

# 인터넷과 웹기초 프로젝트 보고서

작성자

**201724436 김재현**

**인터넷과웹기초[060]**

## I. 사용한 API

- A. 네이버 검색 api
- B. 업비트 마켓 조회
- C. Meme api

## I. 개발 기준

- A. HTML
- B. CSS
- C. JAVASCRIPT

## II. 웹사이트 사용법

- A. 뉴스 검색
- B. 코인가격 확인
- C. 이미지 생성

## 사용한 API

### 1 네이버 검색 api

- <https://developers.naver.com/products/service-api/search/search.md>.

네이버 검색 api 입니다. nodeJS 를 사용해서 연결시켰고 params 로는 검색할 내용(target)과 가져올 뉴스 개수(num)을 받습니다. 보낼때 미리 등록해둔 X-Naver-Client-Id 와 X-Naver-Client-Secret 를 header 에 담아서 보냅니다.

```
app.get('/news/:target/:num', function (req, res) {
  var request = require('request');
  var options = {
    'url': encodeURI('https://openapi.naver.com/v1/search/news.json?query='
      + req.params.target + '&display=' + req.params.num),
    'headers': {
      'X-Naver-Client-Id': 'aJ3xkRZngumFQHAXp8zF',
      'X-Naver-Client-Secret': 'f7K_TLW4w1'
    }
  };
  request(options, function (error, response) {
    if (error) throw new Error(error);
    res.send(response.body);
  });
});
```

이때 검색어(req.params.target)에는 한글이 들어올 수 있어 encodeURI 함수를 이용한 인코딩 작업이 필요합니다.error 가 있으면 throw 하고 없으면 response.body 를 리턴합니다.

### 2 업비트 api

- <https://docs.upbit.com/reference>

업비트 api 입니다. 처음에 라디오 박스를 아래를 사용해서

```
app.get('/coinmenu', function (req, res) {
  var request = require('request');
  var options = {
    'method': 'GET',
    'url': 'https://api.upbit.com/v1/market/all',
    'headers': {
    }
  };
  request(options, function (error, response) {
    if (error) throw new Error(error);
    console.log(response.body);
    res.send(response.body);
  });
});
```

```
<script>
  //window.onload = getcoin();
</script>
```

모든 코인의 종류를 받아와서 라디오 박스에 추가할까 하였으나  
생각보다 코인의 종류가 너무많아서 정해진 코인만 가격을 가져오게  
하였습니다.

```
app.get('/coinprice/:what', function (req, res) {
  var request = require('request');
  var options = {
    'method': 'GET',
    'url': 'https://api.upbit.com/v1/ticker?markets=' + req.params.what,
    'headers': {
    }
  };
  request(options, function (error, response) {
    if (error) throw new Error(error);
    console.log(response.body);
    res.send(response.body)
  });
});
```

Params 는 코인종류(what)하나만을 쓰며 헤더는 필요 없습니다. 해당  
코인의 가격정보를 받아옵니다.

### 3 Meme api

□ <https://apimeme.com>

api 가 있길래 사용을 해보려고 하였는데 막상 사용하려고 하니까 돈을  
내라고 하길래 api 사용은 2 개를 해서 조금 다르게 사용하였습니다.

URL 에 이미지내용에 대한 정보를 넣고 그 링크를 직접  
사용하였습니다. Api 를 사용한 것은 아닙니다. Top 과 bottom 에  
들어갈 내용, 이미지 이름이 있으면 이미지를 생성할 수 있습니다.

```
function setpic() {
  document.getElementById('here').setAttribute('src', "https://apimeme.com/meme?meme=" +
    document.getElementById('meme').value + "&top=" +
    document.getElementById('top').value + "&bottom=" +
    document.getElementById('bottom').value);
}
```

# 최소 개발 기준

## 1 HTML

□ Form 의 서로 다른 type 의 input 3 개 이상 사용

### 1. 뉴스 검색시 text,search 타입 input

```
<form action="#" accept-charset="utf-8" method="set">
  <fieldset style="width:150">
    <legend>뉴스 검색</legend>
    <div>
      <input type="search" value="코인" id="searching" class="beautiful">
      <input class="pretty_button2" type="button" value="검색" id="search" onclick="displayNEWS()">
      <input type="text" value="50" id="num" style="width:25px" onkeyup="chkintCode(event)">개씩 검색
    </div>
  </fieldset>
</form>
```

