

# BIX5

## 위젯 데이터 업데이트 주기 설정 가이드

Guide to Setting Widget Data Update Intervals on BIX5 Server

...

정품을 구입하신 고객에게는 기술상담 및 지원을 제공합니다.

(주)빅스소프트

서울시 영등포구 영신로 220, 610 호 (영등포동 8 가, KnK 디지털타워)

Tel. 02-2655-9784, [bix5cs@bixsoft.net](mailto:bix5cs@bixsoft.net)

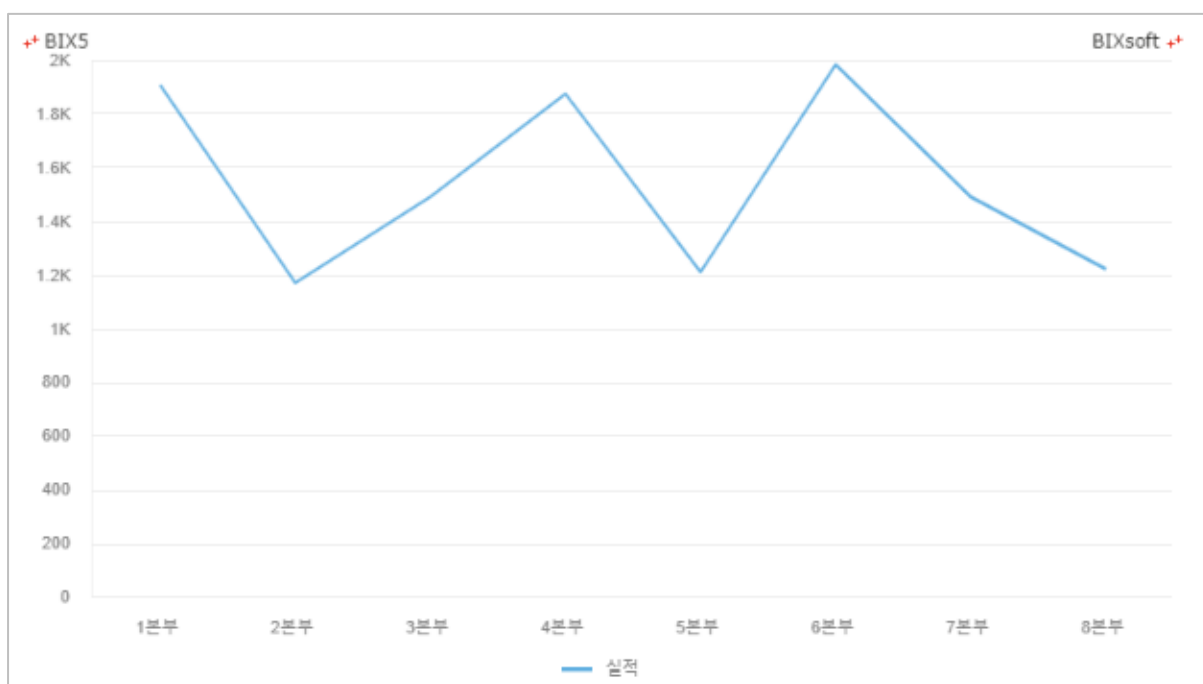
# 목차

1. 개요 .....	2
1.1.    예제에 대한 설명 .....	2
2. 위젯 데이터 업데이트 주기 설정하기 .....	3
2.1.    데이터셋 스케줄 방식 .....	3
2.1.1.    데이터셋 생성 및 설정하기 .....	3
2.1.2.    위젯에 데이터 바인딩 하기 .....	5
2.2.    자바스크립트 interval 방식 .....	7
2.2.1.    데이터셋 생성 및 설정하기 .....	7
2.2.2.    위젯에 데이터 바인딩 하기 .....	8
2.2.3.    자바스크립트에서 바인딩 된 데이터셋 interval 호출 .....	9

## 1. 개요

해당 가이드는 위젯에서 바인딩 된 데이터를 특정 시간 간격으로 주기적으로 업데이트하여 표출하는 방법에 대하여 설명합니다. 여기서 '바인딩'이란 위젯에 데이터셋을 사용하도록 설정하는 것을 말합니다. 이 가이드를 통해 사용자는 별도의 새로 고침 작업 없이도 특정 시간 간격마다 자동으로 데이터를 업데이트하도록 설정할 수 있습니다.

