#1.

1.hello. 2.my name is eom. 3.nice to meet you. 4.my major is cse. 5.day 7 assignment. 6.success. 7.finish.

```
eom@eom-900X5T:~/nodejs$ ares-package sampleapp sampleservice
Create com.domain.app_1.0.0_all.ipk to /home/eom/nodejs
Success
eom@eom-900X5T:~/nodejs$ ares-install -d emulator com.domain.app_1.0.0_all.ipk
Installing package com.domain.app_1.0.0_all.ipk
Success
eom@eom-900X5T:~/nodejs$ ares-launch --device emulator com.domain.app
Launched application com.domain.app on display 0
eom@eom-900X5T:~/nodejs$ ares-inspect --device emulator --app com.domain.app
Application Debugging - http://localhost:38483/devtools/inspector.html?ws=localhost:38483/devtools/page/5A8E2FF573D2096B7070E0785C0A1E9E
```

<sampleapp/index.html code>
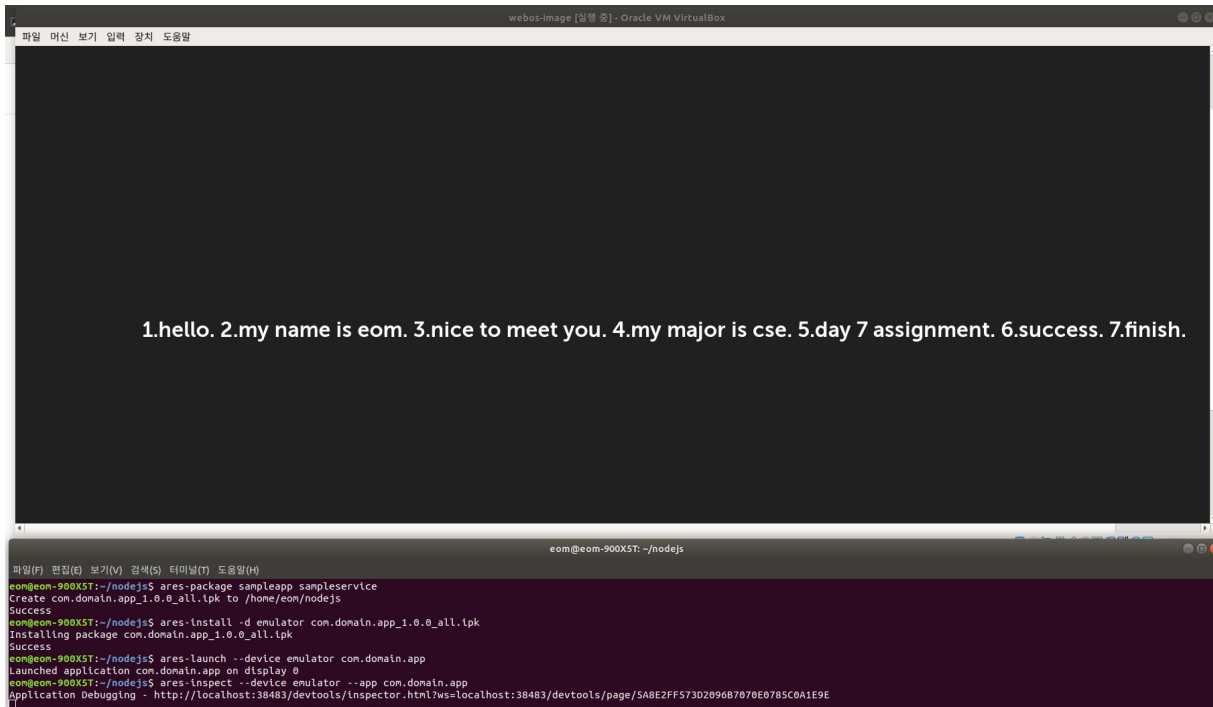
```
30    <script type="text/javascript">
31    window.onload = function() {
32        var bridge = new WebOSServiceBridge();
33        /*
34         *  getTimeApi calls gettime of systemservice, a module in the platform.
35         */
36        var getTimeApi = 'luna://com.webos.service.systemservice/clock/getTime';
37        var getTimeParams = '{}';
38
39        /*
40         *  helloApi calls the hello method of js_service template provided by CLI.
41         *  In this case, the service name is used as default name "com.domain.app.service" is.
42         *  If you change this service name, you need to change the service name of the following API.
43         *
44         *  If you change the name to helloparmas as you want, the contents will be reflected on the sc
45         */
46        var helloApi = 'luna://com.domain.app.service/hello';
47        var helloParams = '{"name":"webOS"};
48
49
50
51        var txtApi = 'luna://com.domain.app.service/listfiles';
52        var txtParams = '{"path":"./test.txt"}';
53
54
55        function txt_callback(msg){
56            var arg = JSON.parse(msg);
57            if (arg.returnValue) {
58                document.getElementById("txt_msg").innerHTML = arg.Response;
59                console.log("[APP_NAME: example web app] CALLHELLO_SUCCESS response : " + arg.Response);
