

[데이터 트랙 - 향상] 스마트폰의 데이터 활용하기

- 스마트폰의 정보(기기 종류와 운영체제 정보를 조회하는 방법 ! 과 스마트폰에 이미지, 데이터 파일 등을 넣고 필요할 때 앱에서 활용하는 방법을 알아보겠습니다.
- 모바일 도서 2편-스마트폰의 정보 사용하기를 참조하시기바랍니다.
- 교육 전에 이 도서를 다운로드 해 두시기 바랍니다.

<https://welcome.devgear.co.kr/guide-embarcadero/book/>

[데이터 트랙 - 향상] 스마트폰의 데이터 활용하기

모바일 앱 개발 방식

- Fmx 컴포넌트 사용
- IFMX 인터페이스를 사용하기
(예 : IFMXScreenService, IFMXPhoneDialerService 등)
- 안드로이드, IOS등의 자체 Native API 사용

OS 기능을 이용한 스마트폰의 정보 조회하기

앱을 컴파일 할 때에 타겟 플랫폼을 안드로이드로 선택한 경우에는 내부적으로 안드로이드의 API를 이용해 앱이 만들어지고, 타겟 플랫폼을 iOS(iOS Device, iOS Simulator)로 선택한 경우에는 내부적으로 iOS의 API를 이용해 앱이 만들어집니다. 스마트폰의 정보 조회하기 기능은 델파이 기본 컴포넌트로 제공되지 않는 기능입니다. 하지만 컴포넌트가 동작하는 원리와 같이 우리는 플랫폼(안드로이드와 iOS) 별로 동작하는 코드를 작성해 스마트폰의 정보를 조회할 수 있습니다.

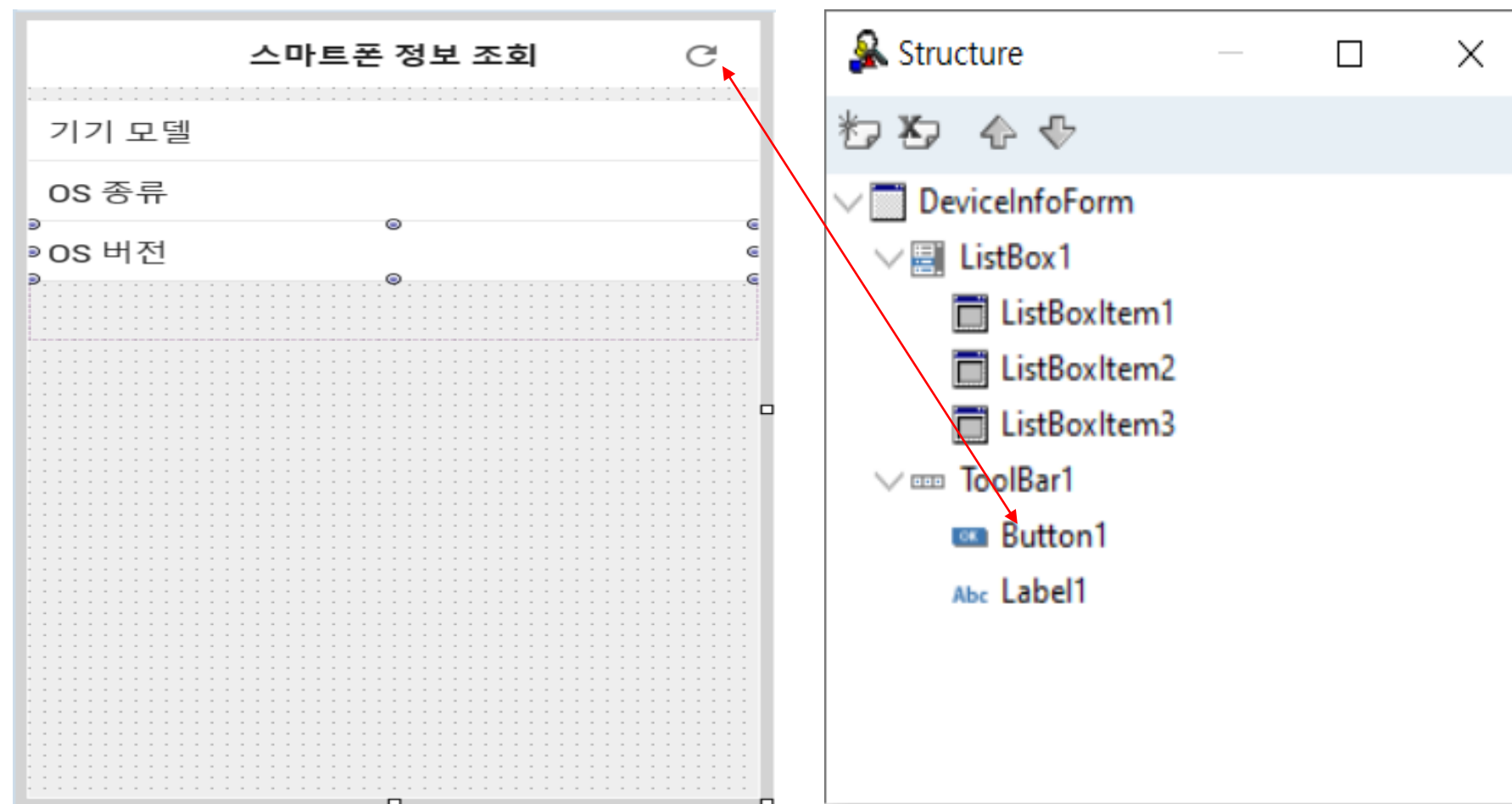
이 과정을 이해하면 소스 코드를 하나로 유지하면서 안드로이드와 iOS의 고유 기능을 활용할 수 있는 앱을 개발할 수 있습니다.

