

웹프로그래밍

목차

A table of contents

- 1 프로젝트 주제 및 목적
- 2 필요한 기능 및 구현 방법
- 3 시연
- 4 향후 발전 계획
- 5 느낀점



주제 : 약국 위치 및 정보를 제공

목적 : 사용자가 약국에 가야 할 경우,
약국의 위치를 알려주어 헤매는 것을 방지 및
정보를 알려주어 아쉬운 발걸음 하는 것을 줄인다.

2 필요한 기능 및 구현 방법

1. 메뉴

특정 카테고리에 묶인 여러 개의 목록들을 평상시에는 보이지 않게 하다 어떤 동작을 통하여 보이게 하는 기능



설명:

상단의 4개의 메뉴를 1차 메뉴라고 지정

마우스를 각각의 1차 메뉴 위에 위치할 시 Hover 기능과 자식선택자 기능을 이용한 2차 메뉴 즉 (코로나 현황, 날씨 현황) 서브메뉴가 보인다.

```
<div class="dmenu">
  <div data="menu-1">
    <div>
      <a href="#">공지사항</a>
      <a href="http://localhost/wd.html">공지사항</a>
      <a href="http://localhost/wd.html">약국 어디?</a>
    </div>
    <div>
      <a href="http://localhost/wd.html">후기게시판</a>
    </div>
    <div>
      <a href="#">약국 뉴스</a>
      <a href="http://localhost/wd.html">백두상단 캠페인</a>
      <a href="http://localhost/wd.html">약국 어디?</a>
      <a href="http://localhost/wd.html">심장 약국 뉴스</a>
    </div>
    <div>
      <a href="#">코로나, 날씨</a>
      <a href="http://localhost/wd.html">코로나 현황</a>
      <a href="http://localhost/wd.html">날씨 현황</a>
    </div>
  </div>
</div>
```

```
/* 마우스가 올라진 메뉴아이템의 자식인 텍스트 */
.menu-1 ul > li: hover > a {
  background-color: black;
  color: white;
  z-index: 1;
}

/* 서브메뉴 */
.menu-1 > ul ul {
  display: none;
  /* 유행화 */
  position: absolute;
  top: 100%;
  left: 0;
}

/* 마우스가 올라진 메뉴아이템의 자식인 메뉴 */
.menu-1 ul > li: hover > ul {
  display: block;
}
```

```
/* 메뉴 엘리먼트 */
.menu-1 {
  text-align: center;
}

/* 메뉴 */
.menu-1 ul {
  width: 220px;
  background-color: #d4d4d4;
}

/* 1차 메뉴 */
.menu-1 > ul {
  width: 1000px;
  display: inline-block;
  padding: 0 10px;
  border-radius: 5px;
}

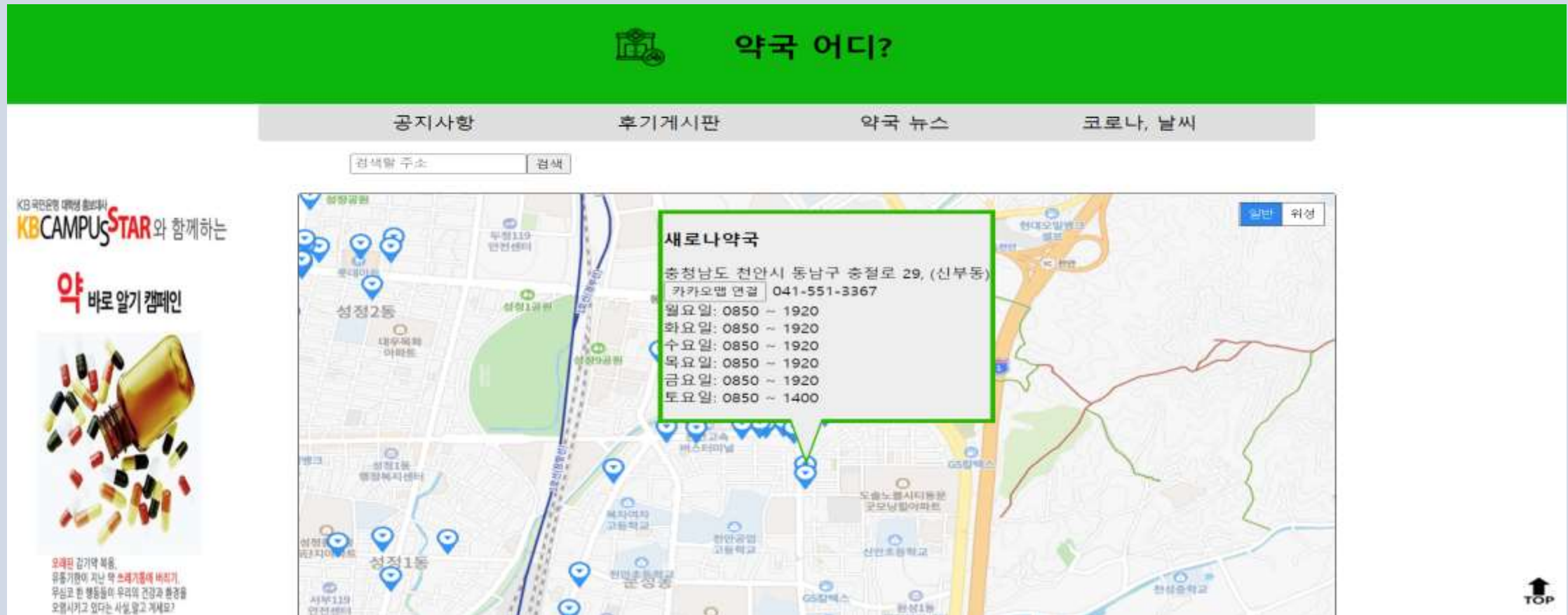
/* 1차 메뉴 아이템 */
.menu-1 > ul > li {
  display: inline-block;
  position: relative;
}

/* 메뉴아이템 텍스트 */
.menu-1 ul > li > a {
  width: 220px;
  display: block;
  padding: 10px;
}
```