### 2. 코인 선택시 radio 타입

```
<form action="#" accept-charset="utf-8" method="set">
  <fieldset style="width:110">
    <legend>관심있는 코인 하나</legend>
    <p class="chk_box">
      <div id="coins">
        <input type="radio" onclick="set_coin()" name="coin" value="KRW-BTC">비트코인
        <input type="radio" onclick="set_coin()" name="coin" value="KRW-ETH">이더리움
        <input type="radio" onclick="set_coin()" name="coin" value="KRW-DOGE">도지
        <input type="radio" onclick="set_coin()" name="coin" value="KRW-XRP">리플
      </div>
    </p>
  </fieldset>
</form>
```

### 3. 사진 검색시 text,button 타입

```
<section class="contain">
  <h1>오늘의 기분을 그림으로 그려보아요!</h1><br><br>
  <select id="meme" class="choice_box"> ...
</select><br /><br />
  <input type="text" id="top" value="HELLO" onkeyup="chkCharCode(event)"><br />
  <input type="text" id="bottom" value="ITS ME" onkeyup="chkCharCode(event)"><br />
  <input type="button" value="클릭!" onclick="setpic()"><br /><br />
  <img id="here">
  <br /><br /><br />
</section>
```

## □ List 또는 Table 1 개 이상 사용

nav 만들 때 list 사용

```
<header>
  <section class="hbody">
    <nav class="nav">
      <div class="title"><a href="project.html">재효니웹기초</a>
      </div>
      <ul id="menu">
        <li><a href="project_news.html">기사검색</a></li>
        <li><a href="project_price.html">가격검색</a></li>
        <li><a href="project_pic.html">사진만들기</a></li>
      </ul>
    </nav>
  </section>
</header>
```

## □ Header , nav , footer 사용

위 사진에서 header 에 전부 nav 를 넣어 줬고,

```
<footer>
  &copy;Kim_jae_hyeon<br />
  111-2507, 10, Busandaehak-ro, Geumjeong-gu, Busan, Republic of Korea
</footer>
```

footer 또한 적용시켰습니다.

# 2 CSS

## □ Attribute,combinator,pseudo-class,pseudo-element

### 1. Attribute Selector 2 개 이상

```
p[title] {
  color: white;
}

discription.setAttribute('title','des');
```

```
input[type="radio"]{
  display:none;
}
input[type="radio"] + label span {
  display:inline-block;
  width:19px;
  height:19px;
  margin:2px 10px 0 0;
  vertical-align:middle;
  background:url(check_radio_sheet.png) -38px top no-repeat;
  cursor:pointer;
}
input[type="radio"]:checked + label span {
  background:url(check_radio_sheet.png) -57px top no-repeat;
}
```

### 2. Combinator 5 개 이상

```
.title {
  font-size: 2em;
}

.title>a {
  color: white;
}
```

```
.article a {
  color: gray;
}
```

```
li>a {
  font-size: 10px;
  color: white;
}
```

```
div input {
  border: 2px solid black;
  background-color: white;
  color: black;
  text-align: left;
  font-size: 20px;
}
```

```
.article h3 {
  font-weight: 900;
  font-size: 20px;
  color: white;
}
```

### 3. Pseudo-class 2 개 이상

```
.pretty_button {
  border-radius: 20px;
  background-color: black;
  color: white;
}

.pretty_button:hover {
  border-radius: 20px;
  border: 2px solid white;
  color: gray;
  background-color: black;
}

.pretty_button2 {
  border: 1px solid white;
  background-color: black;
  color: white;
}

.pretty_button2:hover {
  border: 2px solid white;
  color: gray;
  background-color: black;
}
```

버튼을 사용하였습니다.

