

---

\*\*\*\*\* Advanced Arduino 기말고사. 2017. 12. 13(수) \*\*\*\*\*

Arduino + Node.js + Plot.ly + HTML5

---

1. What is the command to set up a node project by installing node modules defined in package.json?

- A. npm start
- B. **npm install**
- C. npm init
- D. npm setup

2. How can you install a node module socket.io of the specific version 1.7.3?  
The version of installed module will be definitely written in package.json.

- A. npm init —save socket.io@1.7.3
- B. npm init —save socket.io#1.7.3
- C. **npm install —save socket.io@1.7.3**
- D. npm install —save socket.io#1.7.3

3. Now, you have a resistor with four color bands;  
Brown(갈), Black, Red, Silver.

What is the resistance value of this resistor?

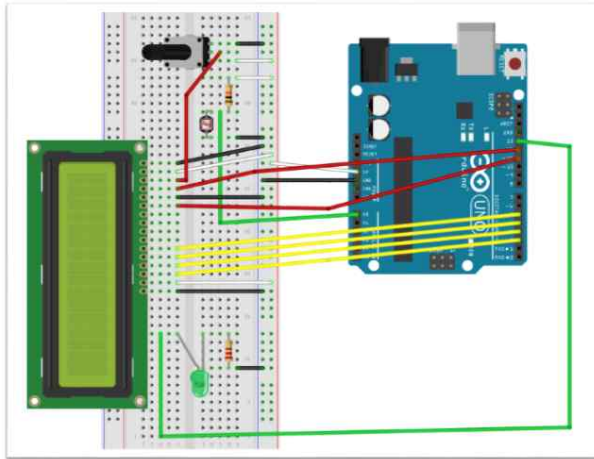
- A. 330 Ω
- B. **1 kΩ**
- C. 10 kΩ
- D. 20 kΩ

4. What is the incorrect introduction to plotly.js?

- A. plotly.js is a JavaScript library.
- B. plotly.js is related with jQuery.
- C. plotly.js is built on top of d3.js.
- D. **plotly.js is a text-based library.**

5-9. 다음은 CdS 센서 측정값을 lux로 구하여 LCD에 표시하고, 조도가 문턱값(90)을 넘는 경우 D13에 위치한 LED를 끄고, 반대인 경우 LED를 켜는 아두이노 코드이다. 밑줄친 곳에 알맞은 코드는?

<pre>#include &lt;LiquidCrystal.h&gt; // LCD 라리브러리 설정 LiquidCrystal lcd(12, 11, 5, 4, 3, 2); // LCD pin 설정  // 0번 아날로그핀을 CdS 셀 입력으로 설정한다. const int CdSPin = 0; const int ledPin = 13; // LED pin 설정  int threshold = 90; // LED OFF above threshold  void setup() {   pinMode(ledPin, OUTPUT);    // 16X2 LCD 모듈 설정하고 백라이트를 켜다.   lcd[5]_____ ;    // 메시지를 표시한다.   lcd.print("HPnn");   lcd.setCursor(0,1);   lcd.print("CdS Cell Test");    // 3초동안 메시지를 표시한다.   delay(3000);    // 모든 메시지를 삭제한 뒤   // 숫자를 제외한 부분들을 미리 출력시킨다.   lcd[6]_____ ; // LCD 전체를 지운다.   lcd.setCursor(0,0);   lcd.print("AAnn: ");   lcd.setCursor(0, [7]_____); // LCD의 아래 칸으로   lcd.print("Light: ");   lcd.setCursor(13,1);   lcd.print("lux"); }</pre>	<pre>void loop(){   int adcValue; // 실제 센서로부터 읽은 값 (0~1023)   int illuminance; // 현재의 밝기. 0~100%   int lux;        // 현재의 밝기. lux    // CdS cell을 통하여 입력되는 전압을 읽는다.   adcValue = analogRead([8]_____);    // luminosity() 함수를 이용해서 Lux 를 계산한다.   lux = int(luminosity(adcValue));    // 전에 표시했던 내용을 지우고   // LCD에 ADC 값과 밝기를 출력한다.   // 지우지 않으면 이전에 표시했던 값이 남게 된다.   lcd.setCursor(12,0);   lcd.print(" ");   // ADC 값을 표시한다   lcd.setCursor(12,0);   lcd.print(adcValue);   // 전에 표시했던 내용을 지운다.   lcd.setCursor(9,1);   lcd.print(" ");   // 밝기를 표시한다   lcd.setCursor(9,1);   lcd.print(lux);    if(lux [9]_____ threshold)     digitalWrite(ledPin, LOW);   else     digitalWrite(ledPin, HIGH);    delay(1000); }  //Voltage to Lux double luminosity (int RawADC0){   double Vout=RawADC0*0.0048828125;   // 5/1024 (Vin = 5 V)   int lux=(2500/Vout-500)/10; // lux = 500 / Rldr,   // Vout = Ildr*Rldr = (5/(10 + Rldr))*Rldr   return lux; }</pre>
--	---