### 1.1. 예제에 대한 설명



위젯에서 바인딩 된 데이터를 특정 시간 간격으로 반복 업데이트하여 위젯에 표출하는 예제를 설명하겠습니다. 이 예제에서는 차트 위젯 하나로 구성된 대시보드를 통해 설명하도록 하겠습니다. 데이터셋의 스케줄 방식 과 자바스크립트의 Interval 메소드를 사용하여 업데이트하는 방식 2 가지 시나리오로 진행하겠습니다.

시나리오에 사용될 데이터셋은 아래와 같습니다.

본부	실적
1본부	1907
2본부	1167
3본부	1490
4본부	1873
5본부	1207
6본부	1980
7본부	1488
8본부	1220

## 2. 위젯 데이터 업데이트 주기 설정하기

### 2.1. 데이터셋 스케줄 방식

데이터셋 스케줄 방식으로 진행하게 될 경우 해당 데이터셋을 쓰는 모든 위젯들은 스케줄 방식에 의하여 데이터를 반복적으로 업데이트합니다.

#### 2.1.1. 데이터셋 생성 및 설정하기

아래의 순서대로 진행하는 것을 권장해 드립니다.

1. [데이터셋 방식]은 맞춤 쿼리 또는 테이블 둘 중 하나를 선택하여 생성합니다. [스케줄 방식]의 콤보 박스를 [반복 형식]으로 선택합니다.

### 데이터셋

이름

데이터셋\_반복\_호출\_데이터(스케줄 방식)

메모

데이터셋 방식

맞춤 쿼리

서비스 방식

원격 데이터 서비스

스케줄 방식

없음

상태

☒ 사용 ☐ 중지

없음

반복 형식

다음

취소

2. 스케줄 방식의 콤보 박스를 [반복 형식]으로 선택 시 스케줄의 반복 시간(초)을 입력할 수 있는 칸이 나타납니다. 원하는 시간(초)을 입력한 뒤 다음 버튼을 클릭합니다. 예제에서는 10 초마다

반복하여 업데이트되도록 설정해보겠습니다.

### 데이터셋

이름

데이터셋\_반복\_호출\_데이터(스케줄 방식)

메모

데이터셋 방식

서비스 방식

스케줄 방식

맞춤 쿼리

원격 데이터 서비스

반복 형식

10

초

상태

☒ 사용 ☐ 중지

다음

취소

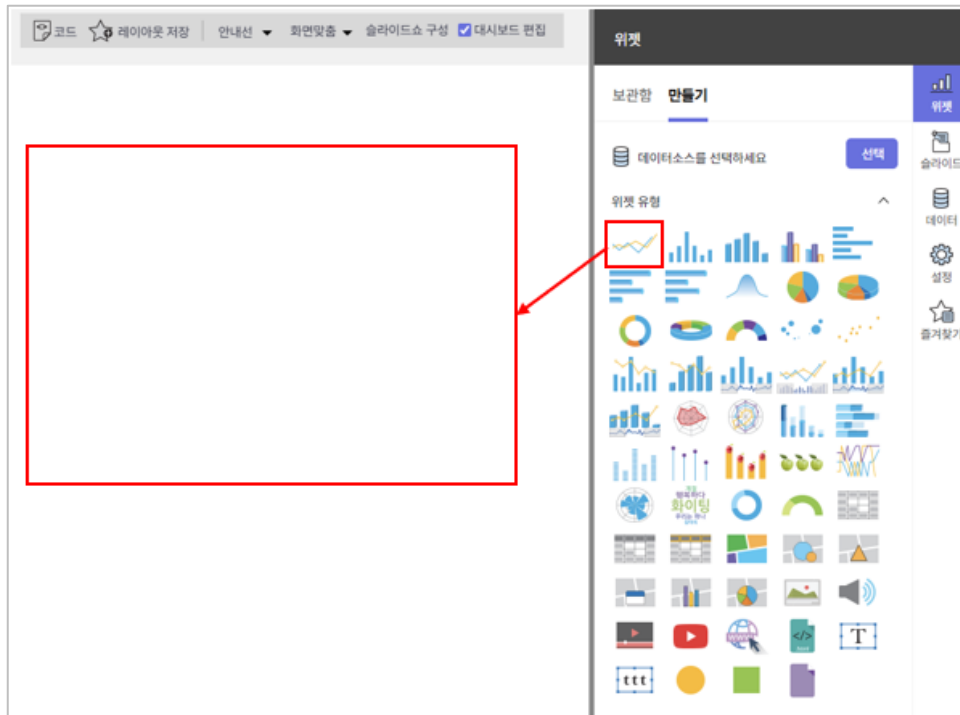
3. 스케줄 방식이 적용된 데이터셋이 생성되었습니다.


