

CH 14

폰갭 API를 활용한 앱 만들기 I

1. 폰갭 API
 2. 장치관리 하이브리드앱
 3. 사진관리 하이브리드앱
- 요약

>> 학습목표 <<

- ❖ 다양한 폰갭 API 종류를 알아본다.
- ❖ 폰갭 API를 활용한 장치관리 하이브리드앱을 개발해본다.
- ❖ 폰갭 API를 활용한 사진관리 하이브리드앱을 개발해본다.

1.1 폰갭 API 개요

● 폰갭 API

- 웹앱이 제공하지 못하는 모바일 장치의 센서, 주소록, 로컬 파일 등의 제어 기능은 폰갭 API를 통해 지원
- 폰갭 API는 모바일 브라우저의 자바스크립트 기능을 확장한 것

● 폰갭 API 종류

API 종류	기능
Accelerometer	모션 센서의 가속도 정보를 제공
Camera	카메라 장치 통해 촬영 및 사진 제공
Capture	오디오, 이미지, 비디오 정보 일부를 캡처(녹음/녹화)
Compass	나침반 센서를 통해 방향 정보 제공
Connection	네트워크의 유/무선망 연결 정보 제공
Contacts	연락처 정보의 추가/삭제/검색 제공
Device	장치의 하드웨어/소프트웨어 정보를 제공
Events	폰갭 앱의 실행 중 발생하는 이벤트 처리 지원
File	파일 읽기/쓰기/이동/복사/삭제 지원
Geolocation	GPS 센서를 통해 위치 정보 제공
Media	오디오 파일 재생/녹음 지원
Notification	장치의 알림/진동/경고음 제공
Storage	장치 저장소(데이터베이스) 저장/검색 지원

2.1 장치관리 앱 : deviceApp

● 장치관리 앱 화면 구성

- 첫 번째 화면 : 장치관리 앱 실행한 실물 단말기로부터 장치 기본 정보를 획득하여 표시
- 두 번째 화면 : 다양한 폰갭 API 이용, 각종 장치를 제어하거나 장치의 상태 정보를 요청



2.2 단말기 정보 확인하기

● document.addEventListener() 메소드

- 특정 이벤트가 발생했을 때 실행되는 콜백 함수(이벤트 리스너)를 지정

```
document.addEventListener('이벤트이름', 콜백함수, false)
```

- 제거는 removeEventListener() 메소드를 사용

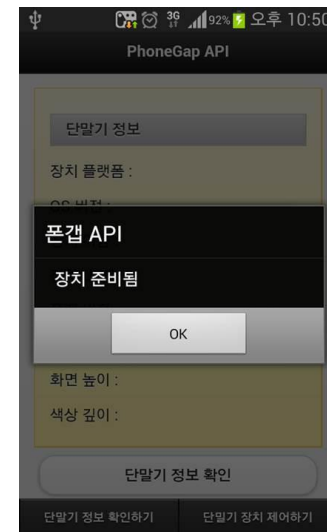
● deviceready 이벤트

- 폰갭 프레임워크의 초기화가 완료되어 폰갭 API를 사용할 수 있게 되었음을 알리기 위한 이벤트

[예제14-2] 단말기 장치 준비 확인하기

index.html의 일부

```
$(document).ready(function() {
    initPhoneGap();
    $('#btnDeviceInfo').click(function() {
        getDeviceInfo();
        ... 생략 ...
    });
    function initPhoneGap() {
        document.addEventListener('deviceready', onDeviceReady, true);
    }
    function onDeviceReady() {
        navigator.notification.alert('장치 준비됨', null, '폰갭 API');
    }
}
```