5. 16X2 LCD 모듈 설정하고 백라이트를 켜는 함수는?

- A. start(16,2)                      B. light(16,2)
- C. begin(16,2)                      D. init(16,2)

6. LCD 화면 전체를 지우는 함수는?

- A. clear()                              B. clean()
- C. wipe()                                D. delete()

7. LCD의 아래 칸으로 커서를 이동할 때 적절한 숫자는? --- (      1      )

8. CdS cell을 통하여 입력되는 전압을 읽기위한 변수는? --- (      CdSPin      )

9. 조도가 문턱값(90)을 넘는 경우 LED를 끄고, 반대인 경우 LED를 켜기 위한 관계식을 완성하시오.

--- (      > or >=      )

10. 다음은 timestamp에서 무작위수를 하나씩 만들어 화면에 30개의 데이터를 갱신하는 데이터 스트리밍 html 문서이다.

```
<body>

  <h1>Data visualization by AAnn</h1>
  <hr>
  <h2>Streaming using 30 points update with timestamp</h2>
  <div id="graph"></div>

  <script>
    var arrayLength = 30
    var newArray = []
    var timeArray = []

    // initial 30 data
    for(var i = 0; i < arrayLength; i++) {
      var y = Math.round(Math.random()*10) + 1
      var time = new Date();
      newArray[i] = y
      timeArray[i] = time
    }

    var data = [{
      x: timeArray,
      y: newArray,
      mode: 'lines+ markers',
      line: {color: '#80CAF6'},
