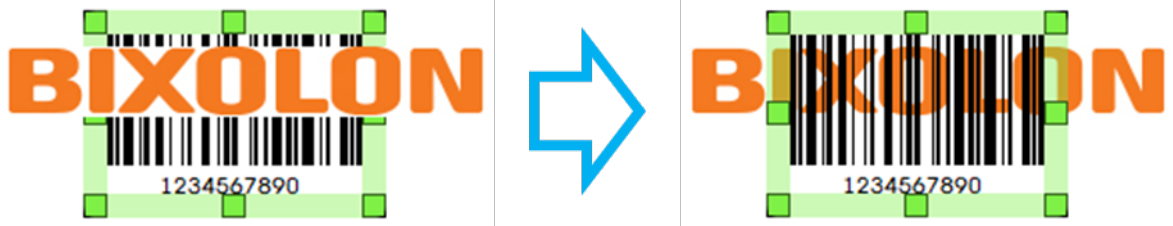
최근에 [실행취소] 했던 내용을 복구합니다.

**5-2-3 순서**

객체가 그려지는 순서를 나타냅니다. 순서에 따라서 뒤에 그려지는 객체가 숨겨질 수 있습니다.

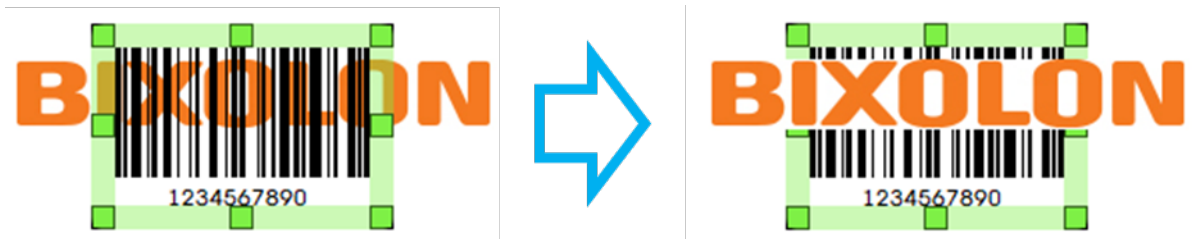
**1) 앞으로 가져오기** 

선택된 객체와 바로 위에 있는 객체의 순서를 바꿉니다.



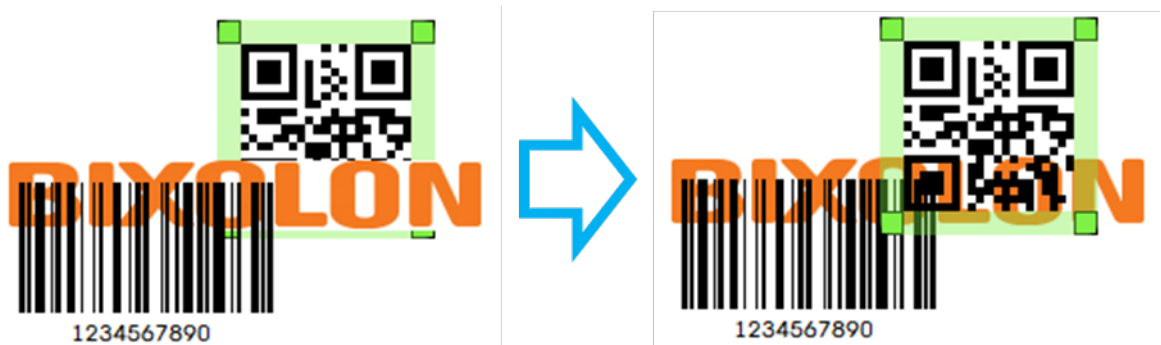
**2) 뒤로 보내기** 

선택된 객체와 바로 아래에 있는 객체의 순서를 바꿉니다.



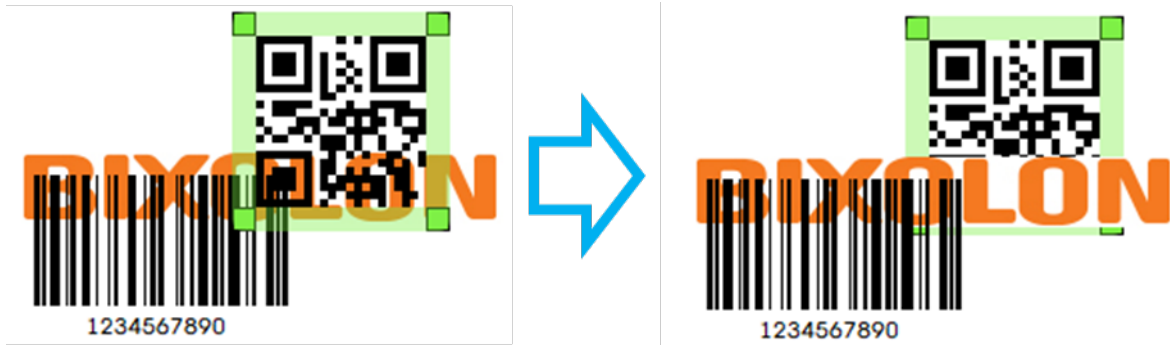
**3) 맨 앞으로 가져오기** 

선택된 객체를 다른 모든 객체들 위에 그리도록 합니다.



### 4) 맨 뒤로 보내기

선택된 객체를 다른 모든 객체들 아래에 그리도록 합니다.



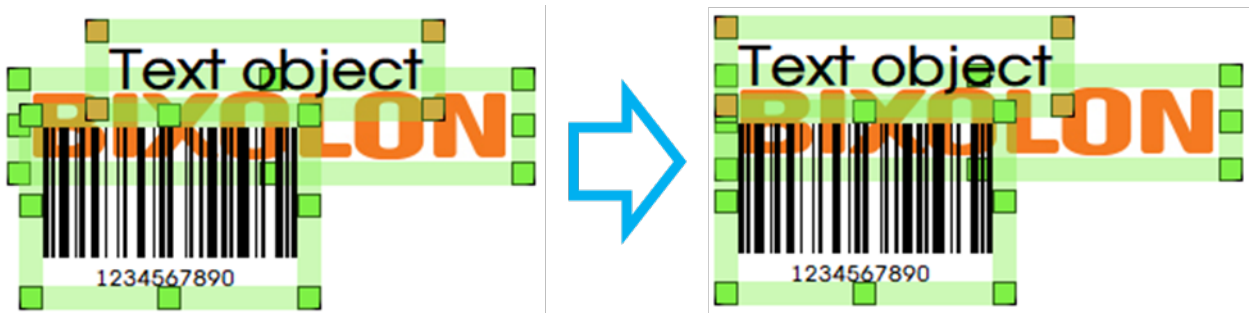
## 5-2-4 맞춤

하나의 정해진 객체를 기준으로 기준선을 맞추는 기능입니다.

컨트롤(Ctrl) 키를 누르면 여러 개의 객체를 선택할 수 있으며, 그 중에 맨 마지막에 선택된 객체가 기준이 됩니다.

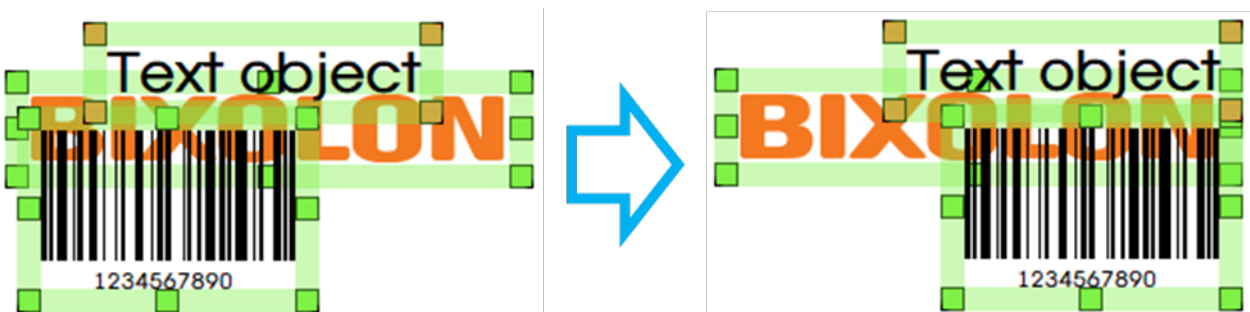
### 1) 왼쪽 맞춤

기준의 왼쪽 좌표와 같은 위치로 이동합니다.



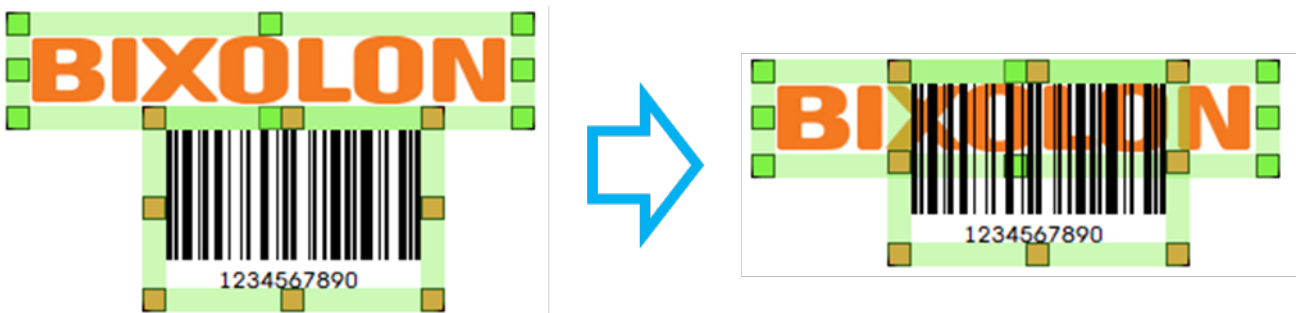
### 2) 오른쪽 맞춤

기준의 오른쪽 좌표와 같은 위치로 이동합니다.



