

Chapter 7 공통 스크립트 작성하기

1. 공통 스크립트

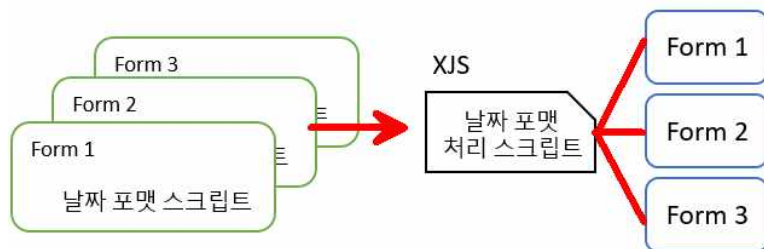
1.1 공통 스크립트(XJS)

공통 스크립트는 프로젝트에서 자주 사용하는 기능을 모아 놓은 일종의 스크립트 라이브러리이다. 특정 폼에만 필요한 스크립트는 폼 파일에 작성하지만 애플리케이션 전체에서 사용하는 스크립트는 별도의 파일(JS)에 작성하여 각 폼에서 이를 참조해 사용하는 것이 좋은데, 넥사크로플랫폼에서는 이를 XJS 파일을 통해 공통스크립트를 작성할 수 있다.

1.2 공통 스크립트의 필요성

공통 스크립트를 한번 만들어 놓으면 같은 기능을 여러 번 개발하는 중복을 피할 수 있고, 여러 명이 작업할 때 통일된 기능을 사용할 수 있어 효율적인 개발 작업이 가능하다.

또한, 공통 스크립트 파일을 그대로 다른 프로젝트에서도 사용할 수 있으므로 코드의 재사용, 개발 시간 단축의 효과를 얻을 수 있다. 그러나 불필요하게 남용하면 애플리케이션의 크기가 커져 리소스 낭비와 속도 지연 등의 원인이 될 수 있으므로 반드시 필요한 상황을 설계하여 공통 스크립트 작업을 남용하지 않도록 해야 한다.



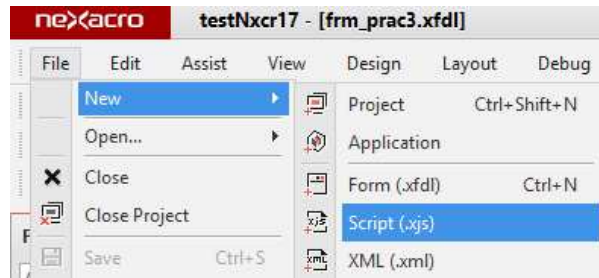
1.3 공통 스크립트 사용처

일반적으로 공통 스크립트로 구현하는 기능은 문자 및 날짜 체크, 문자열 처리, 형식 검증, 각종 연산, 인/디코딩, 통신 연결 등의 공통으로 사용하는 기능은 무엇이든 될 수 있으며 프로젝트에 따라 그 내용은 달라질 수 있다.

2. 공통 스크립트 작성하기

이번 장에서는 날짜 데이터를 사용하며 자주 겪게 되는 두 날짜 사이의 차이를 계산하는 공통 스크립트를 작성해보도록 하자.

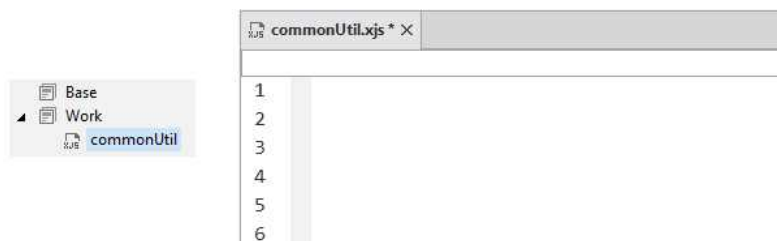
먼저 공통 스크립트는 메뉴에서 File > New > Script(.xjs)를 선택하여 작성한다.



공통 스크립트의 이름은 'commonUtil' 이라고 짓고, Location은 Work form을 선택한다.



잘 생성하였다면 Work Form 내부에 commonUtil.xjs가 생성된 것을 확인할 수 있다.