      marker: {color: '#FC1234'}
    }];

    Plotly.plot('graph', data);
```

```

var cnt = 0;
var interval = setInterval(function() {
    var y = Math.round(Math.random()*10) + 1
    var time = new Date();
    timeArray = timeArray.concat(time)
    timeArray.splice(0, 1) // remove the oldest data
    newArray = newArray.concat(y)
    newArray.splice(0, 1) // remove the oldest data

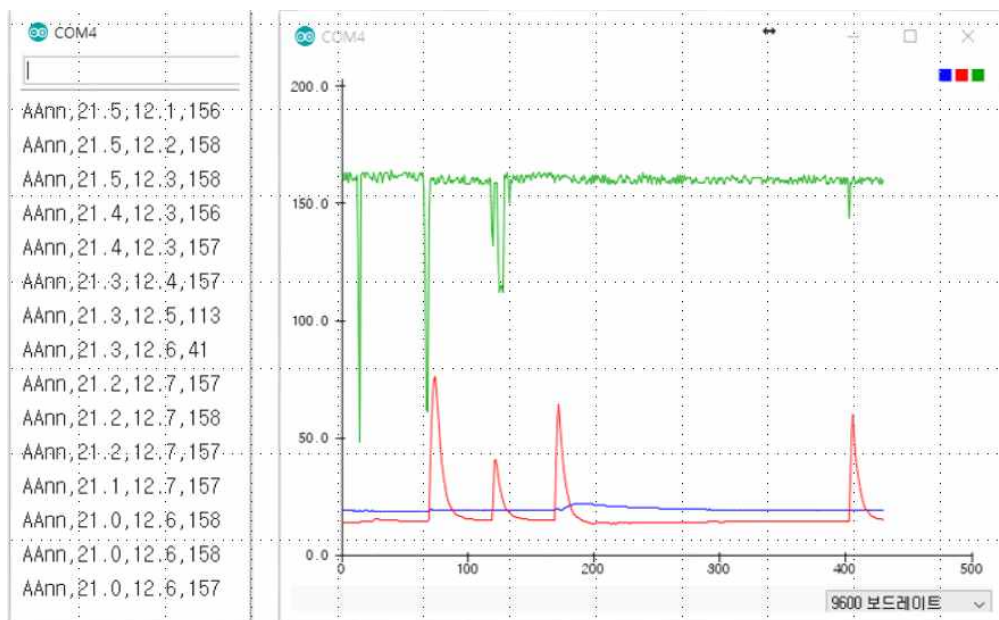
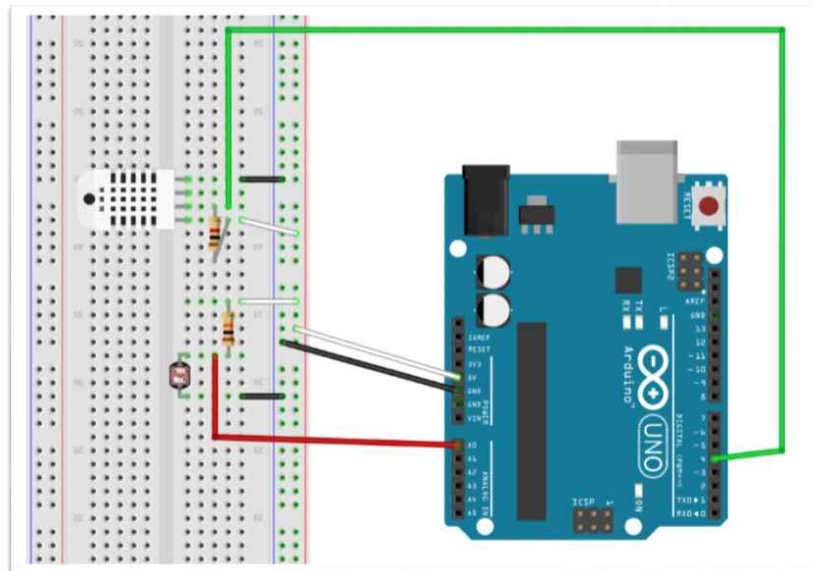
    var update = {
        x: [timeArray],
        y: [newArray]
    }
    Plotly.update('graph', update)
    cnt++;
    if(cnt === 100) [10]_____clearInterval(interval);
}, 200);
</script>
</body>

```

10. 100번째 무작위숫자가 만들어져 그래프에 나타나면 스트리밍을 중단하는 함수는?

- |                |                  |
|----------------|------------------|
| A. endInterval | B. startInterval |
| C. setInterval | D. clearInterval |

[11-20]. 다음은 DHT22 온습도센서와 CdS로 구성된 아두이노 회로로부터 나오는 세 개의 신호를 Node.js와 Plot.ly를 이용해서 실시간 스트리밍하는 과정이다.  
DHT22는 1 K $\Omega$  저항, CdS는 10 K $\Omega$  저항과 연결되어 있다.  
온습도 값은 D4에서, 조도는 A0에서 전송된다.