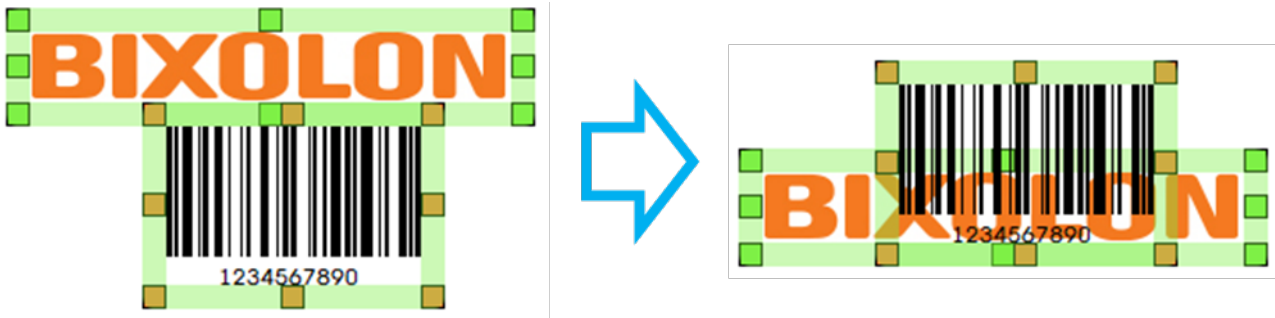
### 3) 위쪽 맞춤

기준의 위쪽 좌표와 같은 위치로 이동합니다.



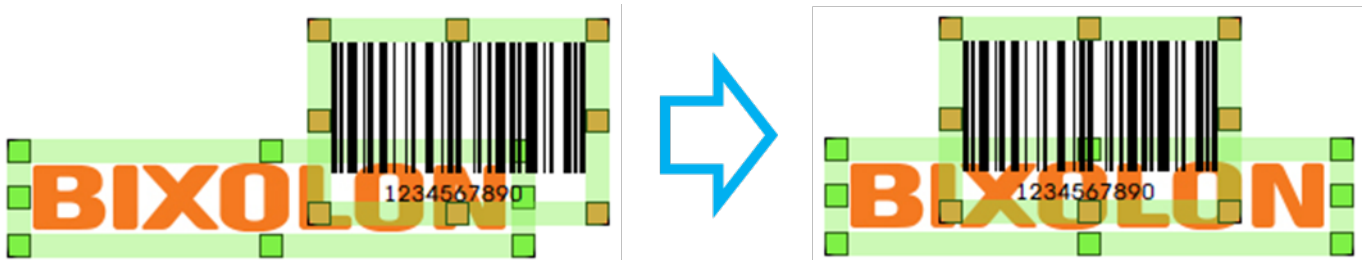
### 4) 아래쪽 맞춤

기준의 아래쪽 좌표와 같은 위치로 이동합니다.



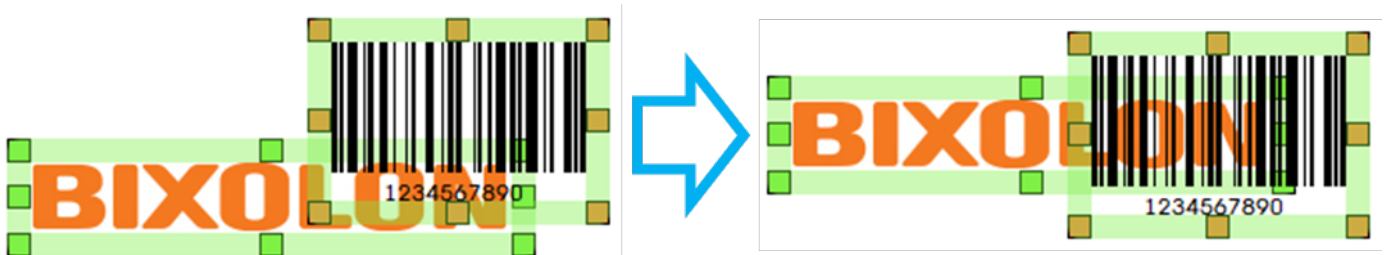
### 5) 가로 가운데 맞춤

각 객체의 가로방향 중심 좌표를 기준의 가로방향 중심 좌표로 이동합니다.



### 6) 세로 가운데 맞춤

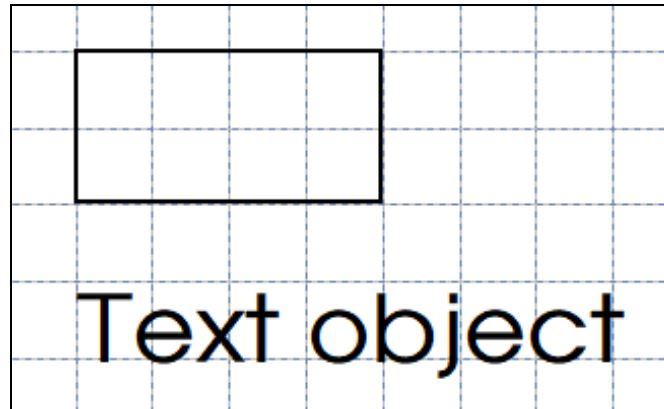
각 객체의 세로방향 중심 좌표를 기준의 세로방향 중심 좌표로 이동합니다.



5-2-5 점선/줄자에 맞추기

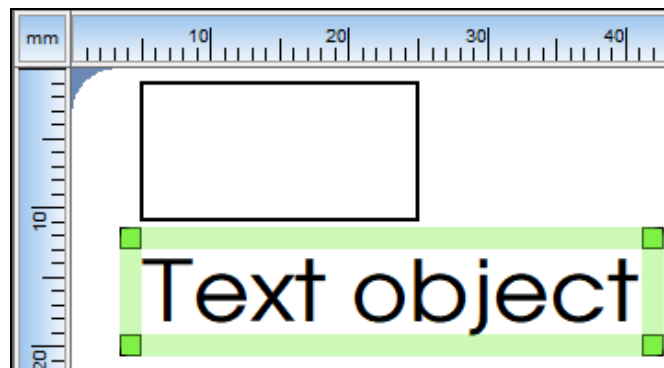
1) 점선에 맞추기

객체 이동 및 크기 조정 시 점선을 기준으로 변경됩니다.



2) 줄자에 맞추기

객체 이동 및 크기 조정 시 줄자를 기준으로 변경됩니다.



## 5-3 <객체> 메뉴

- 객체(텍스트, 바코드, 선 등)를 선택하여 디자인할 수 있습니다.

### 5-3-1 텍스트

- 1) <객체> 메뉴의 <텍스트 A> 항목을 선택한 후 라벨에 마우스 왼쪽 버튼을 누른 상태로 드래그합니다. 기본 속성의 텍스트가 화면에 그려집니다.
- 2) 그려진 텍스트 객체를 선택한 후 마우스 왼쪽 버튼을 연속으로 두 번 클릭하면 텍스트 객체에 대한 속성을 변경할 수 있는 다이얼로그가 나타납니다.

텍스트 속성

텍스트

Text object

☐ 데이터베이스 연결

글꼴

텍스트 양식 위치 템플릿 데이터

☐ 트루 타입 폰트 ☒ 프린터 폰트 크기: 8

종류: KS5601

유형

☐ 굵게 ☐ 기울임 ☐ 밑줄

색상

폰트 색:   배경 색: ☐ None

☐ 잠금 ☐ 인쇄 안함

확인 취소

### 3) 사용할 글꼴 속성을 설정합니다.

#### 프린터 폰트

프린터에 내장된 폰트를 선택하여 출력하는 방법으로 아래의 폰트 종류를 선택할 수 있습니다.



ASCII - 영어, 숫자, 특수 기호 및 코드페이지

KS5601 - 한글

BIG5 - 중문(번체)

GB2312 - 중문(간체)

Shift-JIS - 일어

OCR-A - 영어, 숫자, 특수 기호

OCR-B - 영어, 숫자, 특수 기호

### 4) 텍스트 양식을 설정합니다.

- 텍스트: 단일 줄의 텍스트 입력
- 멀티라인 텍스트: 여러 줄의 텍스트 입력

글꼴	텍스트 양식	위치	템플릿	데이터
<p>유형: <input checked="" type="radio"/> 텍스트 <input type="radio"/> 멀티라인 텍스트</p> <p>정렬: <span>왼쪽</span> ▼</p>				



텍스트 유형 변경 시 멀티라인 텍스트로 전환이 가능합니다.

5) 위치를 설정합니다.

글꼴

텍스트 양식

위치

템플릿

데이터

회전

☒ 회전 안함
 ☐ 90°
 ☐ 180°
 ☐ 270°

위치

X: 20.90 mm

Y: 28.05 mm

객체 참조 점 :

위 왼쪽

#### 객체 참조 점



객체의 회전 및 데이터 변경(오토 카운터, 데이터베이스 연결 등)으로 인한 이동 및 크기 변경 시 기준이 되는 위치

6) 템플릿 사용

“프린터 폰트”를 선택하면 템플릿을 사용할 수 있습니다.

템플릿 관련 자세한 내용은 5-5 템플릿을 참고 바랍니다.

## 7) 오토 카운터 사용

“오토 카운터 사용”을 체크하면 아래와 같이 설정화면 나타납니다.

“오토 카운터”를 적용하면 텍스트는 시작 값으로 변경됩니다.



[마지막 카운터 값] 적용을 체크하면 인쇄가 완료된 후 마지막에 인쇄된 카운터의 다음 값으로 [시작 값]이 변경됩니다.