60            }
61            else {
62                console.error("[APP_NAME: example web app] CALLHELLO_FAILED errorText : " + arg.errorText);
63            }
64        }
65
66        bridge.onservicecallback = txt_callback;
67        bridge.call('luna://com.domain.app.service/listfiles', '{"path":"./test.txt"}');
68
69    }
70    </script>
71    </head>
72    <body>
73        <div>
74            <h1 id="txt_msg">Hello, Web Application!!</h1>
75
76        </div>
77    </body>
78    </html>
79
```
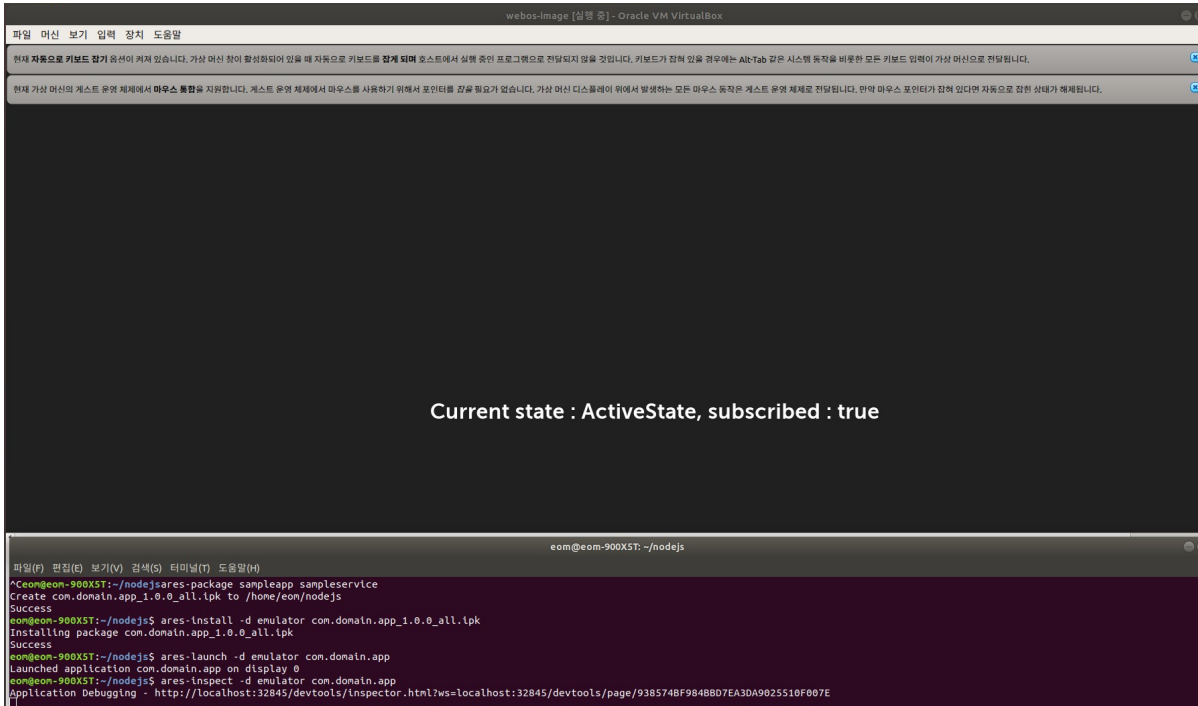
<sampleservice/helloworld_webos_service.js code>

```js
177
178   service.register("pong", function(message) {
179       console.log(logHeader, "SERVICE_METHOD_CALLED:/pong");
180       console.log("Pong!");
181       console.log(message.payload);
182       message.respond({message: "Pong"});
183   });
184
185   service.register("/do/re/me", function(message) {
186       console.log(logHeader, "SERVICE_METHOD_CALLED://do/re/me");
187       message.respond({verses:[
188           {doe: "a deer, a female deer"},
189           {ray: "a drop of golden sun"},
190           {me: "a name I call myself"}
191       ]});
192   });
193
194   service.register("listfiles",function(message){
195       /*var path = ROOT+message.payload.path;
196       fs.readdir(path,function(err,files){
197           if(err){
198               message.respond({
199                   returnValue: false,
200                   errorCode:"Error when reading files"
201               });
202           }
203           else{
204               message.respond({
205                   returnValue:true,
206                   files:files
207               });
208           }
209       });*/
210
211       var path = ROOT+message.payload.path;
212       fs.readFile(path,'utf8',function(err,data){
213           console.log(data);
214           message.respond({
215               message : data,
216               returnValue:true,
217               Response:  data
218           });
219       });
220   });
221
```