11-12. [Arduino] 다음은 Arduino 코드이며, 빈 곳에 적합한 코드를 찾으시오.

```
// DHT22_CdS.ino
#include "DHT.h"
#define DHTPIN 4
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
#define CDS_INPUT 0
void setup() {
  dht.[11]_____begin(); // DHT22 초기화
  Serial.begin(9600);
}
void loop() {
  int cds_value, lux;
  float temp, humi;
  // Lux from CdS (LDR)
  cds_value = analogRead(CDS_INPUT);
  lux = int(luminosity(cds_value));
  // Reading temperature or humidity takes a given interval!
  // Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)
  humi = dht.readHumidity();
  // Read temperature as Celsius (the default)
  temp = dht.readTemperature();
  // Check if any reads failed and exit early (to try again).
  if ([12]_____isnan(humi) || [12]_____isnan(temp) || [12]_____isnan(lux)) {
    Serial.println("Failed to read from DHT sensor or CdS!");
    return;
  }
  else {
    Serial.print("AAnn,");
    Serial.print(temp,1); // temperature
    Serial.print(",");
    Serial.print(humi,1); // humidity
    Serial.print(",");
    Serial.println(lux); // luminosity
  }
  delay(1000); // 1000 msec,
}
//Voltage to LuxLux
double luminosity (int RawADC0){
  double Vout=RawADC0*0.0048828125; // 5/1024 (Vin = 5 V)
  int lux=(2500/Vout-500)/10;
  // lux = 500 / Rldr, Vout = Ildr*Rldr = (5/(10 + Rldr))*Rldr
  return lux;
}
```



11. DHT22 센서를 초기화하는 함수는?

- A. start()
- B. init()
- C. begin()
- D. call()

12. 센서에서 나온 데이터들의 적합성을 확인하는 함수는?

- A. isnan
- B. isvalid
- C. istrue
- D. isnumber

13-15. [Node.js] 다음은 node.js 코드이며, 빈 곳에 적합한 코드를 찾으시오.

```
// dht22_ldr_node.js
var serialport = require('serialport');
var portName = 'COM4'; // check your COM port!!
var port = process.env.PORT || 3000;

var io = require('socket.io').listen(port);

// serial port object
var sp = new serialport(portName,{
  baudRate: 9600, // 9600 38400
  dataBits: 8,
  parity: 'none',
  stopBits: 1,
  flowControl: false,
  parser: serialport.parsers.readline('\r\n')
});

var readData = ''; // this stores the buffer
var temp = '';
var humi = '';
var lux = '';
var mdata = []; // this array stores date and data from multiple sensors
var firstcommaidx = 0;
```

```

sp.on('data', function (data) { // call back when data is received
  readData = data.toString(); // append data to buffer
  firstcommaidx = readData.indexOf(',');

  // parsing data into signals
  if (readData.lastIndexOf(',') > firstcommaidx && firstcommaidx > 0) {
    temp = readData.substring(firstcommaidx + 1,
                                readData.indexOf(', '
[13]_____firstcommaidx+ 1));
    humi      = readData.substring(readData.indexOf(', '
[13]_____firstcommaidx+ 1) + 1,
                                readData.lastIndexOf(', '));
    lux = readData.substring([14]_____readData.lastIndexOf(',')+ 1);
    readData = "";

    dStr = getDateString();
    mdata[0]=dStr; // Date
    mdata[1]=temp; // temperature data
    mdata[2]=humi; // humidity data
    mdata[3]=lux;  // luminosity data
    console.log(mdata);
    io.sockets.emit([15]_____ 'message', mdata); // send data to all
clients
    } else { // error
      console.log(readData);
    }
  });
});

```

13. ','로 구분되는 온도, 습도 값을 문자열로 구하는 방법은?

- A. firstcommaidx                      B. firstcommaidx+ 1  
 C. firstcommaidx+ 2                    D. firstcommaidx+ 3

14. 조도(lux) 값을 문자열로 구하는 방법은?

- A. readData.lastIndexOf(',')                      B. readData.lastIndexOf(',')+ 1  
 C. readData.lastIndexOf(',')+ 2                    D. readData.lastIndexOf(',')+ 3

15. mdata 배열에 저장된 (시간, 온도, 습도, 조도)를 소켓을 통해 모든 클라이언트에게 전달할 때 사용되는 키값은?