## 5-3-2 멀티라인 텍스트

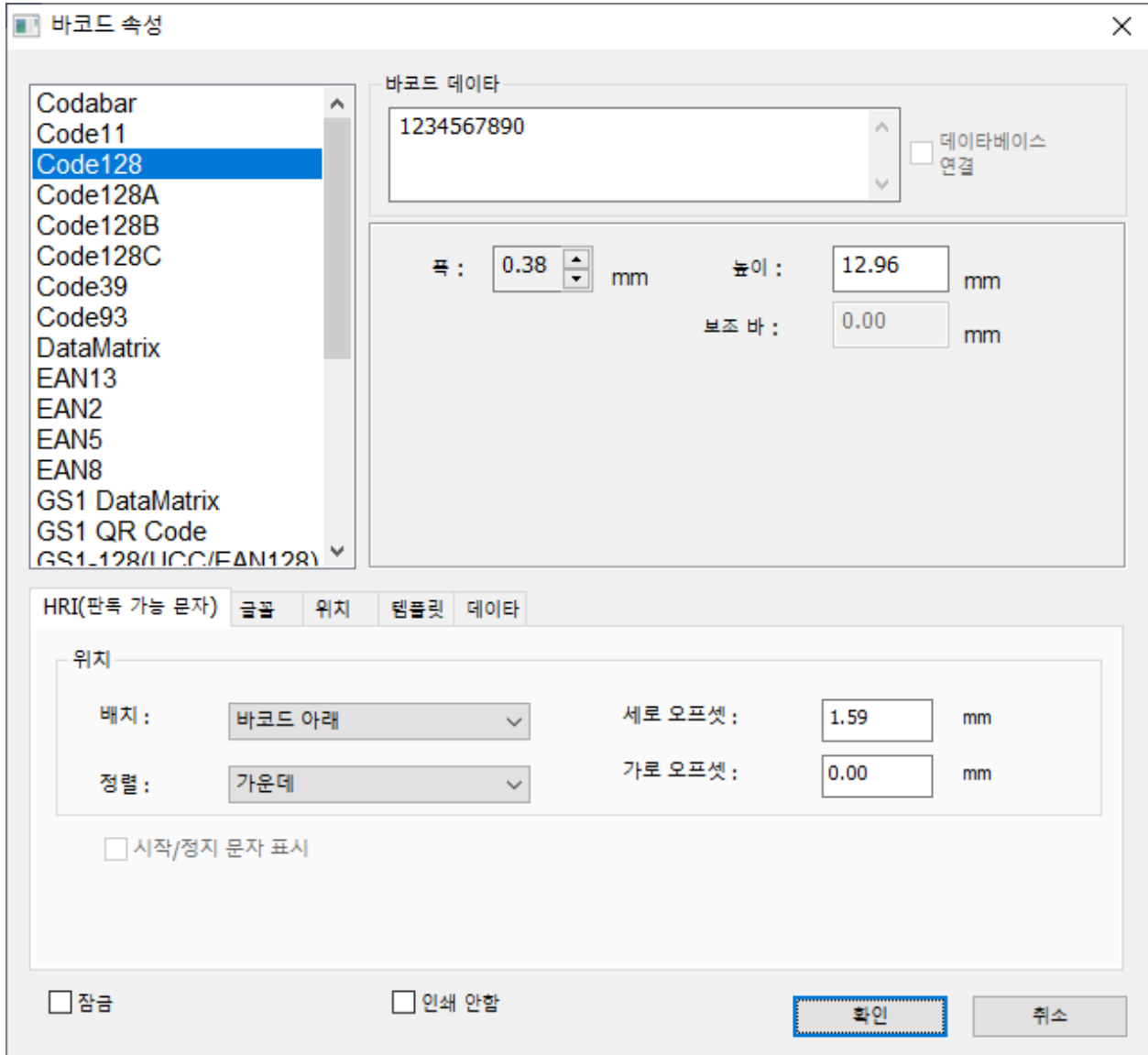
여러 줄의 텍스트를 입력할 수 있는 객체입니다.



텍스트 유형 변경 시 단일 줄 텍스트로 전환이 가능합니다.

### 5-3-3 바코드

- 1) <객체> 메뉴의 <바코드> 항목을 선택한 후 라벨에 마우스 왼쪽 버튼을 누른 상태로 드래그합니다. 기본 속성의 바코드가 화면에 그려집니다.
- 2) 그려진 텍스트 객체를 선택한 후 마우스 왼쪽 버튼을 연속으로 두 번 클릭하면 객체에 대한 속성을 변경할 수 있는 다이얼로그가 나타납니다.



The dialog box titled '바코드 속성' (Barcode Properties) contains the following elements:

- Barcode Type List:** A list on the left with 'Code128' selected. Other options include Codabar, Code11, Code128A, Code128B, Code128C, Code39, Code93, DataMatrix, EAN13, EAN2, EAN5, EAN8, GS1 DataMatrix, GS1 QR Code, and GS1-128(UCC/EAN128).
- Barcode Data:** A text field containing '1234567890' and a '데이터베이스 연결' (Connect to Database) checkbox.
- Dimensions:** Fields for '폭' (Width: 0.38 mm), '높이' (Height: 12.96 mm), and '보조 바' (Auxiliary Bar: 0.00 mm).
- HRI (판독 가능 문자) Tab:**
  - Sub-tabs: 글꼴, 위치, 템플릿, 데이터.
  - 위치 (Position) Sub-tab:**
    - '배치' (Layout): '바코드 아래' (Below barcode).
    - '정렬' (Alignment): '가운데' (Center).
    - '세로 오프셋' (Vertical Offset): 1.59 mm.
    - '가로 오프셋' (Horizontal Offset): 0.00 mm.
    - Checkbox: '시작/정지 문자 표시' (Show start/stop characters).
- Bottom Controls:** Checkboxes for '잠금' (Lock) and '인쇄 안함' (Do not print), and buttons for '확인' (OK) and '취소' (Cancel).

- 3) 사용할 바코드 종류와 속성을 변경합니다.
- 4) HRI(판독 가능 문자)를 설정합니다.

### 5) GS1 바코드 설정

GS1 바코드의 데이터는 GS1 탭에서 설정합니다.

HRI(판독 가능 문자) 글꼴 위치 템플릿 데이터 GS1

데이터 원본

추가

AI:  Data(GTIN):

NO.	AI	DATA(GTIN)

삭제

AI와 데이터를 입력 후 추가 버튼을 클릭합니다.

리스트의 항목을 더블클릭 하면 아이템을 변경할 수 있습니다.

HRI(판독 가능 문자) 글꼴 위치 템플릿 데이터 GS1

데이터 원본

212020102800001 추가

AI:  Data(GTIN):

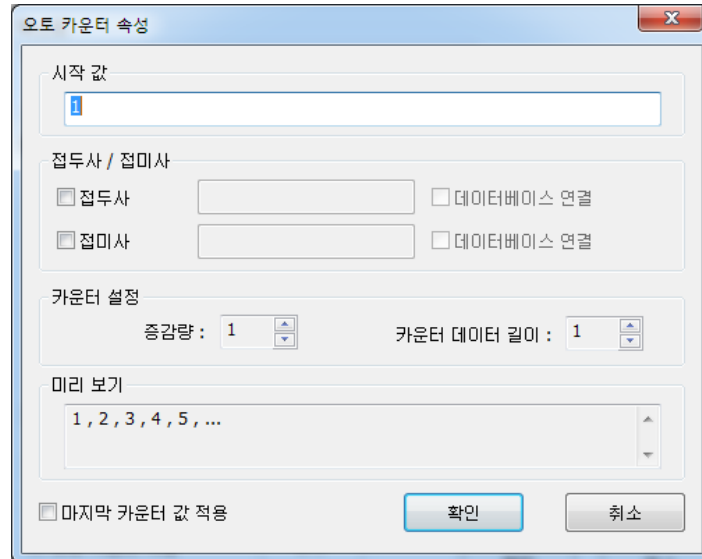
NO.	AI	DATA(GTIN)
1	12	201028
2	21	2020102800001

삭제

## 6) 오토 카운터 사용

“오토 카운터 사용”을 체크하면 아래와 같이 설정화면 나타납니다.

“오토 카운터”를 적용하면 텍스트는 임의의 데이터로 변경됩니다.



오토 카운터 속성

시작 값

접두사 / 접미사

☐ 접두사  ☐ 데이터베이스 연결

☐ 접미사  ☐ 데이터베이스 연결

카운터 설정

증감량 : 1

카운터 데이터 길이 : 1

미리 보기

1, 2, 3, 4, 5, ...

☐ 마지막 카운터 값 적용

확인 취소




[마지막 카운터 값] 적용을 체크하면 인쇄가 완료된 후 마지막에 인쇄된 카운터의 다음 값으로 [시작 값]이 변경됩니다.

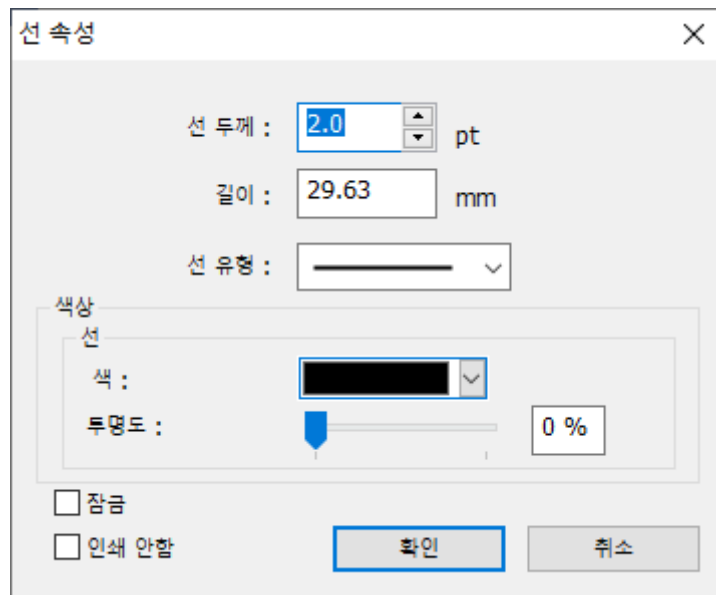
## 7) 템플릿 사용

“1D 바코드”를 선택하면 템플릿을 사용할 수 있습니다.