<terminal>

```
root@qemux86:~# luna-send -n 1 luna://com.domain.app.service/listfiles '{"path":"
./test.txt"}'
{"message":"1.hello.\n2.my name is eom.\n3.nice to meet you.\n4.my major is cse.\
n5.day 7 assignment.\n6.success.\n7.finish.\n","returnValue":true,"Response":"1.h
ello.\n2.my name is eom.\n3.nice to meet you.\n4.my major is cse.\n5.day 7 assign
ment.\n6.success.\n7.finish.\n"}
root@qemux86:~#
```
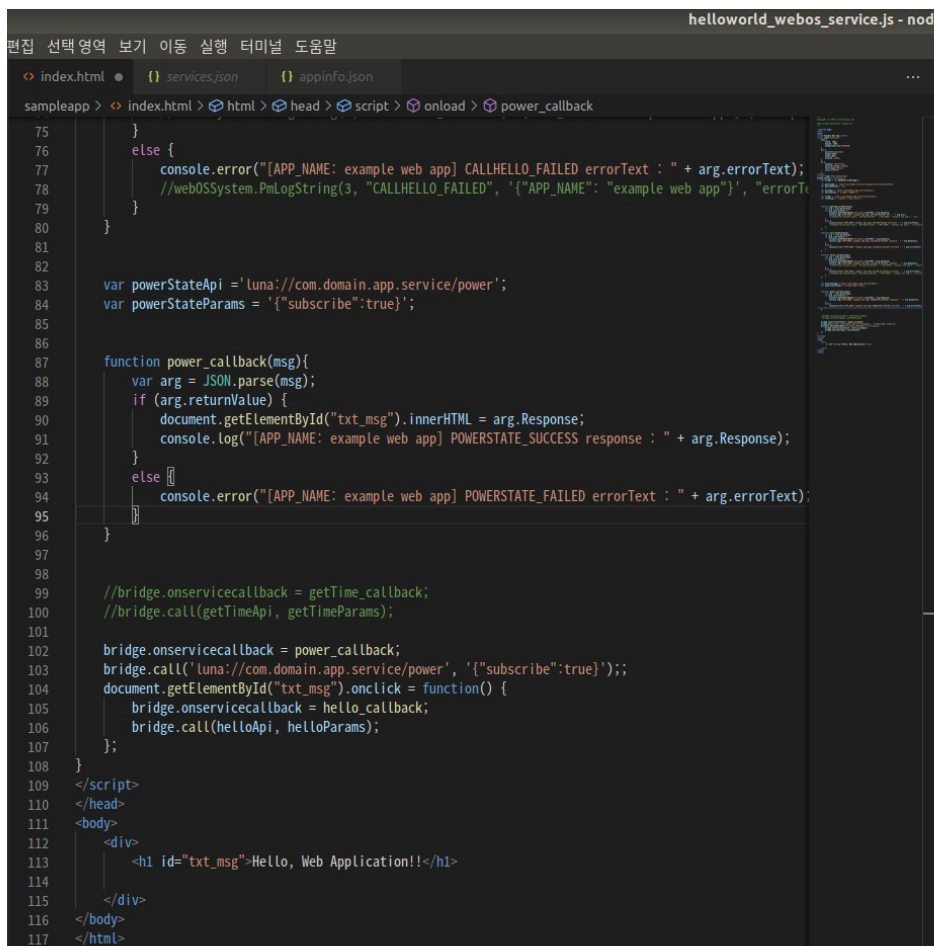
#2.
I used luna-send -n i luna://com.webos.service.power2/getState which was taught in class.