- A. 'connect'                      B. 'data'  
 C. 'send'                      D. 'message'

16-20. [HTML Client] 다음은 dht22\_ldr.html 파일이며, 세 개의 페이지와 세 개의 그래프로 [온도, 습도, 조도] 데이터를 시각화한다.

```
<!DOCTYPE html>
<head>
  <meta charset="utf-8">
  <title>plotly.js Project: Real time signals from multiple sensors</title>
  <script src="https://cdn.plot.ly/plotly-latest.min.js"></script>
  <script type="text/javascript"
src="https://cdnjs.cloudflare.com/ajax/libs/socket.io/1.3.6/socket.io.js"></script>
  <script src="gauge.min.js"></script>
  <style>body{padding:0;margin:30;background:#fff}</style>
</head>

<body> <!-- style="width:100%;height:100%"> -->
  <!-- Plotly chart will be drawn inside this DIV -->
  <h1 align="center"> Real-time Weather Station from sensors </h1>
  <!-- 1st gauge -->
  <div align="center">
    <canvas id="gauge1"> </canvas>
    <!-- 2nd gauge -->
    <canvas id="gauge2"> </canvas>
    <!-- 3rd gauge -->
    <canvas id="gauge3"> </canvas>
  </div>

  <!-- <div id="console"> </div> -->
  <h3 align="center"> on Time: <span id="time"> </span> </h3>
  <div id="myDiv"></div>
  <hr>

<script>
  /* JAVASCRIPT CODE GOES HERE */
  var streamPlot = document.getElementById('myDiv');
  var ctime = document.getElementById('time');
```

```

var xArray = [], // time of data arrival
y1Track = [], // value of sensor 1 : temperature
y2Track = [], // value of sensor 2 : humidity
y3Track = [], // value of sensor 3 : luminosity
numPts = 100, // number of data points in x-axis
dtda = [], // 1 x 4 array : [date, data1, data2, data3] from sensors
preX = -1,
preY = -1,
preZ = -1,
initFlag = true;

var socket = io.connect('http://localhost:3000'); // port = 3000
socket.on('connect', function () {
  socket.on('message', function (msg) {
    // initial plot
    if([16]_____msg[0]!=" && initFlag){
      dtda[0]=msg[0];
      dtda[1]=parseFloat(msg[1]); // Temperature
      dtda[2]=parseFloat(msg[2]); // Humidity
      dtda[3]=parseFloat(msg[3]); // Luminosity
      init();
      initFlag=[17]_____false;
    }
    // Convert value to number
    dtda[0]=msg[0];
    dtda[1] = parseFloat(msg[1]);
    dtda[2] = parseFloat(msg[2]);
    dtda[3] = parseFloat(msg[3]);

    // Only when any of temperature or Luminosity is different from the previous one,
    // the screen is redrawn, when any change occurs.
    if (dtda[1] != preX || dtda[2] != preY || dtda[3] != preZ) {
      preX = dtda[1];
      preY = dtda[2];
      preZ = dtda[3];
    }
  });
});

```

```

// when new data is coming, keep on streaming
    ctime.innerHTML = dtda[0];
    gauge_temp.setValue(dtda[1]) // temp gauge
    gauge_humi.setValue(dtda[2]); // lux gauge
    gauge_lux.setValue(dtda[3]); // lux gauge
    //
    xArray = xArray.concat(dtda[0]);
    xArray.splice(0, 1); // remove the oldest data
    y1Track = y1Track.concat(dtda[1]);
    y1Track.splice(0, 1); // remove the oldest data
    y2Track = y2Track.concat(dtda[2]);
    y2Track.splice(0, 1);
    y3Track = y3Track.concat(dtda[3]);
    y3Track.splice(0, 1);

    var update = {
        x: [xArray, xArray, xArray],
        y: [y1Track, y2Track, y3Track]
    }

    Plotly.update([18]_____streamPlot, update);
}

});

});

function init() { // initial screen ()
    // starting point : first data (temp, humi, lux)
    for ( i = 0; i < numPts; i++ ) {
        xArray.push(dtda[0]); // date
        y1Track.push(dtda[1]); // sensor 1 (temp)
        y2Track.push(dtda[2]); // sensor 2 (humi)
        y3Track.push(dtda[3]); // sensor 3 (lux)
    }

    Plotly.plot([18]_____streamPlot, data, layout);
}