템플릿 관련 자세한 내용은 5-5 템플릿을 참고 바랍니다.



**5-3-4 선**

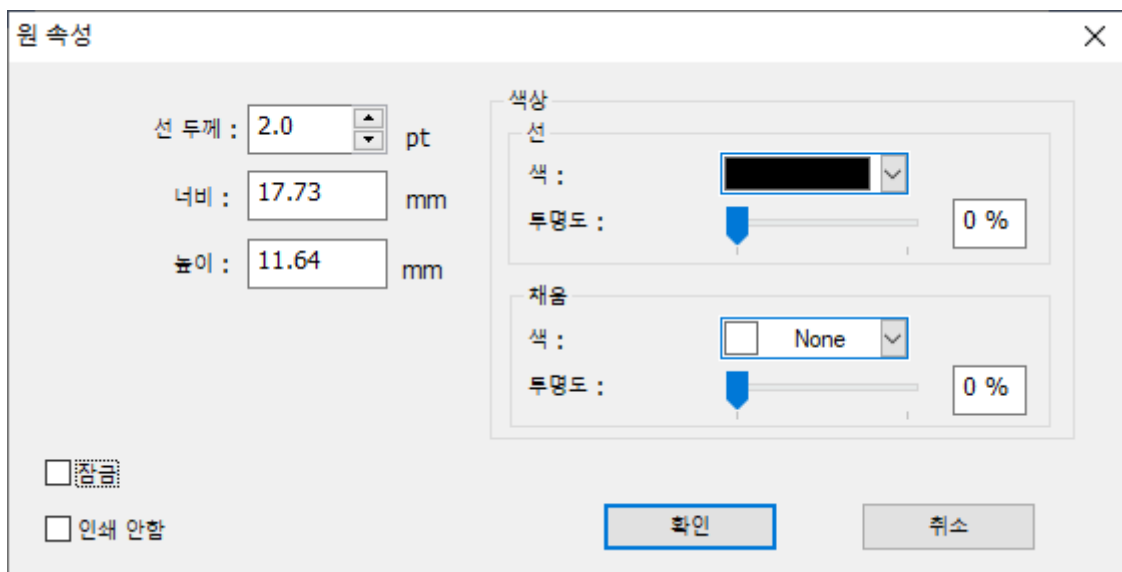
- 1) <객체> 메뉴의 <선> 항목을 선택한 후 라벨에 마우스 왼쪽 버튼을 누른 상태로 드래그하면 선이 화면에 그려집니다.
- 2) 쉬프트(Shift) 키를 누른 상태에서 드래그하면, 대각선이나 직선 방향으로 그릴 수 있습니다.
- 3) 그려진 객체를 선택한 후 마우스 왼쪽 버튼을 연속으로 두 번 클릭하면 객체에 대한 속성을 변경할 수 있는 다이얼로그가 나타납니다.



- 4) 사용할 선 종류와 속성을 입력하고 [확인]을 누르면 입력한 내용이 적용됩니다


**5-3-5 사각형, 원**

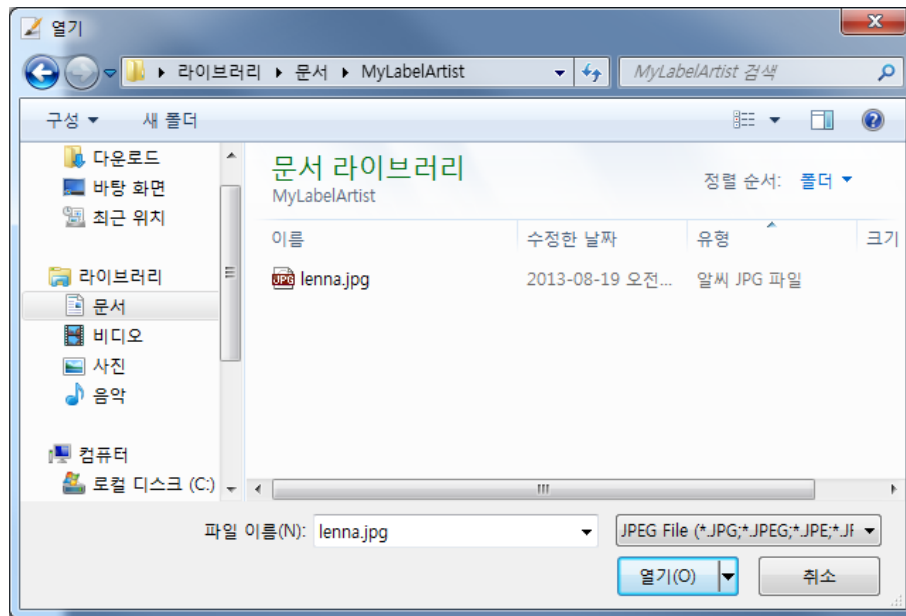
- 1) <객체> 메뉴의 <사각형 > 또는 <원 > 항목을 선택한 후 라벨에 마우스 왼쪽 버튼을 누른 상태로 드래그하면 선이 화면에 그려집니다.
- 2) 쉬프트(Shift) 키를 누른 상태에서 드래그하면, 가로, 세로 길이가 같은 객체로 그릴 수 있습니다.
- 3) 그려진 객체를 선택한 후 마우스 왼쪽 버튼을 연속으로 두 번 클릭하면 객체에 대한 속성을 변경할 수 있는 다이얼로그가 나타납니다.



- 4) 사용할 선 종류와 속성을 입력하고 [확인]을 누르면 입력한 내용이 적용됩니다

### 5-3-6 그림

- 1) <객체> 메뉴의 <그림  > 항목을 선택하면 아래와 같이 그림 파일을 선택할 수 있는 다이얼로가 나타납니다.



- 2) 원하는 그림 파일을 선택하고 “열기” 버튼을 클릭한 후, 라벨에 마우스 왼쪽 버튼을 누른 상태로 드래그하면 그림이 화면에 그려집니다.

- 3) 그려진 객체를 선택한 후 마우스 왼쪽 버튼을 연속으로 두 번 클릭하면 객체에 대한 속성을 변경할 수 있는 다이얼로그가 나타납니다.

### 그림 원본



- 포함된 그림: 그림 로드 시 라벨 디자인 파일에 그림이 포함되어 라벨 인쇄 시 별도의 참조 없이 인쇄됨
- 외부 그림 파일: 라벨 디자인 파일에 그림의 경로가 포함되어 라벨 인쇄 시 해당 경로의 그림이 참조되어 인쇄됨

- 4) 사용할 종류와 속성을 입력하고 [확인]을 누르면 입력한 내용이 적용됩니다.

## 5-3-7 RFID

1) <객체> 메뉴의 <RFID> 항목을 선택하면 아래와 같이 RFID 객체가 생성됩니다.



2) 생성된 객체를 선택한 후 마우스 왼쪽 버튼을 연속으로 두 번 클릭하면 객체에 대한 속성을 변경할 수 있는 다이얼로그가 나타납니다.

RFID 속성

테그 유형 :

EPC(class 1) Gen2

인코더 설정

데이터 위치

위치 :

EPC - 전자 제품 코드

시작 블록 :

4

데이터베이스 연결

오토 카운터 사용

설정

테그 데이터

데이터 유형 :

16진수

데이터 :

1234ABCD

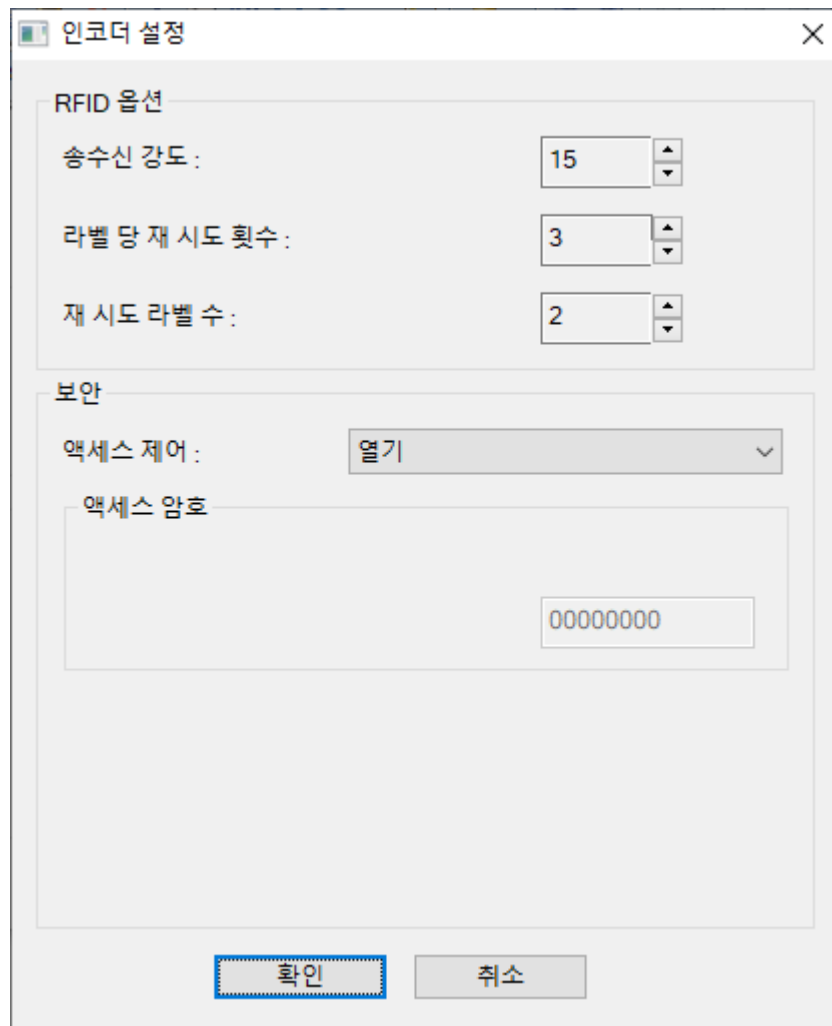
잠금

인쇄 안함

확인

취소

3) 인코더 설정 버튼을 클릭하면 인코더 옵션을 선택할 수 있습니다.