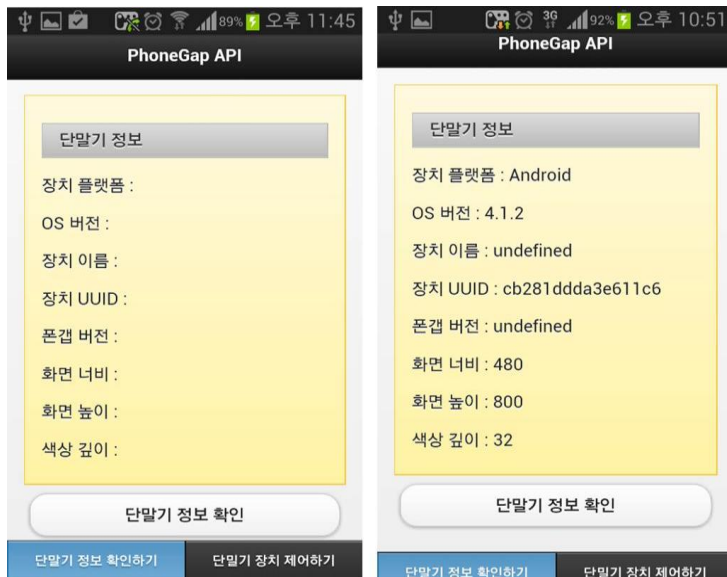


단말기 정보 얻어오기

● 폰갭 device 객체 속성

속성	의미
device.platform	플랫폼 종류
device.version	플랫폼 버전
device.name	단말기 모델명
device.uuid	단말기 고유 식별자(UUID;Universally Unique Identifier)
device.phonegap	폰갭 버전
screen.width/height	모니터 해상도 너비/높이
screen.colorDepth;	픽셀당 색상 표현 비트 수

● [그림 14-2] deviceApp의 단말기 정보 확인



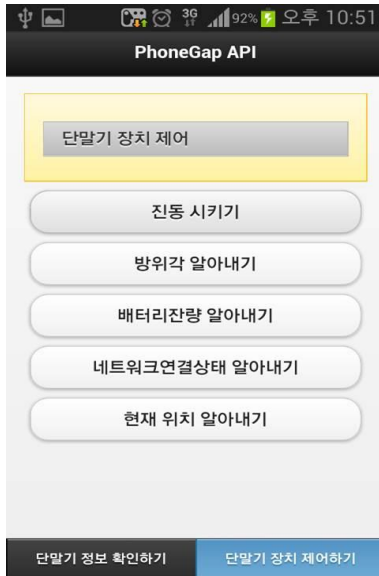
[예제14-3] 단말기 정보 얻어오기

myDeviceInfo.js의 일부

```
function getDeviceInfo() {
    var msgInfo =
        '<p>장치 플랫폼 : ' + device.platform + '</p>'
        + '<p>OS 버전 : ' + device.version + '</p>'
        + '<p>장치 이름 : ' + device.name + '</p>'
        + '<p>장치 UUID : ' + device.uuid + '</p>'
        + '<p>폰갭 버전 : ' + device.phonegap + '</p>'
        + '<p>화면 너비 : ' + screen.width + '</p>'
        + '<p>화면 높이 : ' + screen.height + '</p>'
        + '<p>색상 깊이 : ' + screen.colorDepth; + '</p>'
    $('#infoArea').html(msgInfo);
};
```

2.3 진동 시키기

- [그림 14-3] deviceApp의 단말기 장치 제어(진동 시키기)



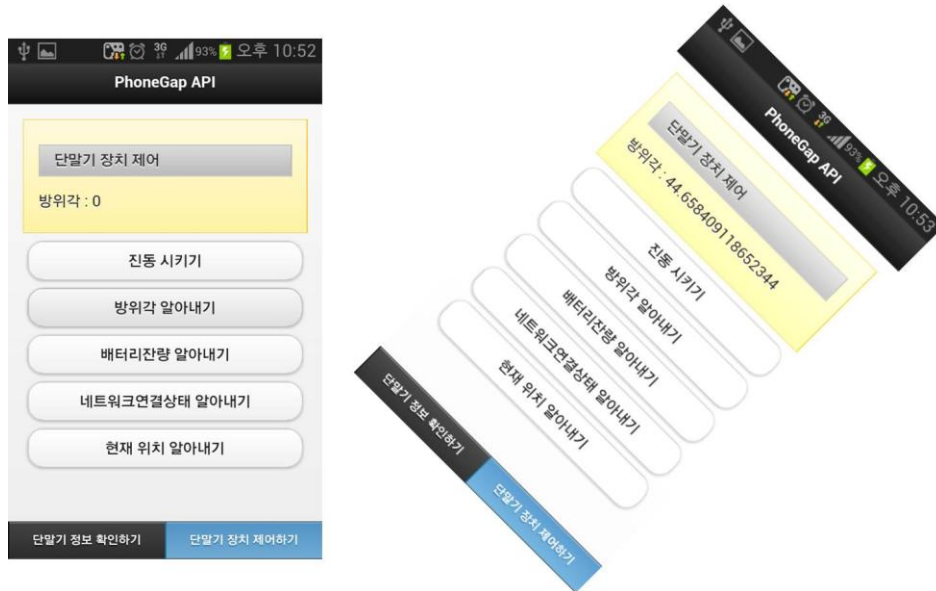
[예제14-4] 단말기 진동시키기

myDeviceInfo.js의 일부

```
function getVibrate() {
    navigator.notification.vibrate(1000);
};
```