</> 데이터셋_반복_호출_데이터(스케줄 방식)					...
상태	사용자 권한	그룹 권한	만든 정보	수정 정보	
사용	없음	없음	bixsoft 2024-01-08 14:05	bixsoft 2024-01-08 19:26	
1586 Hits			데이터 보기	편집하기	

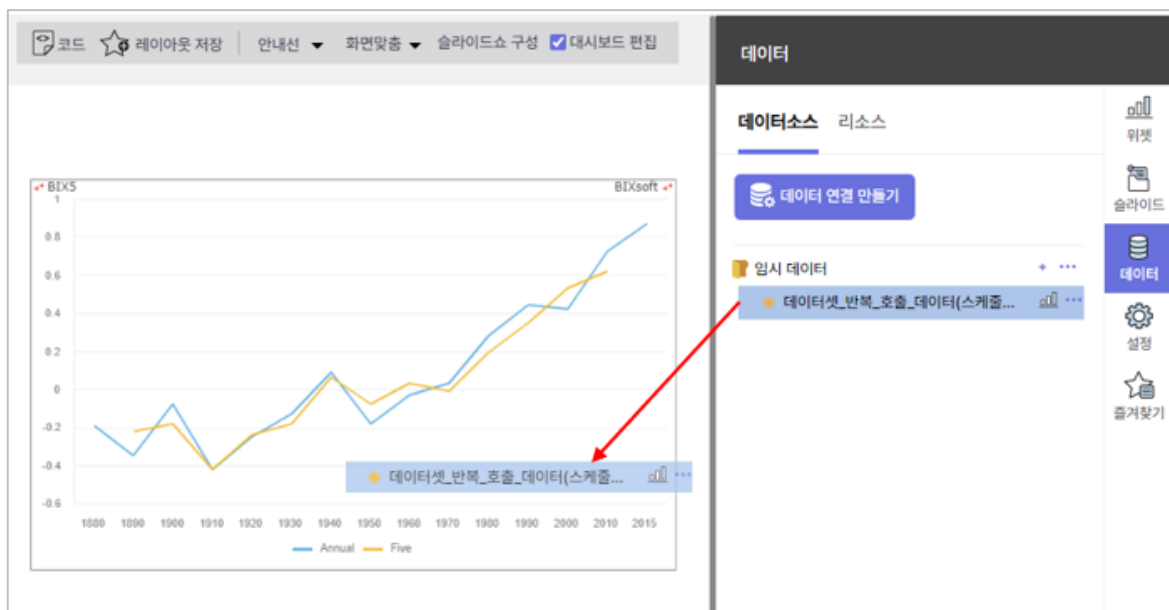
### 2.1.2. 위젯에 데이터 바인딩 하기

아래의 순서대로 진행하는 것을 권장해 드립니다.