### 4. Pseudo-element 2 개 이상

```
.article h3::first-letter{
  color: red;
  font-size: 150%;
}

legend::after{
  content : url("h.png");
}
```

#### □ Float 와 Clear 속성

```
.pic {
  height: 30hw;
  width: 30vw;
  display: table;
  margin-left: auto;
  margin-right: auto;
  margin-top: 10px;
  float: bottom;
}
```

#### □ 모바일 환경 대응(media Query , width 900px)

```
@media screen and (max-width: 900px) {
  .nav {
    padding: 30px 40px;
  }

  li>a {
    font-size: 10px;
    color: white;
  }

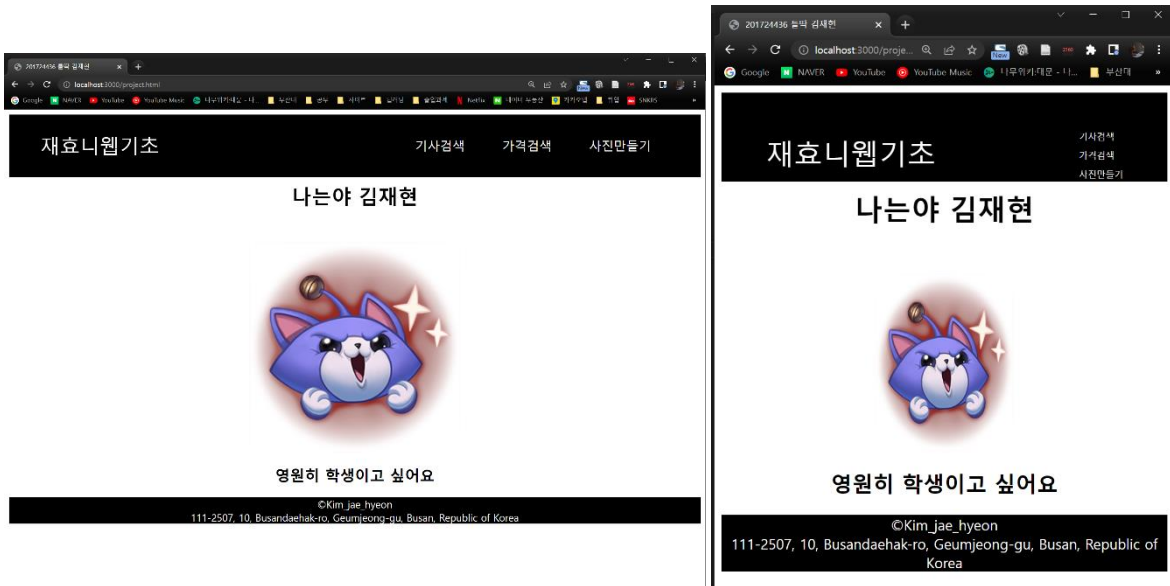
  img {
    height: 200px;
    width: 200px;
  }
}

@media screen and (min-width: 900px) {
  #menu {
    display: flex;
  }

  li>a {
    padding: 20px 30px;
    font-size: 20px;
    color: white;
  }
}
```

다른것들은 거의 vw,vh 를 사용하여 구현하였습니다.

Navigator 는 900 을 기준으로 작아지면 그냥 위에서부터 세로로 나오게 만들었습니다.



### 3 Javascript

□ Dom 을 사용해 HTML Element 나 CSS Property 5 개 이상 바꾸기

```
function setpic() {
    document.getElementById('here').setAttribute('src', "https://apimeme.com/meme?meme=" +
```

```
for (i in storage.items) {
    let temp = document.createElement('div');
    temp.setAttribute('class', 'article');
    let title = document.createElement('h3');
    let discription = document.createElement('p');
    let link = document.createElement('a');
    discription.innerHTML = storage.items[i].description;
    title.innerHTML = storage.items[i].title;
    link.setAttribute('href', storage.items[i].link);
    link.innerHTML = "기사로 바로가기 ><";
    temp.appendChild(title);
    temp.appendChild(discription);
    temp.appendChild(link);
    tagArea.appendChild(temp);
}
```

```
console.log(this.responseText);
let storage = JSON.parse(this.response);
$('#coinresult').empty();
let target = document.getElementById('coinresult');
let temp = document.createElement('h1');
let comment = "어제 마감가격 : " + storage[0].prev_closing_price;
comment = comment + " / 현재가격 : " + storage[0].trade_price;
let temp2 = document.createElement('h3');
temp2.innerHTML = comment;
target.appendChild(temp2);
if (storage[0].prev_closing_price < storage[0].trade_price) {
    temp.innerHTML = "축하해! 올랐습니다!!!!";
} else {
    temp.innerHTML = "적금이나 드세요 ㅋㅋ";
}
target.appendChild(temp);
```

```
let target = document.getElementById('coins');
for (i in storage) {
    let temp = document.createElement('p');
    temp.innerHTML = storage[i].korean_name;
    target.appendChild(temp);
}
```