2.4 방위각 정보 확인하기

- [그림 14-4] deviceApp의 단말기 정보 확인(방위각 알아내기)



[예제14-5] 단말기 방위각 얻어오기

myDeviceInfo.js의 일부

```
function getCompass() {
  navigator.compass.getCurrentHeading(
    function(heading) {
      $('#resultArea').html('<p>방위각 : ' + heading.trueHeading + '</p>');
    },
    function() {
      $('#resultArea').html('<p>방위각 오류</P>');
    });
};
```


2.5 배터리 정보 확인하기

● [그림 14-5] deviceApp의 단말기 정보 확인(배터리 잔량 알아내기)

● 폰갭 배터리 관련 이벤트

- 경고 상태일 때 : batterylow 이벤트
- 심각 상태일 때 : batterycritical 이벤트
- 배터리 충전 상태가 변할 때 : batterystatus 이벤트



[예제14-6] 단말기 배터리 정보 얻어오기

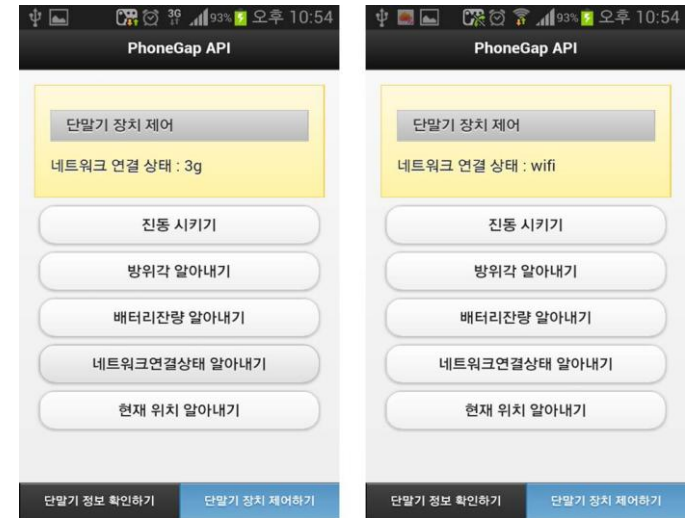
myDeviceInfo.js의 일부

```
function getBattery() {
    document.addEventListener('batterystatus', function(info) {
        $('#resultArea').html('<p>배터리 잔량 : ' + info.level + ' %</p>');
    }, false);
};
```

2.6 네트워크 연결 정보 확인하기

- [그림 14-6] deviceApp의 단말기 정보 확인(네트워크 상태 알아내기)
- 폰갭 이벤트 유형

이벤트 종류	이벤트 유형
장치 초기화 이벤트	deviceready
앱 상태 이벤트	pause, resume
네트워크 상태 이벤트	online, offline
버튼 동작 이벤트	backbutton, menubutton, searchbutton
배터리 관련 이벤트	batterylow, batterycritical, batterystatus



