```
PImage mapimg; //class for storing image

int cenlat = 0; //declare center latitude
int cenlon = 0; //declare center longitude

int ww = 1024; //declare width (using 1024 because of format from mapbox.com)
int hh = 512; //declare height (using 512 because of format from mapbox.com)

int zoom = 1;
String[] earthquakes;

float mercX(float lon) {//mapping X (longitude)
  lon = radians(lon);//translating information from data from degrees to radians
  float a = (256 / PI) * pow(2, zoom);
  float b = lon + PI;
  return a * b;
}

float mercY(float lat) {//mappin Y (latitude)
  lat = radians(lat);
  float a = (256 / PI) * pow(2, zoom);
  float b = tan(PI / 4 + lat / 2);
  float c = PI - log(b);
  return a * c;
}


void setup() {
  //size(1024, 512);
  fullScreen();
  String url = "https://api.mapbox.com/styles/v1/mapbox/dark-v9/static/" +
    cenlon + "," + cenlat + "," + zoom + "/" +
    ww + "x" + hh +

"?access_token=pk.eyJ1IjoiYW9pZmVoYXJ0ZW4iLCJhIjoiY2pnZjFemM5MmxnbTJ3bzVlOGh2
djJ2ZiJ9.AlGIeRY6u7icv3LuaJ25qQ";  //examines each part of the data
  mapimg = loadImage(url, "jpg");
  println(url);
  // earthquakes =
loadStrings("http://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/all_hour.csv");//
view all earthquakes in the past hour
  earthquakes =
loadStrings("http://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/all_day.csv");//vi
ew all earthquakes in the past day
  //earthquakes =
loadStrings("http://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/all_month.csv");
//view all earthquakes in the past month
```

```
  translate(width / 2, height / 2);//placing the map in the center
  imageMode(CENTER);
  image(mapimg, 0, 0);

  float cx = mercX(cenlon);
  float cy = mercY(cenlat);

    for (int i = 1; i < earthquakes.length; i++) {//taking the
    String[] data = earthquakes[i].split(",");
    //console.log(data);
    float lat = float(data[1]);//taking the latitude from the data
    float lon = float(data[2]);//taking the longitude from the data
    float mag = float(data[4]);//taking the magnitude from the data
    float x = mercX(lon) - cx;
    float y = mercY(lat) - cy;

    mag = pow(10, mag);
    mag = sqrt(mag);
    float magmax = sqrt(pow(10, 10));
    float d = map(mag, 0, magmax, 0, 180);
    stroke(255, 0, 255);//placing the dots on the map
    fill(255, 0, 255, 200);
    ellipse(x, y, d, d);

  }
}

void draw(){//interaction - quit application when a key is pressed

  if (keyPressed == true){
    exit();
  }
}
```