The dialog box titled '인코더 설정' (Encoder Setting) contains two sections: 'RFID 옵션' (RFID Option) and '보안' (Security). The 'RFID 옵션' section has three spinners: '송수신 강도' (Transmit/Receive Strength) set to 15, '라벨 당 재 시도 횟수' (Retries per Label) set to 3, and '재 시도 라벨 수' (Number of Retries Label) set to 2. The '보안' section has a dropdown for '액세스 제어' (Access Control) set to '열기' (Open) and a text field for '액세스 암호' (Access Password) containing '00000000'. At the bottom are '확인' (OK) and '취소' (Cancel) buttons.

Section	Parameter	Value
RFID 옵션	송수신 강도	15
	라벨 당 재 시도 횟수	3
	재 시도 라벨 수	2
보안	액세스 제어	열기
	액세스 암호	00000000

RFID 객체는 RFID를 지원하는 프린터 사용 시 사용할 수 있습니다.



RFID 지원 모델

: SLP-T400R/T403R, SLP-TX400R/TX403R, XT5-40NR/43NR/46NR,  
XD5-40tR/43tR

**5-3-8 기타**

잠금: 각 객체 속성을 정의하는 다이얼로그에서 사용여부 선택 가능

- 설정된 객체는 마우스로 이동하거나 크기를 변경할 수 없도록 고정합니다.

인쇄 안함: 각 객체 속성을 정의하는 다이얼로그에서 인쇄 안함 선택 가능

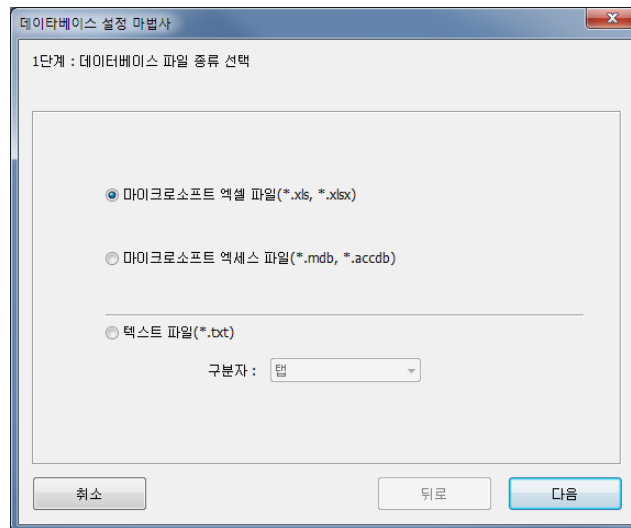
- 설정된 객체는 라벨 인쇄 시 적용되지 않습니다.

## 5-4 <데이터베이스> 메뉴

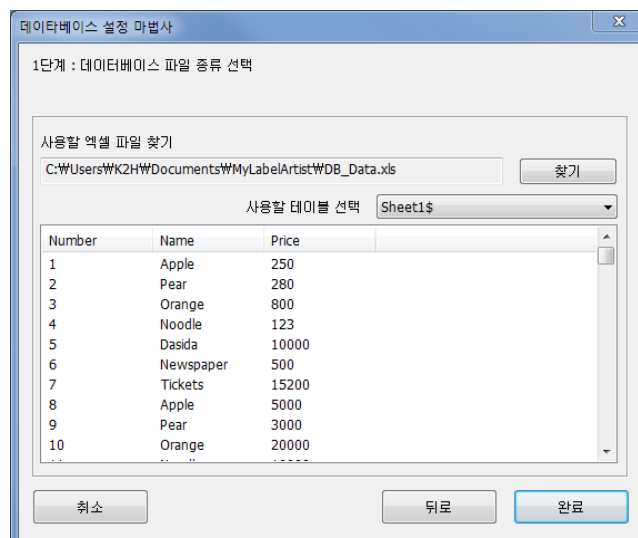
### 5-4-1 DB 연결하기

1) <데이터베이스> 메뉴의 <DB 연결하기> 항목을 선택합니다.

사용할 데이터베이스 종류를 선택하고 “다음” 버튼을 클릭합니다.



2) 아래의 화면에서 “찾기” 버튼을 클릭하고 사용할 데이터베이스 파일을 선택합니다.



3) 사용할 테이블을 선택하고 “완료” 버튼을 클릭하면 라벨 구성에 데이터베이스 내용을 객체에 연결할 수 있습니다.



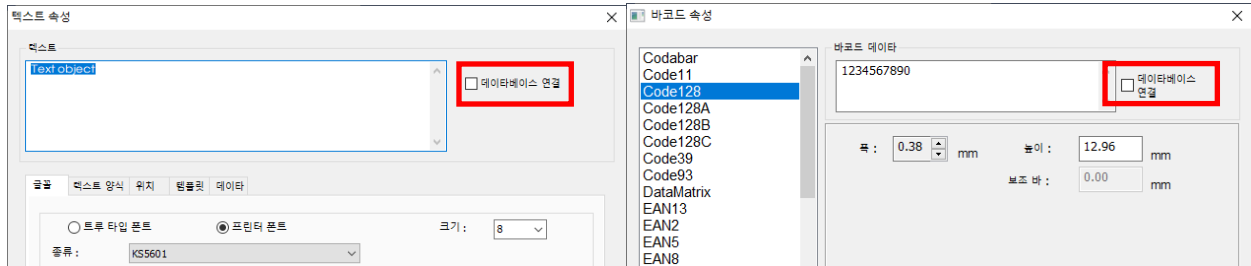
만약 다른 데이터베이스를 사용하려면 이미 연결된 데이터베이스의 연결을 해제해야 합니다.

## 5-4-2 DB 연결 해제하기

<데이터베이스> 메뉴의 <DB 연결끊기> 항목을 선택하여 데이터베이스 연결을 해제할 수 있습니다.

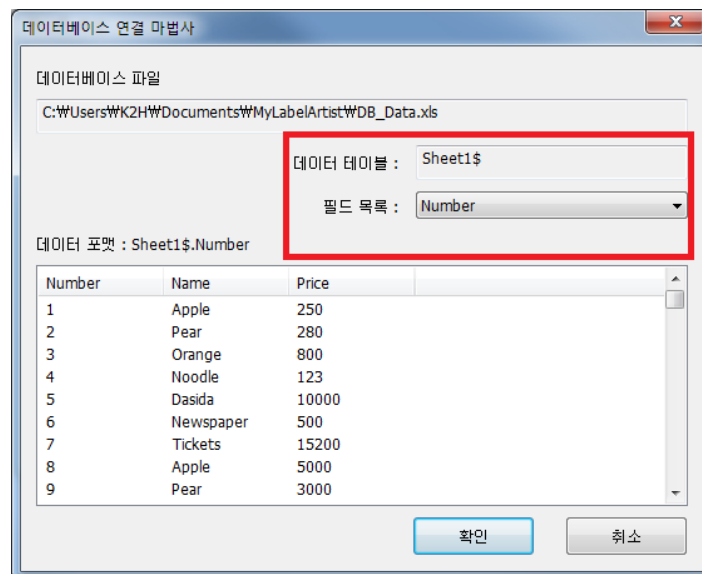
## 5-4-3 객체에 DB 연결하기

1) 텍스트 또는 바코드 객체 속성 다이얼로그에서 “데이터베이스 연결”을 체크합니다.



<데이터베이스> 메뉴에서 <DB 연결하기>가 완료되어야 객체 속성의 체크박스가 활성화됩니다.

2) 필드목록에서 사용할 필드를 선택합니다.



3) 텍스트 객체가 데이터베이스와 연결되면 화면에는 아래와 같이 표시됩니다.

[WorkSheetName].[FieldName]

Ex) Sheet1\$.Num

4) 바코드 객체가 데이터베이스와 연결되면 화면에는 임의의 데이터로 구성된 바코드가 그려집니다.

### 5-5 템플릿

- 빅솔론 템플릿(BIXOLON Template)과 빅솔론 XML(BIXOLON XML Enabled) 기능의 설정 및 내보내기 방법을 기술합니다.

#### 5-5-1 템플릿 설정하기

바코드 객체와 텍스트 객체는 템플릿을 설정할 수 있습니다. 그 외의 도형 객체 와 그림 객체는 템플릿의 배경으로 삽입됩니다.

템플릿 사용: ☒ 프롬프트: text 데이터 길이: 07

카운터 사용: ☐ 증감량: 1

##### 1) 템플릿 사용

텍스트와 바코드 속성 창에 템플릿 사용을 체크하고 프롬프트와 데이터 길이를 입력합니다.

##### 프롬프트란?

시리얼 통신에 의해 호스트(PC)로 전송되며, 변수의 정보를 호스트에 제공하기 위해 사용됨. 인쇄 내용에 영향을 주지 않습니다.

빅솔론 XML(BIXOLON XML Enabled) 기능의 설정 시 변수 이름으로 사용됩니다.