2 필요한 기능 및 구현 방법

2. 지도위에 약국 위치 정보 표시하기

지도 위에 open api 에서 받아온 정보를 가공하여
마커로 표시하는 기능



2 필요한 기능 및 구현 방법

2. 지도위에 약국 위치 정보 표시하기

지도 위에 open api 에서 받아온 정보를 가공하여
마커로 표시 하는 기능

<!-- 네이버 지도 삽입 -->

<script>

\$(document).ready(async function() {

let XY = await getLocation();

// 지도를 삽입할 HTML 요소 또는 HTML 요소의 id를 지정합니다.

var map = new naver.maps.Map("map", {

center: new naver.maps.LatLng(XY.lat, XY.lng),

zoom: 15,

mapTypeControl: true

});

```
async function getLocation() {  
  let XY = new Object();  
  if(navigator.geolocation) {  
  
    let promise = new Promise((resolve, rejected) => {  
      navigator.geolocation.getCurrentPosition((position) => {  
        resolve(position);  
      });  
    });  
  
    let position = await promise;  
    XY.lat = position.coords.latitude;  
    XY.lng = position.coords.longitude;  
  }  
  return XY;  
}  
</script>
```

좌측 사진은 네이버 지도api를
사용하여 지도를 삽입한 코드입니다.

우측 사진은 현재의 위치 정보 코드로
getCurrentPosition을 사용하여 현재 위
치 정보를 받아온 뒤
object형으로 선언된 변수 XY의
XY.lat 에다가는 위도를,
XY.lng 에다가는 경도를 저장한 후
return을 통해 현재 좌표의 위도와
경도를 얻을수 있습니다.

