9th November 2017

APIs:

Host from URL

Use callback functions to prevent your program from freezing up:

    Load the json in a separate function rather than in the draw function

Example 1:

var myData;

var url;

function setup(){

   url = "http://api.open-notify.org/iss-now.json";

   myData = loadJSON(url);

   console.log(myData);

}

function draw(){

   if(myData){

    console.log(myData);

   }

}

function mouseClicked(){

   //loadJSON function takes the packets from the first argument and passes it through the second argument function

   loadJSON(url, dataResponse);

}

function dataResponse(tempData) {

   myData = tempData;

   console.log(tempData);

}

Example 2:

var myData;

var url;

var baseUrl;

var appId;

var query;

function setup(){

   baseUrl = "http://api.openweathermap.org/data/2.5/weather?";

   appId = "appid=4c339eb32322fef3772fdbde1686ccce";

   query = "q=London,uk"

   url = baseUrl + query + "&" + appId;

   console.log(url);

   myData = loadJSON(url);

   console.log(myData);

}

function draw(){

   if(myData){

   }

}

function mouseClicked(){

   //loadJSON function takes the packets from the first argument and passes it through the second argument function

   loadJSON(url, dataResponse);

}

function dataResponse(tempData) {

   myData = tempData;

   console.log(tempData);

}

Example 3:

var myData;

var url;

function setup(){

   url = "http://api.open-notify.org/iss-now.json";

   myData = loadJSON(url);

   console.log(myData);

   setInterval(queryData, 4000);

}

function draw(){

   if(myData){

      var xPos = myData.iss\_position;

      // console.log(xPos);

   } else {

      console.log("Bad data!");

   }

}

function mouseClicked(){

   //loadJSON function takes the packets from the first argument and passes it through the second argument function

   loadJSON(url, dataResponse);

}

//callback function. can't be called except through the loadJSON

function dataResponse(tempData) {

   myData = tempData;

   console.log(tempData);

}

function queryData(){

   loadJSON(url, dataResponse);

}