##### 2) 카운터 사용

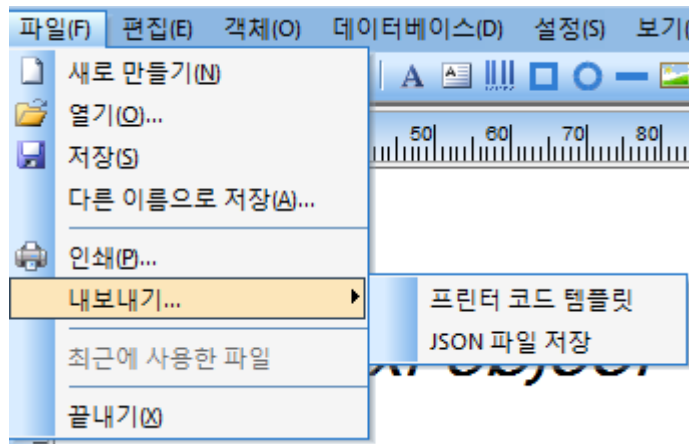
카운터 사용을 체크하고 증가 값을 설정합니다. 카운터는 하나의 템플릿에 10개를 초과할 수 없습니다.



빅솔론 템플릿(BIXOLON Template) 기능에서만 카운터 기능이 동작됩니다.

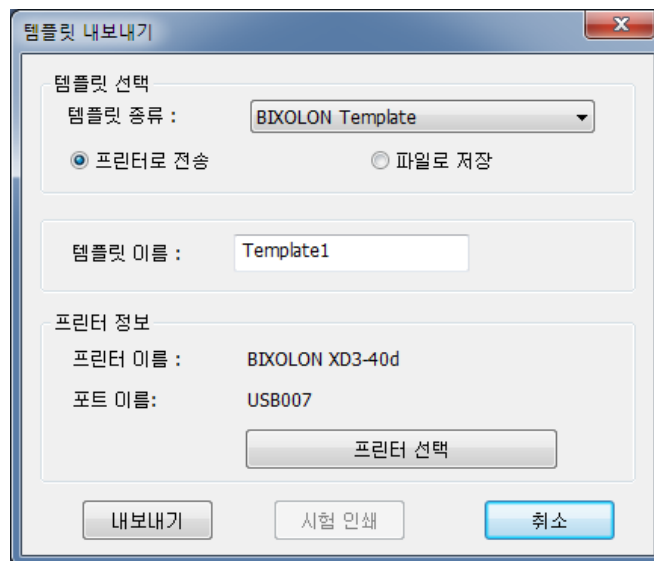
## 5-5-2 템플릿 내보내기

### 1) <파일> - <내보내기> - <프린터 코드 템플릿> 메뉴 선택



### 2) 템플릿 내보내기

템플릿 종류, 전송 방식, 템플릿 이름을 선택 후 내보내기 버튼을 클릭합니다.  
프린터로 내보내기가 성공하면 시험인쇄 버튼이 활성화됩니다.

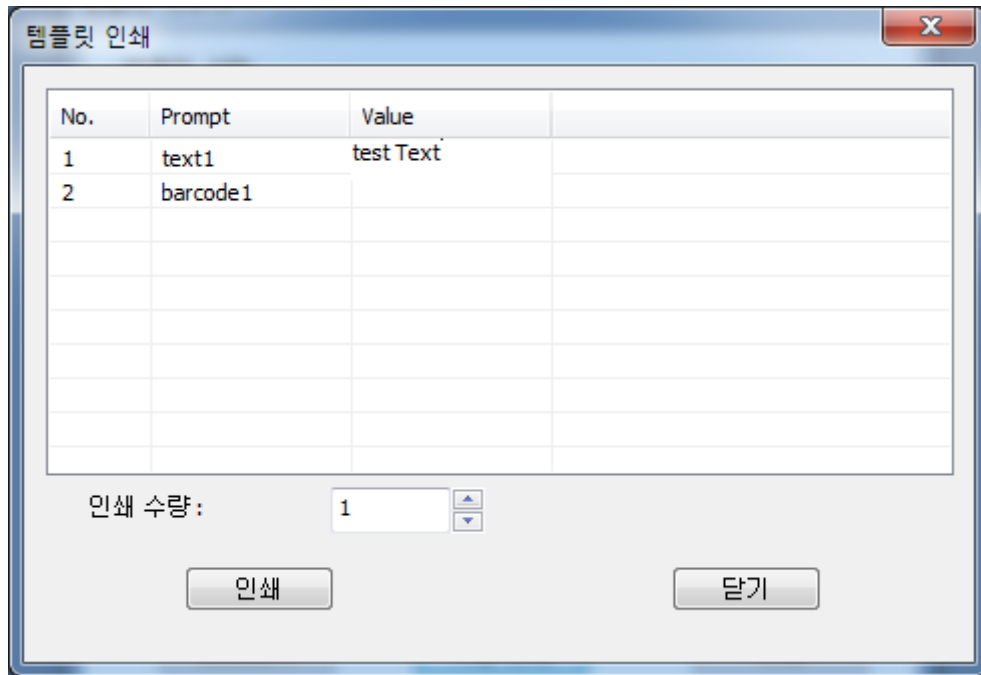


#### 템플릿 종류

- 빅솔론 템플릿(BIXOLON Template): SLCS 기반의 템플릿 양식을 프린터에 저장합니다.
- 빅솔론 XML(BIXOLON XML Enabled): XML 문서를 인쇄할 수 있는 템플릿 양식을 저장합니다.
- 빅솔론 XML(BIXOLON XML Enabled) 지원 모델  
XT5-40/43/46

### 3) 빅솔론 템플릿(BIXOLON Template) 시험 인쇄

“Value” 옆에 데이터와 인쇄 수량을 삽입 후 인쇄 버튼을 클릭합니다.



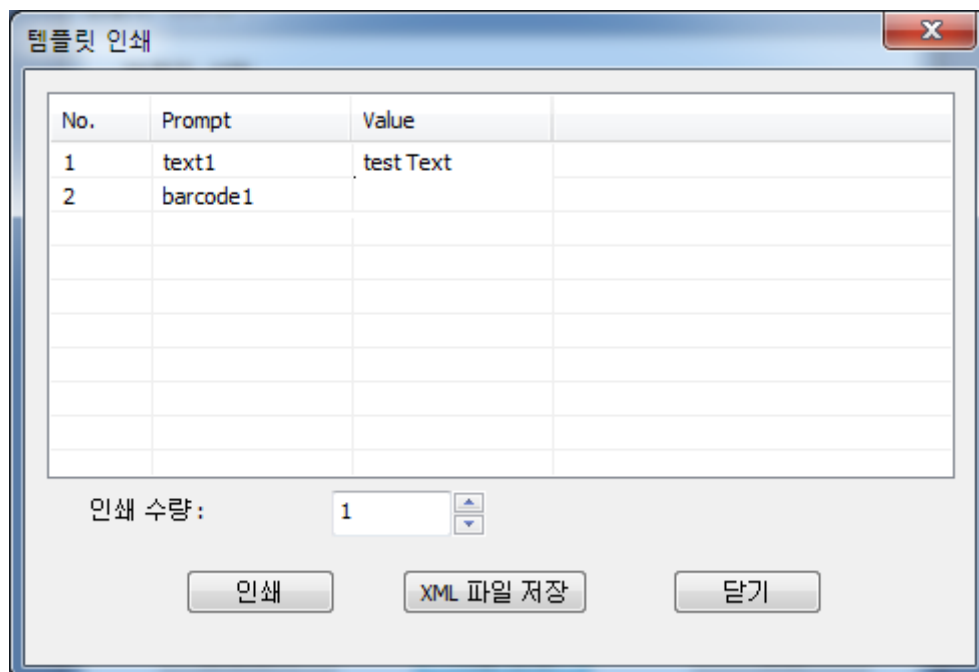
No.	Prompt	Value
1	text1	test Text
2	barcode1	

인쇄 수량: 1

인쇄 닫기

### 4) 빅솔론 XML(BIXOLON XML Enabled) 시험 인쇄

“Value” 옆에 데이터와 인쇄 수량을 삽입 후 인쇄 버튼을 클릭합니다.



No.	Prompt	Value
1	text1	test Text
2	barcode1	

인쇄 수량: 1

인쇄 XML 파일 저장 닫기

XML 파일 저장 버튼을 클릭하면, 입력한 데이터와 인쇄 수량이 적용된 XML 문서를 파일로 저장합니다.

### **5-6 JSON 파일 저장**

- <파일> - <내보내기> - <JSON 파일 저장> 메뉴를 선택하면 Label Artist Mobile과 연동하는 JSON 파일을 저장할 수 있습니다.

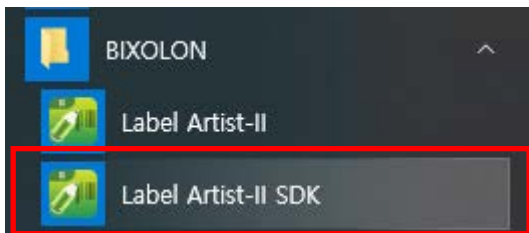
### **5-7 <보기> 메뉴**

- 1) 확대
  - 현재 보여지는 화면의 20%가 확대됩니다.
- 2) 축소
  - 현재 보여지는 화면의 20%가 축소됩니다.
- 3) 100% 크기
  - 원래 크기로 복원합니다.
- 4) 높이에 맞춤
  - 화면의 높이에 맞게 확대/축소합니다.
- 5) 너비에 맞춤
  - 화면의 너비에 맞게 확대/축소합니다.