2

2. 지도위에

지도 위에 open api 에서 받아온 정보를 가공하여
마커로 표시 하는 기능

```
app.get("/phonebook_list", (req, res) => {  
    let api = async() => {  
        let response = null;  
        try {  
            response = await axios.get("http://apis.data.go.kr/655267/ErectInstInfolnquireService/getParamcyListInfolnquire", {  
                params: {  
                    "serviceKey": "yourAppKey%4&nZy)/j4NS585qibjbcJfgepbQvckFpl/vclwApilzBkIEWpaleKlbasoXPFJcsw=",  
                    "@@": req.query["@@"],  
                    "@@": req.query["@@"],  
                    "@@": req.query["@@"],  
                    "@@": req.query["@@"],  
                    "@@": req.query["@@"],  
                    "@@": req.query["@@"],  
                    "pageNo": req.query.pageNo,  
                    "numOfRows": req.query.numOfRows  
                },  
                //headers: {"Content-type": "application/text; charset=utf-8"},  
                headers: {"Accept-Encoding": "application/json"},  
                //options: {charset: "au-kr", responseType: "au-kr"}  
            })  
        }  
        catch(e) {  
            console.log(e);  
        }  
        return response;  
    }  
    api().then((response) => {  
        res.setHeader("Access-Control-Allow-Origin", "");  
        res.json(response.data.response.body);  
    });  
});
```

```
$ajax({  
    url: "/pharmach_list",  
    type: "GET",  
    cache: false,  
    dataType: "json",  
    data: {"Q0": "sido", "Q1": "", "Q2": "", "Q3": "", "Q4": "", "Q5": "", "Q6": "", "Q7": "", "Q8": "", "Q9": "", "Q10": "", "Q11": "", "Q12": "", "Q13": "", "Q14": "", "Q15": "", "Q16": "", "Q17": "", "Q18": "", "Q19": "", "Q20": "", "Q21": "", "Q22": "", "Q23": "", "Q24": "", "Q25": "", "Q26": "", "Q27": "", "Q28": "", "Q29": "", "Q30": "", "Q31": "", "Q32": "", "Q33": "", "Q34": "", "Q35": "", "Q36": "", "Q37": "", "Q38": "", "Q39": "", "Q40": "", "Q41": "", "Q42": "", "Q43": "", "Q44": "", "Q45": "", "Q46": "", "Q47": "", "Q48": "", "Q49": "", "Q50": "", "Q51": "", "Q52": "", "Q53": "", "Q54": "", "Q55": "", "Q56": "", "Q57": "", "Q58": "", "Q59": "", "Q60": "", "Q61": "", "Q62": "", "Q63": "", "Q64": "", "Q65": "", "Q66": "", "Q67": "", "Q68": "", "Q69": "", "Q70": "", "Q71": "", "Q72": "", "Q73": "", "Q74": "", "Q75": "", "Q76": "", "Q77": "", "Q78": "", "Q79": "", "Q80": "", "Q81": "", "Q82": "", "Q83": "", "Q84": "", "Q85": "", "Q86": "", "Q87": "", "Q88": "", "Q89": "", "Q90": "", "Q91": "", "Q92": "", "Q93": "", "Q94": "", "Q95": "", "Q96": "", "Q97": "", "Q98": "", "Q99": ""},  
    success: function(data) {  
        console.log(data);  
    }  
});
```

약국의 정보가 들어있는 api의 사용을 위한 코드로 app.get을 이용하여 사용하였으며

Ajax함수를 이용하여 비동기 방식으로 약국의 데이터를 받아왔습니다.