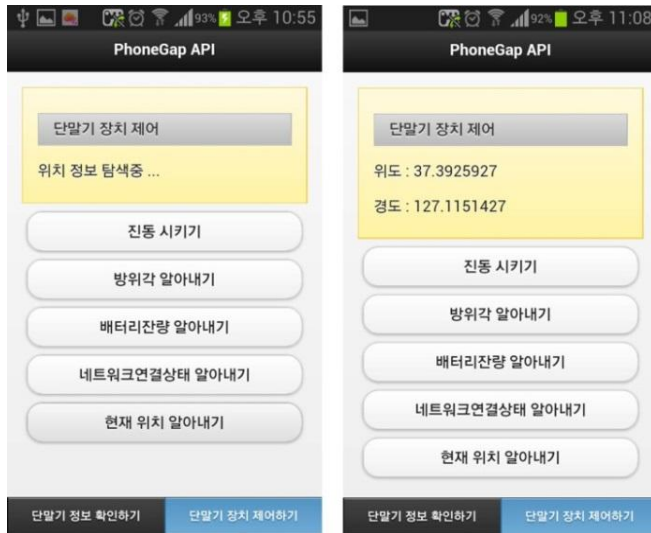
[예제14-7] 단말기 네트워크 정보 얻어오기

myDeviceInfo.js의 일부

```
// 네트워크 연결 상태
function getConnection() {
    var status = navigator.network.connection.type
    $('#resultArea').html('<p>네트워크 연결 상태 : ' + status + '</p>');
};
```

2.7 위치 정보 확인하기

● [그림 14-7] deviceApp의 단말기 정보 확인(현재 위치 알아내기)



[예제14-8] 단말기 위치 정보 얻어오기

myDeviceInfo.js의 일부

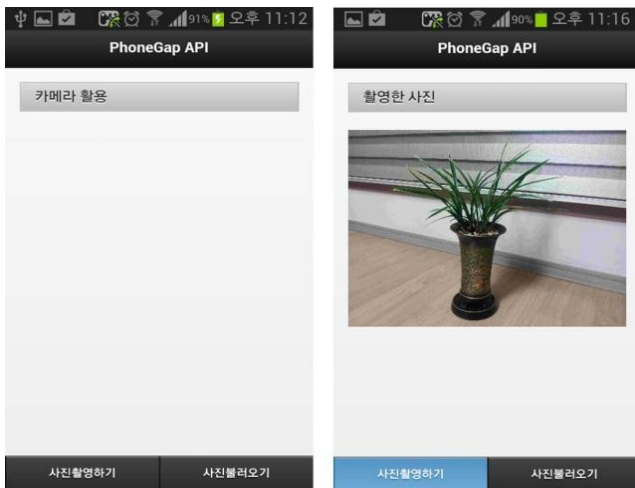
```
// 현재 위치 정보
function getLocation() {
    $('#resultArea').html('<p> 위치 정보 탐색중 ...</p>');
    navigator.geolocation.getCurrentPosition(
        function(Position) {
            $('#resultArea').html('<p>위도 : ' + Position.coords.latitude + '</p>'
                + '<p>경도 : ' + Position.coords.longitude + '</p>');
        },
        function(posError) {
            $('#resultArea').html('<p> 위 치 정 보   오 류 : ( ' + posError.code + ' ) ' +
                posError.message + '</P>');
        },
        { maximumAge: 3000, timeout: 50000, enableHighAccuracy: true }
    );
};
```

● [표 14-3] getCurrentPosition() 메소드의 세 번째 인자, 선택사항(options)

속성	의미
maximumAge	실제 위치 정보를 대신할 이전 캐시 정보의 최대 유효 지속 시간
timeout	콜백 함수 호출되기까지의 최대 대기시간
enableHighAccuracy	앱 정밀값 측정 여부(true로 설정)
frequency	위치 정보 측정 주기

3.1 사진관리 앱 : cameraApp

● [그림 14-8] cameraApp의 화면 구성



[예제14-9] 사진관리 앱 초기 화면 생성하기

index.html의 일부

```
<style>
    #displayArea { display: block; width: 100%; }           // ②
</style>
    ... 생략 ...
<div data-role="page" id="page1" data-position="fixed">
    <div data-role="header">
        <h1>PhoneGap API</h1>
    </div>
    <div data-role="content">
        <div class="ui-bar ui-bar-d">
            <h3>카메라 활용</h3>
        </div> <br/>
        <img id="displayArea" src=""/>                       // ①
    </div>
    <div data-role="footer" data-position="fixed">
        <div data-role="navbar">
            <ul>
                <li><a id="btnTakePhoto">사진촬영하기</a></li>           // ③
                <li><a id="btnGetPhoto">사진불러오기</a></li>
            </ul>
        </div>
    </div>
</div>
```

3.2 사진 촬영하기[1]