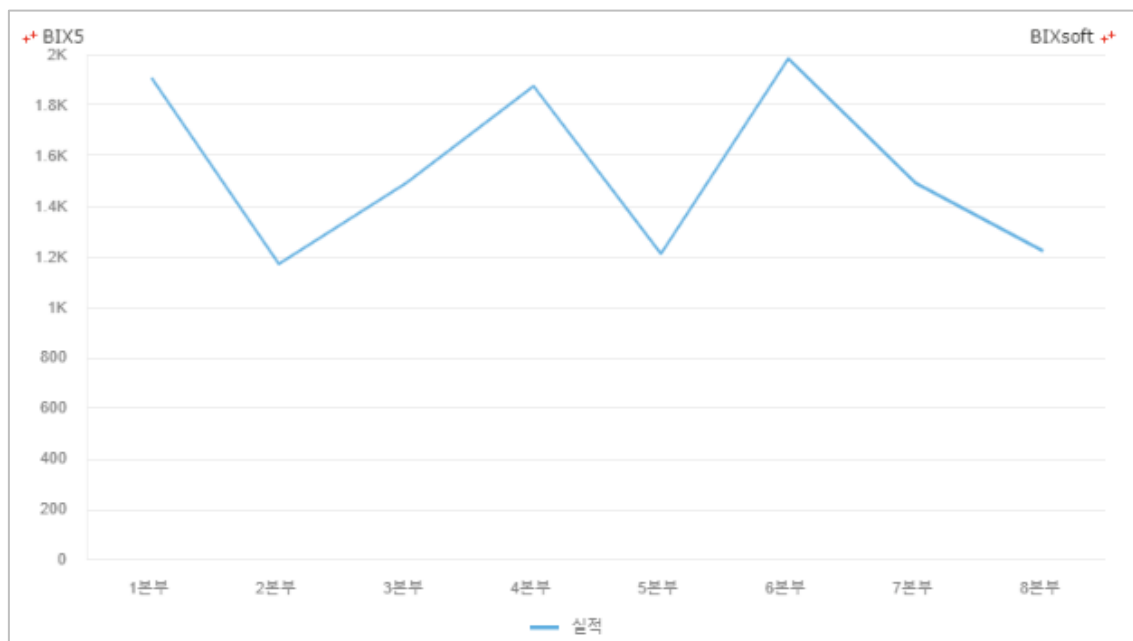
1. 데이터를 표출하고자 하는 위젯을 생성합니다. 사이드바에서 드래그 혹은 더블 클릭 등으로 생성 가능합니다.



2. 위젯에 [2.1.1.데이터셋 생성 및 설정하기](#)에서 만든 데이터셋을 바인딩 합니다. 사이드바에서 드래그 혹은  버튼을 클릭하여 바인딩 할 수 있습니다.



3. 정상적으로 적용이 되었다면 10 초마다 데이터가 업데이트되어 차트에 반영됩니다.



## 2.2. 자바스크립트 interval 방식

### 2.2.1. 데이터셋 생성 및 설정하기

아래의 순서대로 진행하는 것을 권장해 드립니다.

1. [데이터셋 방식]은 맞춤 쿼리 또는 테이블 둘 중 하나를 선택하여 생성합니다. [스케줄 방식]의 콤보박스를 [없음]으로 선택합니다.

데이터셋

이름

데이터셋\_반복\_호출\_데이터(인터벌 방식)

메모

데이터셋 방식

테이블

서비스 방식

원격 데이터 서비스

스케줄 방식

없음

상태

☒ 사용
 ☐ 중지

다음

취소

2. 데이터셋이 생성되었습니다.

</>

데이터셋\_반복\_호출\_데이터(인터벌 방식)

...

상태	사용자권한	그룹권한	만든정보	수정정보
사용	없음	없음	bixsoft 2024-01-08 13:52	bixsoft 2024-01-08 16:58

