2019.02.14

9회차

# 터치10인치 연결 및 가상화키보드 연결

# 기본 모듈공부

## Notification으로 출력 제어하기

notificationReceived: function(notification, payload, sender) {

switch(notification){

case "DOM\_OBJECTS\_CREATED":

var timer = setInterval(()=>{

var countElm = document.getElementById("COUNT")

countElm.innerHTML= "Count:" + this.count

this.count++

/\*

this.updateDom()

this.count++

\*/

}, 1000)

break

}

},

getElementById를 통해 카운트를 증가시킨다. updatedom의 대신역할이다.

getDom: function() {

var element = document.createElement("div")

element.className = "myContent"

element.innerHTML = "Hello, World! " + this.config.foo

var subElement = document.createElement("p")

subElement.innerHTML = "Count:" + this.count

subElement.id = "COUNT"

element.appendChild(subElement)

return element

},

notificationReceived: function(notification, payload, sender) {

switch(notification){

case "DOM\_OBJECTS\_CREATED":

var timer = setInterval(()=>{

this.updateDom()

this.count++

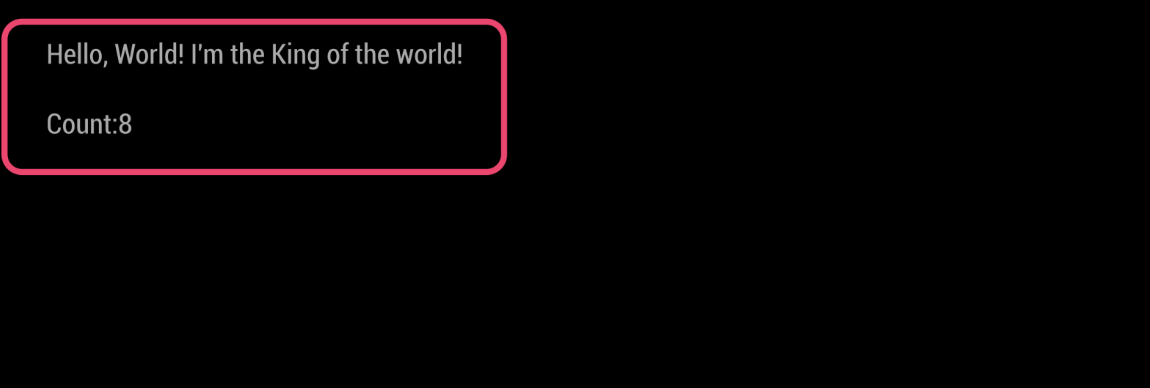
}, 1000)