## 6. Label Artist-II SDK

Label Artist-II 문서(.lafx)를 고객 응용 프로그램에서 자동으로 인쇄할 수 있도록 Label Artist-II SDK를 제공합니다. 데이터가 변경되는 텍스트 및 바코드 객체를 변수로 지정하여 응용 프로그램에서 변수의 값을 치환 후 인쇄하는 기능을 지원합니다.

Label Artist-II SDK는 <시작> - <프로그램 목록> - <BIXOLON> -<Label Artist-II SDK>에 있습니다.



Label Artist-II SDK는 C++ 라이브러리(Dll)와 C# 클래스를 제공합니다.

### -C# 구성 요소

파일 명	설명	위치
LabelArtistAPI.dll	x86 DLL 파일	_01_Bin 폴더
LabelArtistAPI_x64.dll	x64 DLL 파일	_01_Bin 폴더
LabelArtistAPI.cs	C# 클래스	C# 샘플

### -C++ 구성 요소

파일 명	설명	위치
LabelArtistAPI.dll	x86 DLL 파일	_01_Bin 폴더
LabelArtistAPI.lib	x86 LIB 파일	_01_Bin 폴더
LabelArtistAPI_x64.dll	x64 DLL 파일	_01_Bin 폴더
LabelArtistAPI_x64.lib	x64 LIB 파일	_01_Bin 폴더
LabelArtistSDK.h	헤더파일	VC++ 샘플

### 6-1 변수 설정

- Label Artist-II에서 텍스트 및 바코드 객체를 변수로 설정할 수 있으며, 변수 이름 및 초기 값을 지정해야 합니다.



변수는 텍스트(싱글라인 텍스트) 와 바코드 객체만 설정할 수 있습니다.

텍스트 및 바코드 속성창의 템플릿 사용을 선택합니다.

<input checked="" type="checkbox"/> 템플릿 사용	프롬프트: <input type="text" value="model"/>	데이터 길이: <input type="text" value="11"/>
<input type="checkbox"/> 카운터 사용	증감량: <input type="text" value="1"/>	

변수 이름을 프롬프트 항목에 기입 합니다.

<input checked="" type="checkbox"/> 템플릿 사용	프롬프트: <input type="text" value="model"/>	데이터 길이: <input type="text" value="11"/>
<input type="checkbox"/> 카운터 사용	증감량: <input type="text" value="1"/>	



변수 이름은 중복하여 사용할 수 없습니다.

### 6-2 C# 프로그래밍

- Label Artist-II SDK는 C# 프로그래밍을 위해 Visual Studio 2015(.NET 4.5)에서 작성된 C# 샘플을 제공합니다. C++ 라이브러리 (LabelArtistAPI.dll)를 임포트(Import)한 래핑 (Wrapping) 클래스인 LabelArtist를 제공하며, LabelArtist 클래스는 C# 샘플에 포함된 LabelArtistAPI.cs에 정의 되어있습니다.



LabelArtist 클래스를 사용하기 위해서는 DLL(LabelArtistAPI.dll, LabelArtistAPI\_x64.dll)파일을 응용프로그램 배포 시 패키지에 포함되어야 합니다.

#### 6-2-1 LabelArtist 프로퍼티

LabelArtist 클래스의 프로퍼티는, 변수에 대한 리스트와 각 메소드들의 리턴 정보로 구성됩니다.

##### 1) NamedSubStrings

변수 정보를 담은 SubString의 컬렉션(Collections) 입니다. OpenLabel 메소드가 성공적으로 호출되면 리스트 정보가 갱신됩니다.

#### SubString 클래스 프로퍼티



프로퍼티	설명
Name	변수 이름
Value	변수 값

**2) 메소드 리턴 정보**

메소드 호출 후 반환되는 에러 값들을 정의한 프로퍼티이며, 에러 내용은 아래의 표를 참조 바랍니다.

LabelArtist 프로퍼티	값	설명
BXL_SUCCESS	0	메소드 실행 성공
BXL_ERR_SEARCH_FILE	-100	파일을 찾을 수 없음
BXL_ERR_SEARCH_PRINTER	-101	프린터를 찾을 수 없음
BXL_ERR_SEARCH_OBJECT	-102	객체를 찾을 수 없음
BXL_ERR_OPEN_FILE	-103	파일 열기 실패
BXL_ERR_OPEN_PRINTER	-104	프린터 연결 실패
BXL_ERR_SUPPORT_FILE	-105	지원되지 않는 파일
BXL_ERR_SUPPORT_PRINTER	-106	지원되지 않는 프린터
BXL_ERR_PARAMETER	-107	유효하지 않는 입력 값(파라미터)
BXL_ERR_ALREADY_OPEN	-108	이미 프린터와 연결 됨
BXL_ERR_NOT_OPEN	-109	프린터와 연결되지 않음
BXL_ERR_VARIABLE_NAME	-110	유효하지 않은 변수 명 사용
BXL_ERR_PRINT_RESOURCE	-111	프린터 리소스 사용 실패
BXL_ERR_PRINT	-112	인쇄 실패

### 6-2-2 LabelArtist 메소드

LabelArtist 클래스는 Label Artist-II 문서(.lafx) 오픈, 변수 값 설정, 인쇄의 기능을 제공합니다.

#### 1) OpenLabel

Label Artist-II 문서를 열고 인쇄 할 프린터를 지정합니다. OpenLabel 메소드 호출이 성공되면 NamedSubStrings를 갱신 합니다.

#### [구문]

```
public int OpenLabel(string filePath, string printerName)
```

#### [파라미터]

string filePath: Label Artist-II 문서(.lafx 파일) 경로

string printerName: 프린터 드라이버 이름(공백 가능, 옵션)

#### [반환 값]

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환



- printerName 파라미터는 옵션으로 공백을 입력하면 아래의 순서로 인쇄 프린터를 지정합니다.

1. 입력된 파일(.lafx)에 지정된 프린터로 인쇄
  2. 만약 파일에 지정된 프린터가 시스템에 없다면, 시스템 기본 프린터로 인쇄
- 파일 경로 및 프린터 드라이버 이름에 한글이 포함되어 유니코드(UNICODE) 문자열 사용 시 OpenLabelW 메소드를 사용 바랍니다.

### 2) CloseLabel

현재 열려있는 Label Artist-II 문서를 닫습니다.

#### **[구문]**

```
public int CloseLabel()
```

#### **[파라미터]**

없음

#### **[반환 값]**

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환

### 3) SetNamedSubStringValue

변수의 값을 설정합니다.

#### [구문]

```
public int SetNamedSubStringValue(string name, string value)
```

#### [파라미터]

string name: 변수 이름

string value: 변수 값

#### [반환 값]

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환



변수 이름 및 값에 한글이 포함되어 유니코드(UNICODE) 문자열 사용 시 **SetNamedSubStringValueW** 메소드를 사용 바랍니다.

### 4) Find

객체 이름으로 객체 정보를 가져옵니다.

#### [구문]

```
public DesignObject Find(string objectName)
```

#### [파라미터]

string objectName: 객체 이름

#### [반환 값]

성공 시 DesignObject 객체를 반환하며, 객체를 찾을수 없을 시 null 반환

#### DesignObject 클래스 프로퍼티



프로퍼티	설명
X	객체의 x(left) 좌표 위치
Y	객체의 y(left) 좌표 위치
Width	객체의 너비
Height	객체의 높이

### 5) PrintOut

열린 Label Artist-II 문서를 인쇄합니다.

#### **[구문]**

```
public int PrintOut()
```

#### **[파라미터]**

없음

#### **[반환 값]**

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환

### 6) Save

변경된 객체 정보를 Label Artist-II 문서에 적용 후 파일에 저장합니다.

#### **[구문]**

```
public int Save()
```

#### **[파라미터]**

없음

#### **[반환 값]**

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환

### 6-2-3 LabelArtist 활용 예제

LabelArtist 클래스를 활용하여 C# 프로그래밍 방법을 기술합니다.

#### 1) Label Artist-II 문서 열기 후 변수 가져오기

OpenLabel 메소드로 Label Artist-II 문서를 연 후 NamedSubStrings 컬렉션(Collections)으로 변수 이름 및 초기값을 가져옵니다.

```
LabelArtist format = new LabelArtist();
int ret = format.OpenLabel(filePath, printerName);
if (ret == LabelArtist.BXL_SUCCESS)
{
    DataTable table = new DataTable();
    table.Columns.Add("Variable Name");
    table.Columns.Add("Variable Value");

    // Getting for variable name, value
    foreach (SubString laLabelVar in format.NamedSubStrings)
    {
        table.Rows.Add(laLabelVar.Name, laLabelVar.Value);
    }
}
```

#### 2) 변수 값 설정 후 문서 인쇄

SetNamedSubStringValue 메소드로 변수의 값 설정 후 PrintOut 메소드로 인쇄합니다.

```
DataRow[] rows = table.Select();
for (int i = 0; i < rows.Length; i++)
{
    format.SetNamedSubStringValue(rows[i]["Variable Name"].ToString(),
        rows[i]["Variable Value"].ToString());
}
format.PrintOut();
```

### 3) 객체 위치 변경 후 저장

```
DesignObject obj = format.Find("picture01");  
if(obj != null)  
{  
    obj.X += 3; //3mm  
    obj.Y -= 2;  
    obj.Width -= 5;  
    obj.Height += 2;  
    format.Save(); //Save  
}
```

### 6-3 C++ 프로그래밍

- Label Artist-II SDK는 C++ 프로그래밍을 위해 Visual Studio 2015에서 작성된 VC++ 샘플을 제공합니다. VC++ 샘플은 샘플과 함께 제공된 Label Artist-II 문서 Sample.lafx 기준으로 작성되었습니다.

이름		수정한 날짜
C#		2020-02-12 오후 2:51
VC++		2020-02-12 오후 2:51
Sample.lafx		2020-02-10 오전 11:30

Label Artist-II SDK에서 제공되는 API 및 정의된 반환 값은 VC++ 샘플에 포함된 헤더파일(LabelArtistSDK.h)을 참고 바랍니다.