4430 Hits

데이터 보기

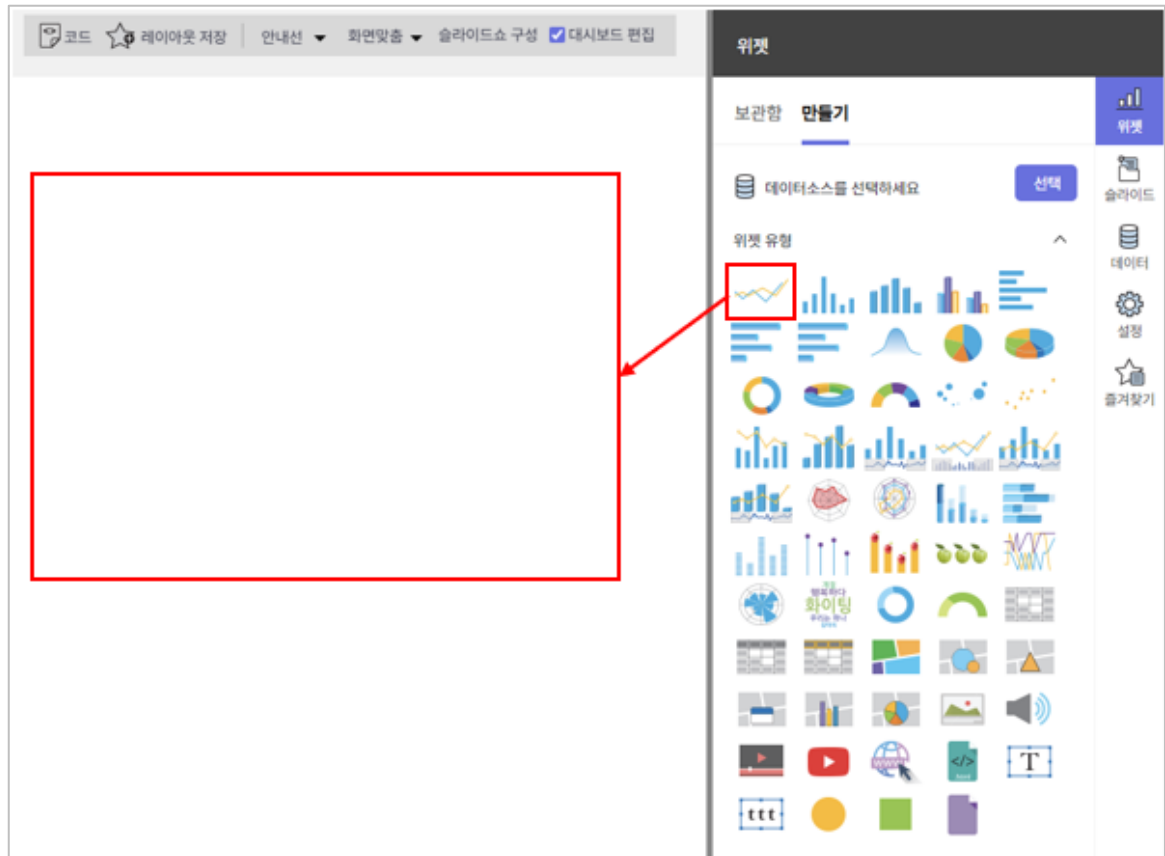
편집하기




### 2.2.2. 위젯에 데이터 바인딩 하기

아래의 순서대로 진행하는 것을 권장해 드립니다.

1. 데이터를 표출하고자 하는 위젯을 생성합니다. 사이드바에서 드래그 혹은 더블 클릭 등으로 생성 가능합니다.