5 개 이상 무조건 넘습니다

□ 입력 format 오류, 네트워크 연결 오류 등의 예외 처리 기능 구현

```
<input type="text" id="top" value="HELLO" onkeyup="chkCharCode(event)"><br />
<input type="text" id="bottom" value="ITS ME" onkeyup="chkCharCode(event)"><br />
```



```
<input type="text" value="50" id="num" style="width:25px" onkeyup="chkintCode(event)">개씩 검색
```

```
function chkCharCode(event) {  
    const regExp = /^[^0-9a-zA-Z ]/g;  
    const ele = event.target;  
    if (regExp.test(ele.value)) {  
        ele.value = ele.value.replace(regExp, '');  
    }  
};  
function chkintCode(event) {  
    const regExp = /^[^0-9]/g;  
    const ele = event.target;  
    if (regExp.test(ele.value)) {  
        ele.value = ele.value.replace(regExp, '');  
    }  
};
```

한글입력 오류 방지 , 숫자 아닌것들 걸러내기

## □ 추가 기능

코인의 가격이 오르면 축하음악을, 떨어지면 비웃음 소리를 나오게 만들었습니다.

```
function openresult() {  
    var xhr = new XMLHttpRequest();  
    xhr.withCredentials = true;  
    xhr.addEventListener("readystatechange", function () {  
        if (this.readyState === 4) {  
            console.log(this.responseText);  
            let storage = JSON.parse(this.response);  
            $('#coinresult').empty();  
            let target = document.getElementById('coinresult');  
            let temp = document.createElement('h1');  
            let comment = "어제 마감가격 : " + storage[0].prev_closing_price;  
            comment = comment + " / 현재가격 : " + storage[0].trade_price;  
            let temp2 = document.createElement('h3');  
            temp2.innerHTML = comment;  
            target.appendChild(temp2);  
            if (storage[0].prev_closing_price < storage[0].trade_price) {  
                var audio = new Audio('good.MP3');  
                audio.play();  
                temp.innerHTML = "축하해요! 올랐습니다!!!!";  
            }  
            else {  
                var audio = new Audio('bad.MP3');  
                audio.play();  
                temp.innerHTML = "적금이나 드세요 ㅋㅋ";  
            }  
            target.appendChild(temp);  
        }  
    });  
};
```

Good.mp3 가 축하소리 , bad.mp3 가 비웃음소리 입니다.

## 웹사이트 설명

### 서로 다른 사용자 기능 시나리오 3 개 이상 구현

#### □ 뉴스 검색

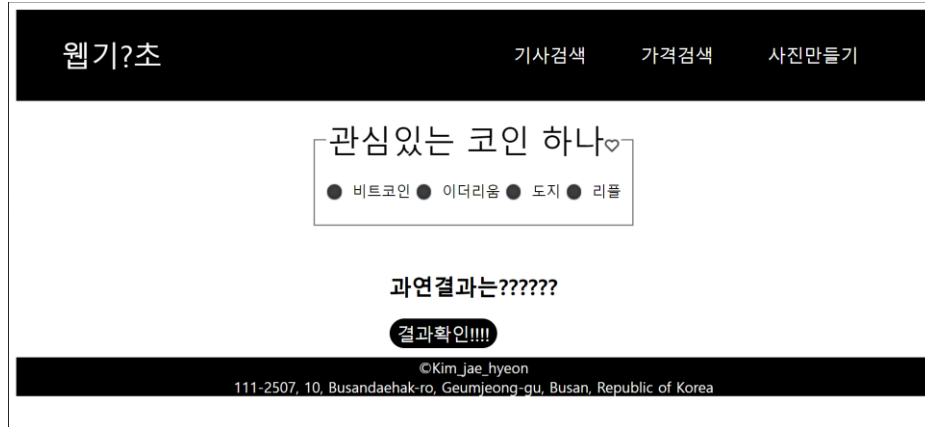
The image displays three wireframe mockups of a news search interface, illustrating different user scenarios:

- Top Left Mockup:** Shows a search bar with the placeholder text "뉴스 검색" and a button labeled "검색 [50]개씩 검색". Below the search bar is a large black rectangular area representing the search results.
- Top Right Mockup:** Shows a search bar with the placeholder text "뉴스 검색" and a button labeled "검색 [50]개씩 검색". Below the search bar is a black rectangular area with the text "뉴스를 보여드립니다!" (We show you news!).
- Bottom Left Mockup:** Shows a search bar with the placeholder text "뉴스 검색" and a button labeled "검색 [50]개씩 검색". Below the search bar is a list of news items, each with a title, a brief description, and a "기사로 바로가기" (Go to article) link. The news items are: "델리오-블루베이, 첫 클라보 NFT '클레이다아노 블루베이 에디션' 발매...", "비트코인 3700만원...겨울, 올까?", "이더리움 긍정-부정 지표 혼재...ETH 방향성 불투명", and "변동 폭 좁은 박스권..시장 지켜볼 필요..비트코인,이더리움 횡보 속 웨이브 ...".
- Bottom Right Mockup:** Shows a search bar with the placeholder text "뉴스 검색" and a button labeled "검색 [2]개씩 검색". Below the search bar is a list of news items, each with a title, a brief description, and a "기사로 바로가기" (Go to article) link. The news items are: "델리오-블루베이, 첫 클라보 NFT '클레이다아노 블루베이 에디션' 발매...", "비트코인 3700만원...겨울, 올까?", and "이더리움 긍정-부정 지표 혼재...ETH 방향성 불투명".

뉴스 검색을 해서 뉴스를 얻을 수 있습니다. 검색어를 입력 후 검색을 누르면 뉴스가 찾아지고 오른쪽 개수도 정해서 찾을 수 있습니다. 오른쪽 숫자입력칸에 다른 문자가 들어오면 자동으로 삭제됩니다.

검색은 아무거나 입력 가능합니다.

#### □ 가격 확인

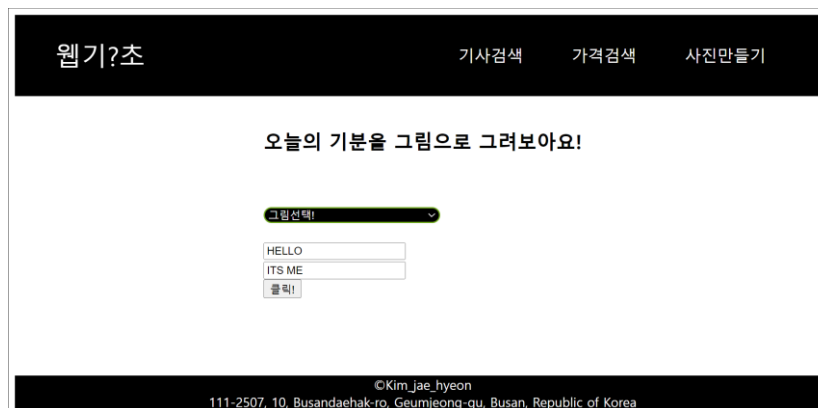


비트코인,이더리움,도지,리플 중 하나의 가격을 확인할 수 있습니다.



확인을 하고 가격이 올라갔다, 내려갔다는 정보를 알려주고 음악이 재생됩니다. 라디오박스 조금 이쁘게 했습니다

## □ 이미지 생성



사진,Top 넣을 문구, bottom 넣을 문구를 사용해서 이미지 생성이 가능합니다.