<Index.html>


```
 75              }
 76              else {
 77                  console.error("[APP_NAME: example web app] CALLHELLO_FAILED errorText : " + arg.errorText);
 78                  //webOSSystem.PmLogString(3, "CALLHELLO_FAILED", '{"APP_NAME": "example web app"}', "errorTe
 79              }
 80          }
 81
 82
 83          var powerStateApi ='luna://com.domain.app.service/power';
 84          var powerStateParams = '{"subscribe":true}';
 85
 86
 87          function power_callback(msg){
 88              var arg = JSON.parse(msg);
 89              if (arg.returnValue) {
 90                  document.getElementById("txt_msg").innerHTML = arg.Response;
 91                  console.log("[APP_NAME: example web app] POWERSTATE_SUCCESS response : " + arg.Response);
 92              }
 93              else {
 94                  console.error("[APP_NAME: example web app] POWERSTATE_FAILED errorText : " + arg.errorText);
 95              }
 96          }
 97
 98
 99          //bridge.onservicecallback = getTime_callback;
100          //bridge.call(getTimeApi, getTimeParams);
101
102          bridge.onservicecallback = power_callback;
103          bridge.call('luna://com.domain.app.service/power', '{"subscribe":true}');;
104          document.getElementById("txt_msg").onclick = function() {
105              bridge.onservicecallback = hello_callback;
106              bridge.call(helloApi, helloParams);
107          };
108      }
109      </script>
110      </head>
111      <body>
112          <div>
113              <h1 id="txt_msg">Hello, Web Application!!</h1>
114
115          </div>
116      </body>
117      </html>
```

<appinfo.json>

```json
{
  "id": "com.domain.app",
  "version": "1.0.0",
  "vendor": "My Company",
  "type": "web",
  "main": "index.html",
  "title": "new app",
  "icon": "icon.png",
  "requiredPermissions": [
    "time.query",
    "activity.operation",
    "power.operation",
    "power.management"
  ]
}
```

<helloworld_webos_service.js>

```js
        const name = message.payload.name ? message.payload.name : 'World';

        message.respond({
            returnValue: true,
            Response: "Hello, " + name + "!"
        });
});




service.register("power", function(message) {
    console.log(logHeader, "SERVICE_METHOD_CALLED:/power");
    console.log("power callback");
    var subscribe = message.payload.subscribe;

    service.call("luna://com.webos.service.power2/getState", {"subscribe":subscribe}, function(m2) {
        console.log(logHeader, "SERVICE_METHOD_CALLED:luna://com.webos.service.power2/getState");
        const response = m2.payload.state;
        const response2 = m2.payload.subscribed;

        console.log(response);

        message.respond({
            returnValue :true,
            Response: "Current state : " + response + ", subscribed : " + response2
        });
    });
});
```

<terminal>

```
root@qemux86:~# luna-send -n 1 luna://com.domain.app.service/power '{"subscribe":true}'
{"returnValue":true,"Response":"Current state : ActiveState, subscribed : true"}
root@qemux86:~#
```