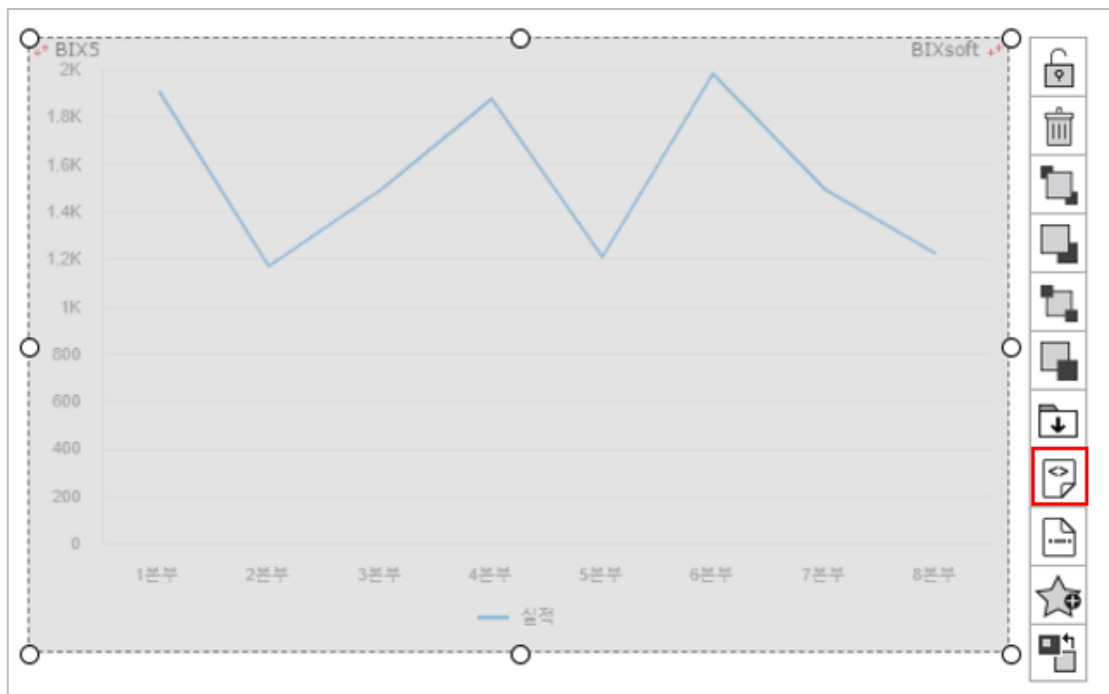
4. 위젯에 [2.2.1.데이터셋 생성 및 설정하기](#)에서 만든 데이터셋을 바인딩 합니다. 사이드바에서 드래그 혹은  버튼을 클릭하여 바인딩 할 수 있습니다.



### 2.2.3. 자바스크립트에서 바인딩 된 데이터셋 interval 호출

아래의 순서대로 진행하는 것을 권장해 드립니다.

1. 위젯의 [코드]를 클릭합니다.



2. [스크립트] 탭을 클릭합니다.

레이아웃	스크립트	컴포넌트 옵션	사용자 데이터
<pre> 1 &lt;script&gt; 2 /** 3  * component를 사용할 준비가 되면 호출합니다. 4  */ 5 widget.componentReadyHandler = function(componentId){ 6 7 } 8 9 /** 10 * 데이터 변형 사용자 정의 함수를 등록합니다. 11 * 콜백함수에 인자로 topic, payload가 전달됩니다. 12 * payload: 수신된 데이터. 해당 값은 참조된 값이기 때문에 유의하여 사용 바랍니다. 13 */ 14 /** 15 widget.setTransformSourceHandler(function(topic, payload){ 16     return payload; 17 }); 18 */ 19 20 /** 21 * component가 삭제되기 전 호출됩니다. 22 */ 23 widget.componentRemoveHandler = function(){ 24 25 } 26 &lt;/script&gt; </pre>			

다음과 같이 작성합니다.

```

1 <script>
2 /**
3  * component를 사용할 준비가 되면 호출합니다
4  */
5 widget.componentReadyHandler = function(componentId){
6     if(widget.getBindingSourceObject()){ //데이터 바인딩이 되어있을 경우 데이터셋을 호출합니다.
7         clearInterval(widget.interval);
8         widget.interval = setInterval(widget.rdsServiceFunc, 10000); //10초마다 반복
9     }
10 }
11
12 widget.rdsServiceFunc = function(){
13     //데이터셋을 다시 호출합니다.
14     widget.getBindingSourceObject().setParam({});
15     widget.getBindingSourceObject().serviceStart();
16 }
17
18 /**
19  * 데이터 변형 사용자 정의 함수를 등록합니다.
20  * 콜백함수에 인자로 topic, payload가 전달됩니다.
21  * payload: 수신된 데이터. 해당 값은 참조된 값이기 때문에 유의하여 사용 바랍니다.
22  */
23 /**
24     widget.setTransformSourceHandler(function(topic, payload){
25
26         return payload;
27     });
28 */
29
30 /**
31  * component가 삭제되기 전 호출됩니다.
32  */
33 widget.componentRemoveHandler = function(){
34     clearInterval(widget.interval);
35 }
36 </script>
37