break

}

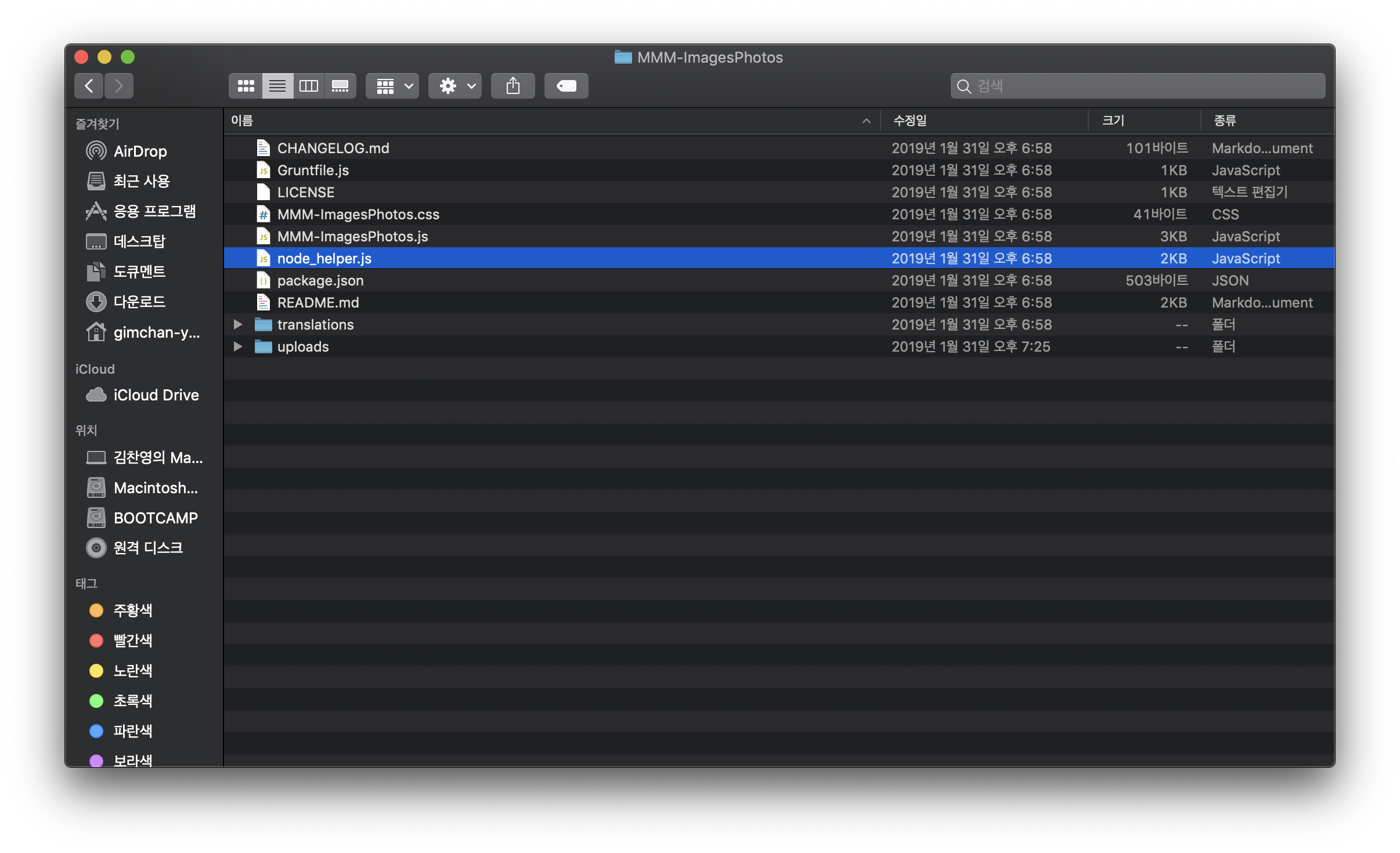
},

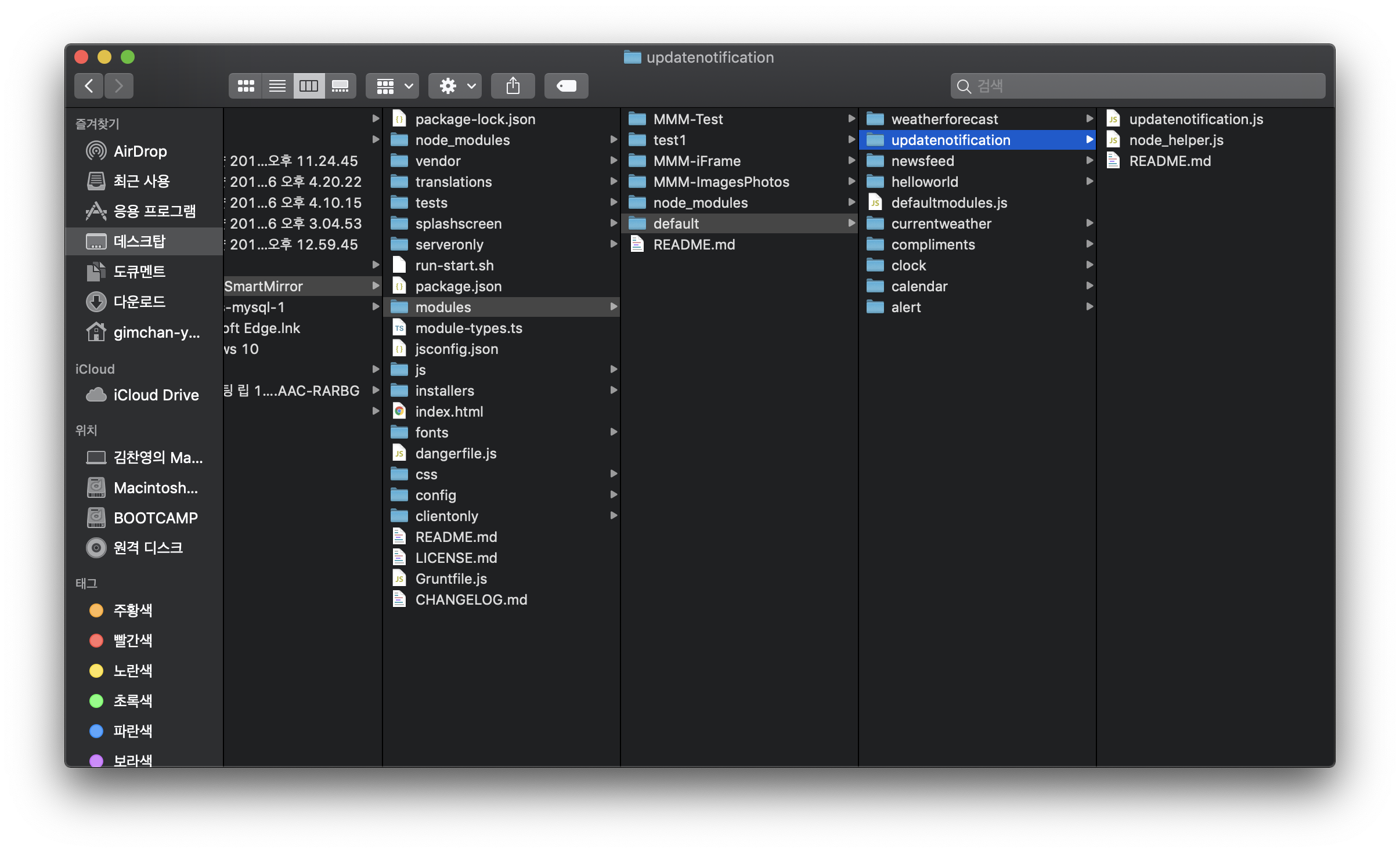
그 전함수와 비교하면 출력은 같지만 원리는 다르다.