이름			수정한 날짜	유형
LabelArtistSDK.h			2020-01-20 오후 4:53	C/C++ Header

### 6-3-1 Label Artist-II SDK 메소드

#### 1) OpenLabel

Label Artist-II 문서를 열고 인쇄 할 프린터를 지정합니다.

##### [구문]

```
long OpenLabel(LPCSTR filePath, LPCSTR szPrinterName = "")
```

##### [파라미터]

LPCSTR filePath: Label Artist-II 문서(.lafx 파일) 경로

LPCSTR szPrinterName: 프린터 드라이버 이름(공백 가능, 옵션)

##### [반환 값]

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환



• szPrinterName 파라미터는 옵션으로 공백을 입력하면 아래의 순서로 인쇄 프린터를 지정합니다.

1. 입력된 파일(.lafx)에 지정된 프린터로 인쇄

2. 만약 파일에 지정된 프린터가 시스템에 없다면, 시스템 기본 프린터로 인쇄

• 파일 경로 및 프린터 드라이버 이름에 한글이 포함되어 유니코드(UNICODE) 문자열 사용 시 OpenLabelW 메소드를 사용 바랍니다.

#### 2) CloseLabel

현재 열려있는 Label Artist-II 문서를 닫습니다.

##### [구문]

```
long CloseLabel()
```

##### [파라미터]

없음

##### [반환 값]

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환

### 3) SaveLabel

열린 Label Artist-II 문서를 저장합니다.

#### **[구문]**

```
long SaveLabel()
```

#### **[파라미터]**

없음

#### **[반환 값]**

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환

### 4) GetVariableCount

변수의 개수를 가져옵니다.

#### **[구문]**

```
long GetVariableCount(int& count)
```

#### **[파라미터]**

int& count: 문서에 포함된 변수 개수

#### **[반환 값]**

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환

### 5) GetVariableName

위치(POS)에 해당하는 변수 이름을 가져옵니다.

#### [구문]

```
long GetVariableName(LPCSTR pName, int pos = BXL_IN_ORDER)
```

#### [파라미터]

LPCSTR pName: 변수 이름(IN)

int pos: 변수 값(OUT).

BXL\_IN\_ORDER(-1)을 설정하면 호출 순차적으로 변수 이름을 가져옵니다.

#### [반환 값]

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환



변수 이름에 한글 등이 포함된다면 **GetVariableNameW** 메소드를 사용하여 유니코드(UNICODE) 타입의 문자열로 사용 바랍니다.

### 6) GetVariableValue

변수 이름으로 변수 값을 가져옵니다.

#### [구문]

```
long GetVariableValue(LPCSTR pName, LPCSTR pValue)
```

#### [파라미터]

LPCSTR pName: 변수 이름(OUT)

LPCSTR pValue: 변수 값(IN)

#### [반환 값]

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환



변수 이름 및 값에 한글 등이 포함된다면 **GetVariableValueW** 메소드를 사용하여 유니코드(UNICODE) 타입의 문자열로 사용 바랍니다.

### 7) SetVariableValue

변수 값을 설정합니다.

#### [구문]

```
long SetVariableValue(LPCSTR pName, LPCSTR pValue)
```

#### [파라미터]

LPCSTR pName: 변수 이름(OUT)

LPCSTR pValue: 변수 값(OUT)

#### [반환 값]

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환



변수 이름 및 값에 한글 등이 포함된다면 **SetVariableValueW** 메소드를 사용하여 유니코드(UNICODE) 타입의 문자열로 사용 바랍니다.

### 8) FindObject

객체 이름으로 객체 정보를 가져옵니다.

#### [구문]

```
long FindObject(LPCSTR objectName, Object* obj);
```

#### [파라미터]

LPCSTR objectName: 변수 이름(OUT)

Object\* obj: 객체 정보(OUT)

#### [반환 값]

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환



변수 이름에 한글 등이 포함된다면 **FindObjectW** 메소드를 사용하여 유니코드(UNICODE) 타입의 문자열로 사용 바랍니다.

### 9) SetObject

객체 정보를 설정합니다.

#### **[구문]**

```
long SetObject(Object* obj);
```

#### **[파라미터]**

Object\* obj: 객체 정보

#### **[반환 값]**

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환

### 10) PrintOut

열린 Label Artist-II 문서를 인쇄합니다.

#### **[구문]**

```
long PrintOut()
```

#### **[파라미터]**

없음

#### **[반환 값]**

성공 시 반환 값은 0 이며 그 이외 경우 0 아닌 값 반환

### 6-3-2 Label Artist-II SDK 에러코드

아래 에러 코드 표는 Label Artist-II SDK에서 제공되는 API를 호출하였을 때 반환되는 값을 요약한 것입니다.

에러코드	값	설명
BXL_SUCCESS	0	실행 성공
BXL_ERR_SEARCH_FILE	-100	파일을 찾을 수 없음
BXL_ERR_SEARCH_PRINTER	-101	프린터를 찾을 수 없음
BXL_ERR_SEARCH_OBJECT	-102	객체를 찾을 수 없음
BXL_ERR_OPEN_FILE	-103	파일 열기 실패
BXL_ERR_OPEN_PRINTER	-104	프린터 연결 실패
BXL_ERR_SUPPORT_FILE	-105	지원되지 않는 파일
BXL_ERR_SUPPORT_PRINTER	-106	지원되지 않는 프린터
BXL_ERR_PARAMETER	-107	유효하지 않는 입력 값(파라미터)
BXL_ERR_ALREADY_OPEN	-108	이미 프린터와 연결 됨
BXL_ERR_NOT_OPEN	-109	프린터와 연결되지 않음
BXL_ERR_VARIABLE_NAME	-110	유효하지 않은 변수 명 사용
BXL_ERR_PRINT_RESOURCE	-111	프린터 리소스 사용 실패
BXL_ERR_PRINT	-112	인쇄 실패

### 6-3-3 Label Artist-II API 활용 예제

Label Artist-II API를 활용한 C++ 프로그래밍 방법을 기술합니다.

#### 1) Label Artist-II 문서 열기

OpenLabel 메소드로 Label Artist-II 문서를 오픈(Open) 합니다. 파일 경로에 한글이 포함 될 경우 문자열을 유니코드(UNICODE) 타입을 사용하며, OpenLabelW 메소드를 사용합니다.

```
long ret = OpenLabelW(m_filePath, m_printerName);  
if (ret == BXL_SUCCESS) AfxMessageBox(_T("Open Success"));  
else AfxMessageBox(_T("Open Fail"));
```

#### 2) 변수 값 설정

GetVariableName 메소드로 변수 이름을 검색 후 SetVariableValue 메소드로 변수 값을 설정 합니다.

```
long ret;  
TCHAR variableName[100];  
CString name;  
  
wmemset(variableName, 0, 100);  
  
ret = GetVariableNameW(variableName);  
  
if (ret != BXL_SUCCESS) return;  
name.Format(_T("%s"), variableName);  
  
while (name.GetLength()  
{  
    if (!name.Compare(_T("model"))) {  
        SetVariableValueW(name, _T("SDK Printer"));  
    }  
    if (!name.Compare(_T("sn"))) {  
        SetVariableValueW(name, _T("NUMBER2020010001"));  
    }  
    wmemset(variableName, 0, 100);  
    GetVariableNameW(variableName);  
    name.Format(_T("%s"), variableName);  
}
```



변수 이름 및 변수 값을 가져오기 위해 파라미터(Parameter)에 사용되는 문자열(LPCSTR / LPCWSTR)은 응용프로그램에서 메모리를 관리(할당 및 해제)해야 합니다.

### 3) 객체 위치 조정

FindObject 메소드로 객체 정보를 가져온 후 SetObject 메소드로 객체를 설정합니다.

```
Object obj;
ret = FindObjectW(_T("picture01"), &obj);
if ((ret == BXL_SUCCESS)) {
    obj.x += 3;      //3mm
    obj.y -= 2;
    obj.width -= 5;
    obj.height += 2;
    SetObject(&obj);
}
```

### 4) 인쇄

PrintOut 메소드로 Label Artist-II 문서를 인쇄 합니다.

```
long ret = PrintOut();
if (ret == BXL_SUCCESS) AfxMessageBox(_T("Print Success"));
else AfxMessageBox(_T("Print Fail"));
```

## 개정 이력

버전	날짜	설명
2.00	2018-03-12	신규제작
2.01	2018-05-18	모델 명 변경
2.02	2019-01-14	SPP-L310/L410 모델 추가, 한 부씩 인쇄 옵션 추가
2.03	2019-03-04	XD3-40d 모델 추가
2.04	2019-04-05	프린터 폰트 선택 추가, 템플릿 기능 추가, JSON 파일 저장
2.05	2019-06-24	XD3-40t, XL5-40/43, XD5-40d/43d 모델 추가 BIXOLON XML Enabled 기능 추가 인쇄 속도, 농도 설정 기능 추가
2.06	2019-12-03	XD5-40t/43t, XM7-40 모델 추가, XL5-40CT/43CT 모델명 변경
2.07	2020-03-11	XM7-20 모델 추가, 그림 속성 창 변경 Label Artist-II SDK 기능 추가
2.08	2020-11-10	SRP-S3000 모델 추가 지원 OS 변경, 코드페이지 설정 추가, 객체 설정 변경 및 RFID 추가 객체 인쇄 안함 설정 추가
2.09	2021-01-27	캡처 이미지 변경
2.10	2021-08-24	XM7-40 RFID 표기 추가 XT3-40/43 모델 추가 BDP-T42/T43/D42/D43, BTP-42/43/46 모델 추가