```

사용된 스크립트

```

<script>
    /**
     * component 를 사용할 준비가 되면 호출합니다
     */
    widget.componentReadyHandler = function(componentId){
        if(widget.getBindingSourceObject()){ //데이터 바인딩이 되어있을 경우
            데이터셋을 호출합니다.
            clearInterval(widget.interval);
            widget.interval = setInterval(widget.rdsServiceFunc, 10000);
            //10 초마다 반복
        }
    }
}

```

```
widget.rdsServiceFunc = function(){
    //데이터셋을 다시 호출합니다.
    widget.getBindingSourceObject().setParam({});
    widget.getBindingSourceObject().serviceStart();
}

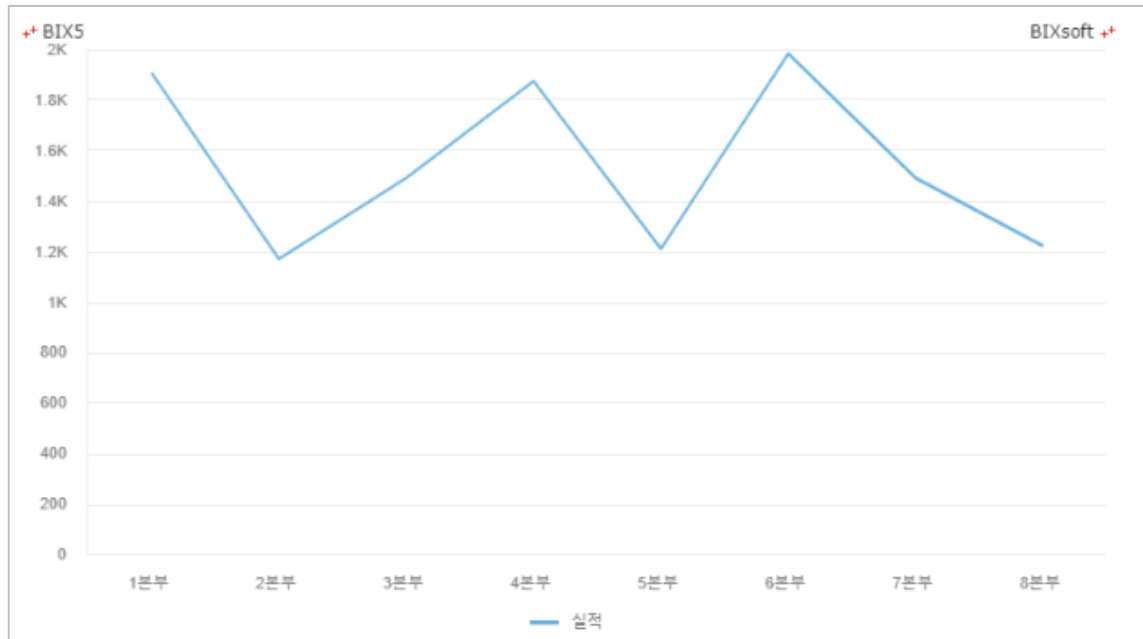
/**
 * 데이터 변형 사용자 정의 함수를 등록합니다.
 * 콜백함수에 인자로 topic, payload 가 전달됩니다.
 * payload: 수신된 데이터. 해당 값은 참조된 값이기 때문에 유의하여 사용
바랍니다.
 */

/**
widget.setTransformSourceHandler(function(topic, payload){

    return payload;
});
*/

/**
 * component 가 삭제되기 전 호출됩니다.
 */
widget.componentRemoveHandler = function(){
    clearInterval(widget.interval);
}
</script>
```

3. 정상적으로 적용이 되었다면 10 초마다 데이터가 업데이트되어 차트에 반영됩니다. 만약, 반영되지 않을 경우 콘솔 에러를 확인하세요.



감사합니다