출력된값

## NODE\_HELPERS.js





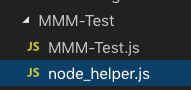
이런식으로 모듈에는 항상 node\_helper가 있는데 이건무엇일까?

node\_help란 무엇인가

모듈.js는 제한이 있다. 브라우저(electron)나 mm프레임워크위에서

실행하기때문에 브라우저에서의 기능보다 다른 기능이 필요할때 nodehelp를 쓴다

예를들어 로컬파일을 읽거나 쓰는것이런것은 브라우져로 하지못하는데 nodehelp로는 가능하다.

node helper.js를 만들어준다.

var NodeHelper = require("node\_helper")

module.exports = NodeHelper.create({

start: function() {

this.countDown = 10000000

},

socketNotificationReceived: function(notification, payload) {

switch(notification) {

case "DO\_YOUR\_JOB":

this.sendSocketNotification("I\_DID", (this.countDown - payload))

break

}

},

})

다음과 같이 추가

nodehelper에서의 start는 모듈.js와 같이 모듈이 시작될때 시작하는 부분이다.

socketNotificationReceived는 모듈.js와 nodehelp를 통신할때 쓰는 메소드이다.

(예를 들면 notification과 같다)

여기서 socketNotification을 받았을때 즉 socketNotificationReceived

경우일때 sendSocketNotification함수로 보내준다.

자여기서 모듈.js를 다음과 같이 수정한다.

Module.register("MMM-Test", {

defaults: {

foo: "I'm alive!"

},

start: function () {

this.count = 0

},

getDom: function() {

var element = document.createElement("div")

element.className = "myContent"

element.innerHTML = "Hello, World! " + this.config.foo

var subElement = document.createElement("p")

subElement.id = "COUNT"

element.appendChild(subElement)

return element

},

notificationReceived: function(notification, payload, sender) {

switch(notification) {

case "DOM\_OBJECTS\_CREATED":

var timer = setInterval(()=>{

this.sendSocketNotification("DO\_YOUR\_JOB", this.count)

this.count++

}, 1000)

break

}

},

socketNotificationReceived: function(notification, payload) {

switch(notification) {

case "I\_DID":

var elem = document.getElementById("COUNT")

elem.innerHTML = "Count:" + payload

break

}

},

})