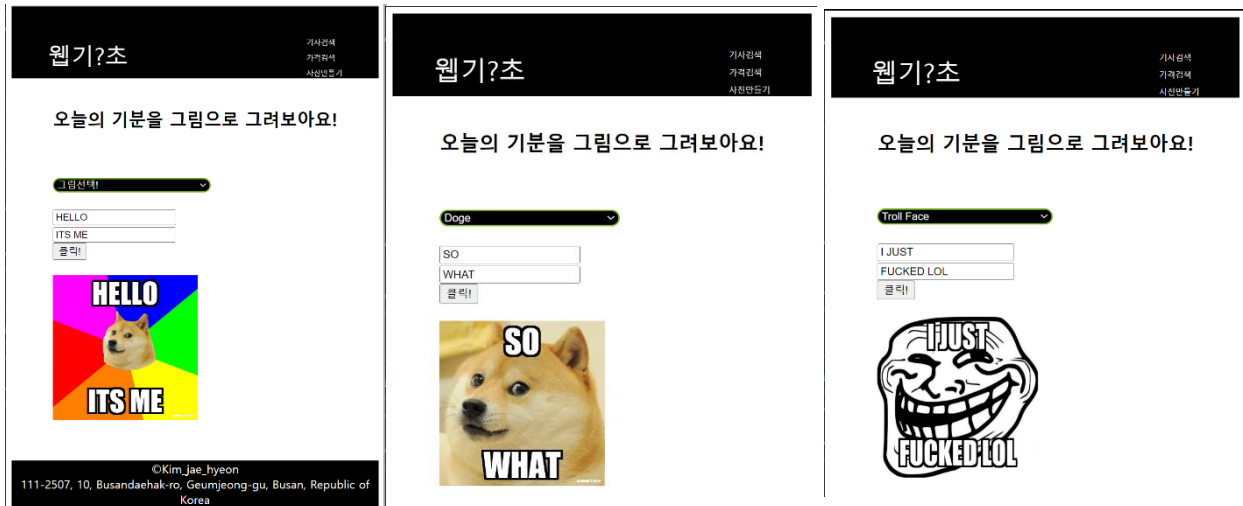
기본 선택은 Advice Dorge 로 되어 있습니다.

오늘의 기분을 그림으로 그려보아요!



©Kim jae\_hyeon  
111-2507, 10, Busandaehak-ro, Geumjeong-gu, Busan, Republic of Korea

해당 이미지는 view width 가 900 이하가 되면 작아집니다.



여러가지 사진이 많으니 골라서 사용하시면 되나, 이 사이트가 영어만 지원하여 입력에 한글이 들어오지 못하도록 막아두었습니다.

## 코드 간략설명

### 1 JS

Index.js 는 서버부분입니다. Data 를 forwarding 하는 역할만 구현하였습니다.

```
app.get('/news/:target/:num', function (req, res) {
  var request = require('request');
  var options = {
    'url': encodeURI('https://openapi.naver.com/v1/search/news.json?query='
      + req.params.target + '&display=' + req.params.num),
    'headers': {
      'X-Naver-Client-Id': 'aJ3xkRZngumFQHAXp8zF',
      'X-Naver-Client-Secret': 'f7K_TLW4w1'
    }
  };
  request(options, function (error, response) {
    if (error) throw new Error(error);
    res.send(response.body);
  });
});
```

뉴스정보 forwarding 입니다. 인증이 필요합니다.encode 도 필요했습니다.

```
app.get('/coinmenu', function (req, res) {
  var request = require('request');
  var options = {
    'method': 'GET',
    'url': 'https://api.upbit.com/v1/market/all',
    'headers': {
    }
  };
  request(options, function (error, response) {
    if (error) throw new Error(error);
    console.log(response.body);
    res.send(response.body);
  });
});
```

```
app.get('/coinprice/:what', function (req, res) {
  var request = require('request');
  var options = {
    'method': 'GET',
    'url': 'https://api.upbit.com/v1/ticker?markets=' + req.params.what,
    'headers': {
    }
  };
  request(options, function (error, response) {
    if (error) throw new Error(error);
    console.log(response.body);
    res.send(response.body);
  });
});
```

코인정보와 코인 가격 forwarding 입니다.