2 필요한 기능 및 구현 방법

2. 지도 위에 약국 위치 정보 표시하기

지도 위에 open api 에서 받아온 정보를 가공하여
마커로 표시 하는 기능

```
var contentString = [
    '<div class="div_line">',
    '<div>+dutyName+</div>',
    '<p>+dutyAddr+</p>',
    '<button onclick="window.open(\''+https://map.kakao.com/link/to'+dutyName+', '+it.wgs84Lat+', '+it.wgs84Lon+'&quot;);>카카오맵 연결</button>',
    '+dutyTel+<br />',
    '+dutyTime',
    '</p>',
    '</div>'
].join("");

var infoWindow = new naver.maps.InfoWindow({
    content: contentString,
    maxWidth: 440,
    backgroundColor: "#eee",
    borderColor: "#ddd",
    borderWidth: 5,
    anchorSize: new naver.maps.Size(30, 30),
    anchorSkew: true,
    anchorColor: "#eee",
    pixelOffset: new naver.maps.Point(20, -20)
});
```

```
data.items.item.forEach(function(it, index){
    let dutyName = it.dutyName; //약국명
    let dutyAddr = it.dutyAddr; //주소
    let dutyTel = it.dutyTel; //전화번호

    let dutyTime = "";
    if(it.dutyTime1s && it.dutyTime1c) {
        dutyTime += "필요일: " + it.dutyTime1s + " ~ " + it.dutyTime1c + "<br>";
    }
    if(it.dutyTime2s && it.dutyTime2c) {
        dutyTime += "화요일: " + it.dutyTime2s + " ~ " + it.dutyTime2c + "<br>";
    }
    if(it.dutyTime3s && it.dutyTime3c) {
        dutyTime += "수요일: " + it.dutyTime3s + " ~ " + it.dutyTime3c + "<br>";
    }
    if(it.dutyTime4s && it.dutyTime4c) {
        dutyTime += "목요일: " + it.dutyTime4s + " ~ " + it.dutyTime4c + "<br>";
    }
    if(it.dutyTime5s && it.dutyTime5c) {
        dutyTime += "금요일: " + it.dutyTime5s + " ~ " + it.dutyTime5c + "<br>";
    }
    if(it.dutyTime6s && it.dutyTime6c) {
        dutyTime += "토요일: " + it.dutyTime6s + " ~ " + it.dutyTime6c + "<br>";
    }
    if(it.dutyTime7s && it.dutyTime7c) {
        dutyTime += "일요일: " + it.dutyTime7s + " ~ " + it.dutyTime7c + "<br>";
    }
    if(it.dutyTime8s && it.dutyTime8c) {
        dutyTime += "공휴일: " + it.dutyTime8s + " ~ " + it.dutyTime8c + "<br>";
    }

    let pharmacy_location = new naver.maps.LatLng(it.wgs84Lat, it.wgs84Lon);
```

마커에 표시할 내용으로
contentString 으로 선언된 변수에
Open api에서 제공하고있는
약국의 이름, 주소, 전화번호,
운영시간을 넣었습니다.

이렇게 완성된 정보는 InfoWindow
함수를 통해 보여지게 됩니다.

2 필요한 기능 및 구현 방법

2. 지도위에 약국 위치 정보 표시하기

지도 위에 open api 에서 받아온 정보를 가공하여
마커로 표시 하는 기능

```
let marker = new naver.maps.Marker({  
  map: map,  
  position: pharmacy_location  
});
```

```
naver.maps.Event.addListener(marker, "click", function(e) {  
  if (infowindow.getMap()) {  
    infowindow.close();  
  } else {  
    infowindow.open(map, marker);  
  }  
});
```

마커를 생성하는 코드이며
addListener를 사용하여 click시
앞서 말한 약국의 정보창이
보여지게 하였습니다.

1. 마커만 보았을 때 해당약국이 운영하고 있는지 안하고 있는지 추가할 계획입니다.
2. 지도 왼쪽 상단의 주소 검색창에서 주소를 검색하면 해당 주소의 위치로 이동하게 된다. 이때 만약 입력한 주소의 지역이 근처가 아닌 다른 곳이라면 기기의 접속위치정보를 받아 근처의 약국정보를 띄어주는 지도에서는 검색한 주소근처의 약국이 표시되지 않는다.

=>전국 약국을 접속한 위치에 상관없이 표시하게 되면 이러한 문제를 해결 할 수 있으나 무수히 많은 마커들이 찍히기에 정보를 표시하는데 상당한 시간이 소요, 따라서 접속 위치정보를 검색한 주소에 따라 바뀌게 할 계획입니다.

문제를 직면하면 혼자가 아닌 팀원과 고민하고 해결하여 더 좋은 방향으로 문제를 풀어나갈 수 있었고, 프로젝트를 진행하며 open api를 이용한 여러가지 기능들을 새로 학습하고 직접 적용하며 많은 성장을 할 수 있는 기회였던것 같습니다.

20192266 부성현

프로젝트를 진행하면서 어려운 부분이나 결과물이 더 좋게 나오기 위해 팀원과 소통하며 진행한 점이 좋았고, open api를 사용하여 프로젝트를 진행하다보니 적응 되었고 웹프로그래밍에 대해 더 잘 이해하며 한 단계 더 성장한 것 같습니다.

20193423 정의범

감사합니다