● 단말기 장치 준비 확인

[예제14-10] 단말기 장치 준비 확인하기

index.html의 일부

```
$(document).ready(function() {
    initPhoneGap();
    $('#btnTakePhoto').click(function() {
        takePhoto();
        ...        생략 ...
    });
    function initPhoneGap() {
        document.addEventListener('deviceready', onDeviceReady, true);
    }
    function onDeviceReady() {
        navigator.notification.alert('장치 준비됨', null, '폰갭 API');
    }
}
```

● 사진 촬영

[예제14-11] 사진 촬영하기

myCamera.js의 일부

```
// 사진 촬영
function takePhoto() {
    navigator.camera.getPicture(onPhotoDataSuccess, onFail,
        {quality: 10, destinationType: Camera.DestinationType.DATA_URL,
        sourceType: Camera.PictureSourceType.CAMERA });
}
```

사진 촬영 하기(2)

- navigator.camera 카메라를 제어 폰갭 객체

- getPicture() 메소드 : 스마트폰에서 제공하는 카메라 앱을 실행시켜 사진을 촬영

```
navigator.camera.getPicture( 성공콜백함수, 실패콜백함수 [, 선택사항 ] );
```

- getPicture() 함수의 세 번째 인자, 선택사항(options)

속성	속성값	특징
quality	0~100	사진 이미지의 저장 품질(백분율) 지정
destinationType	Camera.DestinationType.DATA_URL(기본값) Camera.DestinationType.FILE_URI	반환할 사진 이미지의 타입 지정 base64 인코딩 형식의 원본 이미지 데이터 이미지 파일의 위치 경로
sourceType	Camera.PictureSourceType.PHOTOLIBRARY (Camera.PictureSourceType.SAVEDPHOTOALBUM) Camera.PictureSourceType.CAMERA	사진 이미지 소스 대상을 지정 포토라이브러리 (포토앨범 : 안드로이드에서는 무의미) 카메라
mediaType	Camera.MediaType.PICTURE Camera.MediaType.VIDEO Camera.MediaType.ALLMEDIA	미디어 유형 지정 이미지 파일만 선택 동영상 파일만 선택 모든 파일 선택 가능
encodingType	Camera.EncodingType.JPEG Camera.EncodingType.PNG	사진 이미지 압축 형식 지정 JPEG 형식 요청 PNG 형식 요청
targetWidth	수치값	사진 이미지의 너비
targetHeight	수치값	사진 이미지의 높이

cameraApp의 사진 촬영 하기

- `getPicture()` 함수를 호출한 뒤 실행되는 성공 콜백 함수 `onPhotoDataSuccess()`

[예제14-12] 촬영사진 표시하기

myCamera.js의 일부

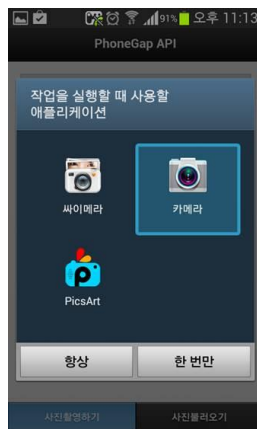
```
// 사진촬영 성공콜백함수
function onPhotoDataSuccess(imageData) {
    $('#displayArea').attr('src', 'data:image/jpeg;base64,' + imageData);
    $('h3').replaceWith($('h3>촬영한 사진</h3>'));
}
```

- `getPicture()` 함수와 `getPhoto()` 함수 호출한 뒤 실패 했을 때 실행되는 콜백 함수

[예제14-13] 사진 촬영/검색 오류 처리하기

myCamera.js의 일부

```
// 사진촬영/검색 실패콜백함수
function onFail(message) {
    alert('실패 : ' + message);
}
```



3.3 사진 불러오기

● cameraApp의 사진 불러 오기

[예제14-14] 사진 불러오기

myCamera.js의 일부

```
// 사진 촬영 검색
function getPhoto() {
    navigator.camera.getPicture(onPhotoURISuccess, onFail,
        { quality: 50, destinationType: Camera.DestinationType.FILE_URI,
          sourceType: Camera.PictureSourceType.PHOTOLIBRARY });
}
```

[예제14-15] 검색 사진 표시하기

myCamera.js의 일부

```
// 사진검색 성공콜백함수
function onPhotoURISuccess(imageURI) {
    $('#displayArea').attr('src', imageURI);
    $('h3').replaceWith($(' <h3>불러온 사진 </h3> '));
}
```