Project.js 는 html 의 script 입니다.

```
function displayNEWS() {
  console.log(document.getElementById("searching").innerHTML);
  var xhr = new XMLHttpRequest();
  xhr.withCredentials = true;
  xhr.addEventListener("readystatechange", function () {
    if (this.readyState === 4) {
      console.log(this.responseText);
      let storage = JSON.parse(this.response);
      $('#news').empty();
      let tagArea = document.getElementById("news");
      for (i in storage.items) {
        let temp = document.createElement('div');
        temp.setAttribute('class', 'article');
        let title = document.createElement('h3');
        let discription = document.createElement('p');
        discription.setAttribute('title', 'des');
        let link = document.createElement('a');
        discription.innerHTML = storage.items[i].description;
        title.innerHTML = storage.items[i].title;
        link.setAttribute('href', storage.items[i].link);
        link.innerHTML = "기사로 바로가기 ><";
        temp.appendChild(title);
        temp.appendChild(discription);
        temp.appendChild(link);
        tagArea.appendChild(temp);
      }
    }
  });
  url = "http://localhost:3000/news/" + document.getElementById('searching').value + "/" + document.getElementById('num').value;
  xhr.open("GET", url);
  xhr.send();
}
```

displayNews()는 데이터를 받아와서 파싱하고 뉴스영역으로 남겨둔 div 에 뉴스요소를 추가해주는 작업을 합니다. createElement 로 div 를 만들어서 하나씩 추가하는식으로 구성했습니다. 처음에 뉴스 div 를 clear 하고 시작합니다.

```
function openresult() {
  var xhr = new XMLHttpRequest();
  xhr.withCredentials = true;
  xhr.addEventListener("readystatechange", function () {
    if (this.readyState === 4) {
      console.log(this.responseText);
      let storage = JSON.parse(this.response);
      $('#coinresult').empty();
      let target = document.getElementById('coinresult');
      let temp = document.createElement('h1');
      let comment = "어제 마감가격 : " + storage[0].prev_closing_price;
      comment = comment + " / 현재가격 : " + storage[0].trade_price;
      let temp2 = document.createElement('h3');
      temp2.innerHTML = comment;
      target.appendChild(temp2);
      if (storage[0].prev_closing_price < storage[0].trade_price) {
        var audio = new Audio('good.MP3');
        audio.play();
        temp.innerHTML = "축하해요! 올랐습니다!!!!";
      }
      else {
        var audio = new Audio('bad.MP3');
        audio.play();
        temp.innerHTML = "적금이나 드세요 ㅋㅋ";
      }
      target.appendChild(temp);
    }
  });
  var chose;
  coinlist = document.getElementsByName('coin');
  coinlist.forEach(node => {
    if (node.checked) {
      chose = node.value;
    }
  });
  xhr.open("GET", "http://localhost:3000/coinprice/" + chose);
  xhr.send();
}
```

코인의 가격정보를 가져오는 함수입니다. Json 을 받아와서 storage 에 저장 후 어제가격과 현재가격을 비교해서 넣어줍니다. 넣어줄 때 가격이

올랐냐 내렸냐에 따라서 다르게 보여주고 음악도 다르게 재생됩니다.

```
function setpic() {  
    document.getElementById('here').setAttribute('src', "https://apimeme.com/meme?meme=" +  
        document.getElementById('meme').value + "&top=" +  
        document.getElementById('top').value + "&bottom=" +  
        document.getElementById('bottom').value);  
}
```

사진정보를 가져오는 함수입니다. API 호출이 따로 없고 링크로 가져옵니다.

```
function chkCharCode(event) {  
    const regExp = /^[^0-9a-zA-Z ]/g;  
    const ele = event.target;  
    if (regExp.test(ele.value)) {  
        ele.value = ele.value.replace(regExp, '');  
    }  
};  
function chkintCode(event) {  
    const regExp = /^[^0-9]/g;  
    const ele = event.target;  
    if (regExp.test(ele.value)) {  
        ele.value = ele.value.replace(regExp, '');  
    }  
};
```

입력 제한 부분입니다. Chkcharcode 는 한글입력을 못하게 막고  
chkintcode 는 숫자만 입력받도록 합니다.

## 2 Html

html 파일은 4 개가 있습니다.

Project.html-기본 자기소개 파일

Project\_news.html-뉴스 페이지

Project\_price.html-코인 가격 확인 페이지

Project\_pic.html-그림 생성 페이지

위 4 개 파일은 nav 에 걸린 링크로 열 수 있습니다.

감사합니다.