[데이터 트랙 - 향상] 스마트폰의 데이터 활용하기

[따라 하기]

스마트폰의 기기 모델, OS 종류, OS 버전을 OS의 고유기능을 이용해 조회하고, 화면에 표시하는 앱을 만들어봅니다.

1. 화면 UI 디자인



상위 오브젝트	오브젝트	속성	값(또는 설명)
Form1	ToolBar1		
ToolBar1	Label1	Align	Contents
		Text	스마트폰 정보 조회
	Button1	TextSettings.HorzAlign	Center
Form1	Button1	Align	Right
		StyleLookup	refreshtoolbarbutton
	ListBox1	Align	Top
Form1	ListBox1	StyleLookup	transparentlistboxstyle
	ListBox1	Text	각각 다음과 같이 입력 '기기모델 :' 'OS 종류 :' 'OS 버전 :'
ListBox1	ListBoxItem1 ListBoxItem2 ListBoxItem3	StyleLookup	3개 모두 'listboxitemrightdetail' 로 설정

[데이터 트랙 - 향상] 스마트폰의 데이터 활용하기

[따라 하기]

2. 스마트폰의 정보 조회에 필요한 플랫폼 별 고유의 유닛 사용을 위해 아래 코드를 플랫폼 별로 추가합니다.
(implementation과 (\$R * fmx} 사이의 코드를 입력합니다.)

```
implementation
{ $IFDEF IOS }
uses
    iOSapi.UIKit, iOSapi.Foundation, Macapi.Helpers;
{ $ENDIF }
{ $IFDEF ANDROID }
uses
    androidapi.JNI.JavaTypes,
    Androidapi.Helpers,
    androidapi.JNI.Os;
{ $ENDIF }
```

[데이터 트랙 - 향상] 스마트폰의 데이터 활용하기

[따라 하기]

3. 안드로이드 버전 문자열을 코드명으로 반환하는 함수(GetAndroidCodename)를 {\$R *.frm 아래에 추가합니다. 안드로이드 플랫폼으로 컴파일하는 경우에만 사용하도록 조건부 컴파일 문((\$IFDEF ANDROID ... {\$ENDIF)으로 함수를 감싸줍니다.(이 부분은 C:\Users\Public\Documents\Embarcadero\Studio\21.0\Samples\Object Pascal\Mobile Snippets\DeviceInfo 에 있는 소스 부분을 복사해서 사용하겠습니다).

```
{$IFDEF ANDROID}
```

```
Function GetAndroidCodename(VerString: string): string;
```

```
begin
```

```
if VerString = '1.0' then
```

```
    Result := 'BASE'
```

```
else if VerString = '1.1' then
```

```
    Result := 'BASE_1_1'
```

```
else if VerString = '1.5' then
```

```
    Result := '컵케이크' ..... 이하 생략
```

[데이터 트랙 - 향상] 스마트폰의 데이터 활용하기

[따라 하기]

4. 정보 구하기 버튼의 OnClick 이벤트를 핸들러를 만들고 아래 코드를 작성합니다.

```
procedure TDeviceInfoForm.Button1Click(Sender: TObject);  
  
begin  
    {$IFDEF ANDROID}  
        ListBoxItem1.ItemData.Text := Format('기기타입: %s', [JStringToString(TJBuild.JavaClass.MODEL)]);  
  
        ListBoxItem2.ItemData.Text :=  
            Format('OS 이름: %s', [GetAndroidCodename(JStringToString(TJBuild_VERSION.JavaClass.RELEASE))]);  
        GetAndroidCodename(JStringToString(TJBuild_VERSION.JavaClass.RELEASE));  
  
        ListBoxItem3.ItemData.Text := Format('OS 버전: %s', [JStringToString(TJBuild_VERSION.JavaClass.RELEASE)]);  
    {$ENDIF}  
  
    {$IFDEF IOS}  
        var Device : UIDevice;  
  
        Device := TUIDevice.Wrap(TUIDevice.OCClass.currentDevice);  
  
        ListBoxItem1.ItemData.Text := Format('OS Name: %s', [NSStrToStr(Device.systemName)]);  
  
        ListBoxItem2.ItemData.Text := Format('OS Version: %s', [NSStrToStr(Device.systemVersion)]);  
  
        ListBoxItem3.ItemData.Text := Format('Device Type: %s', [NSStrToStr(Device.model)]);  
    {$ENDIF}  
  
end;
```

[데이터 트랙 - 향상] 스마트폰의 데이터 활용하기

[따라 하기]

4. 앱을 실행하여 본인 디바이스,os 정보들을 확인하여 봅니다.