이제 사용할 스크립트를 다음과 같이 작성한다.

```
01  /* Common Script */
02
03  // 함수 선언 시 this를 붙이지 않으면 전역변수로 처리한다.
04  this.dayDiff = function(calFrom, calTo){
05      var fromDate = new Date();
06      var toDate = new Date();
07      var calDate;
08      var day = 1000*60*60*24;
09
10      fromDate.setFullYear(calFrom.getFullYear());
11      fromDate.setMonth(calFrom.getMonth()-1);
12      fromDate.setDate(calFrom.getDay());
13
14      toDate.setFullYear(calTo.getFullYear());
15      toDate.setMonth(calTo.getMonth()-1);
16      toDate.setDate(calTo.getDay());
17
18      calDate = fromDate.getTime() - toDate.getTime();
19
20      return Math.abs(calDate/day);
21  }
```

3. 공통 스크립트 사용하기

Form에서 공통 스크립트를 사용하려면 해당 페이지에서 include문을 선언해주면 된다.
먼저 새로운 Form을 하나 작성하여 다음과 같이 디자인하자.

From Date	-- --
To Date	-- --
계산하기	계산결과

다음으로 Script 창을 열어 다음과 같이 코드를 작성한다.

```

01 include "Work::commonUtil.xjs";
02 /* 공통 코드가 많은 경우 다중 선언도 가능하다 */
03 /* Ex)
04 * include "Work::isNumber.xjs";
05 * include "Work::isNotNull.xjs";
06 */
07
08 this.Button00_onclick =
09     function(obj:nexacro.Button,e:nexacro.ClickEventInfo)
10 {
11     var result = this.dayDiff(this.Calendar00, this.Calendar01);
12
13     this.Edit00.set_value(result+' 일');
14 };
  
```

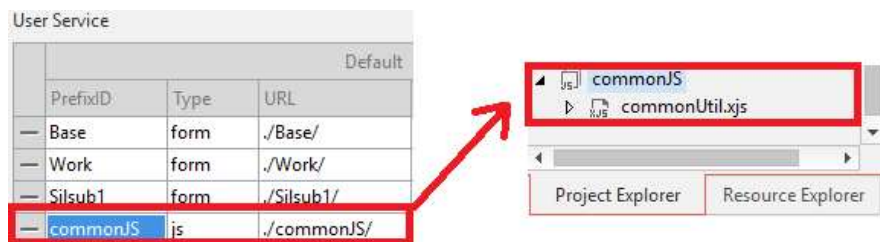
작성이 완료되었다면 실행하여 정상적으로 동작하는지 확인해보자.

From Date	-- --	From Date	2018-11-12 월
To Date	-- --	To Date	2018-12-25 화
계산하기	계산결과	계산하기	43 일

4. 공통 스크립트 등록하기

만약 다른 곳에서 작성된 XJS를 등록하고자 한다면, 해당 .xjs파일을 사용하고자 하는 Form에 붙여넣기 하거나 서비스로 등록하는 방법이 있다.

그럼, 서비스로 등록하여 사용하는 방법에 대해 알아보자. Project Explorer의 TypeDefinition > Services를 선택 후 마우스 오른쪽 버튼을 클릭하여 컨텍스트 메뉴에서 [Edit] 항목을 선택하고 서비스 편집 창을 연 뒤 js 타입의 서비스를 등록한다.



만약 필요로 하는 공통 스크립트가 프로젝트의 메인인 즉, 애플리케이션의 시작이나 종료 시에 동작해야 하는 스크립트라면 App Information에 등록하여 사용하기도 하는데, 해당 방법은 Project Explorer에서 프로젝트의 App '[프로젝트명]'을 선택한 후 마우스 오른쪽 버튼을 클릭하여 [Edit Script]를 선택하여 생성되는 .xadl 편집 창에서, nexacro 접근자를 사용하여 구현한다.

또한 사용하는 Form 화면에서는 nexacro.getApplication() 메소드를 사용하여 application 객체를 얻은 후 .xadl에 정의된 함수를 사용하면 된다.