```

<pre> var data = [{   x : xArray,   y : y1Track,   name : 'temperature',   mode: "markers+ lines",   line: {     color: "#1f77b4",     width: 1   },   marker: {     color: "rgb(255, 0, 0)",     size: 6,     line: {       color: "black",       width: 0.5     }   } }, {   x : xArray,   y : y2Track,   name : 'humidity',   xaxis: 'x2',   yaxis : 'y2',   mode: "markers+ lines",   line: {     color: "#1f77b4",     width: 1   },   marker: {     color: "rgb(0, 0, 255)",     size: 6,     line: {       color: "black",       width: 0.5     }   } }, {   x : xArray,   y : y3Track,   name : 'luminosity',   xaxis: 'x3',   yaxis : 'y3',   mode: "markers+ lines", </pre>	<pre>     line: {       color: "#1f77b4",       width: 1     },     marker: {       color: "rgb(0, 255, 0)",       size: 6,       line: {         color: "black",         width: 0.5       }     }   } }]; var layout = {   xaxis : {     title : 'time',     domain : [0, 1]   },   yaxis : {     title : 'temp (°C)',     domain : [0, 0.3],     range : [-30, 50]   },   xaxis2 : {     title : "",     domain : [0, 1],     position : <b>[19]_____0.35,</b> <b>[20]_____showticklabels: false</b>   },   yaxis2 : {     title : 'humi (%)',     domain : [0.35, 0.65],     range : [0, 100]   },   xaxis3 : {     title : "",     domain : [0, 1],     position : 0.7, <b>[20]_____showticklabels: false</b>   },   yaxis3 : {     title : 'lumi (lux)',     domain : [0.7, 1],     range : [0, 500]   } }; </pre>
---	---

16. 네트워크 소켓에서 message가 처음 들어오는 경우를 확인하는 조건문은?

- A. `msg[0]!=" || initFlag`
- B. `msg[0]==" || initFlag`
- C. `msg[0]!=" && initFlag`
- D. `msg[0]==" && initFlag`

17. `initFlag` = ?

--- ( `false` )

18. 그래프가 그려지고 갱신되는 객체명은?

--- ( `streamPlot` )

19. 두 번째 그래프(습도 그래프)의 적절한 `position` 값은?

--- ( `0.35` )

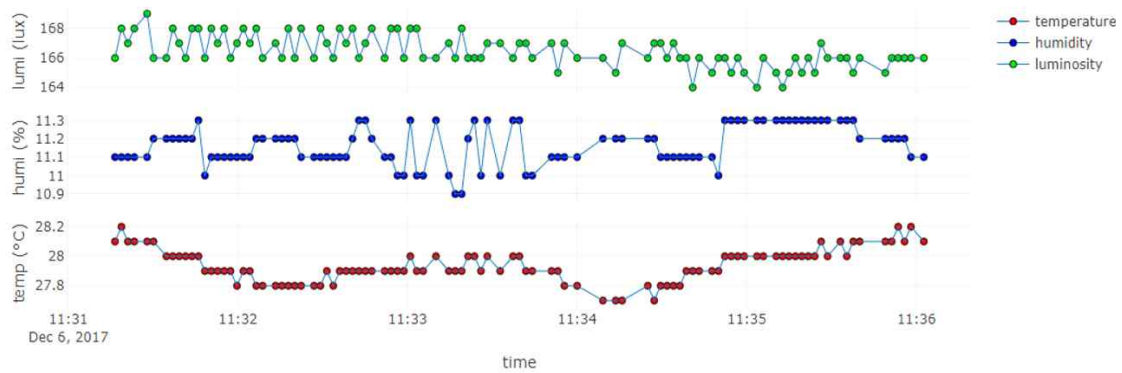
20. 두 번째, 세 번째 그래프의 가로 축에 시간이 나타나지 않게하는 설정은?

- A. `showtickvalues: false`
- B. `showticklabels: false`
- C. `showxtickvalues: false`
- D. `showxticklabels: false`

## Real-time Weather Station from sensors



on Time: 2017-12-06 11:36:02.639



===== 보람찬 겨울 방학을 보내세요 ^-^