

# Vizualizacija podataka

**LV5**

**PROJEKCIJE**

Student: Kristijan Koščak

Smjer: DRC, 1.godina

Datum: 24.04.2020.

## Zadatak 1:



Zadarska

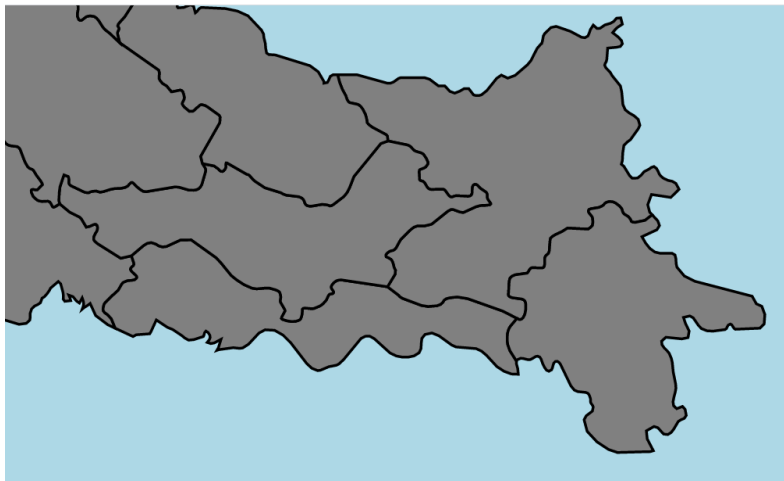


Vukovarsko-Srijemska

```
d3.json("cro_regv3.json", function (error, cro) {
  var data = topojson.feature(cro, cro.objects.layer1);
  var states = svg.selectAll("path.county")
    .data(data.features)
    .enter()
    .append("path")
    .attr("class", "county")
    .attr("id", function (d) { return d.id; })
    .attr("d", path).style("fill", "gray")
    .style("stroke", "black")
    .style("stroke-width", 1)
    .style("stroke-opacity", 1);
  states.on("mouseover", function (d) { // registira pokret iznad
    $("#zupanija").empty()           // svake zupanije
    $("#zupanija").append("<span>" + d.properties.name + "</span>")
  })
});
```

Koristeći kod iz predloška napravili smo traženo u zadatku. Kako bi navedeno bilo moguće morali smo dodati par linija koda kojim bi registrirali pokrete miša nad svakom županijom što je dano kodom iznad. Dodane su strelice crvene boje kako bi prikazali gdje je miš u trenutku ispisa županije jer prilikom slikanja ne pokazuje se pokazivač miša. Lijeva slika ispisuje Zadarska dok desna ispisuje Vukovarsko-Srijemska.

## Zadatak 2:



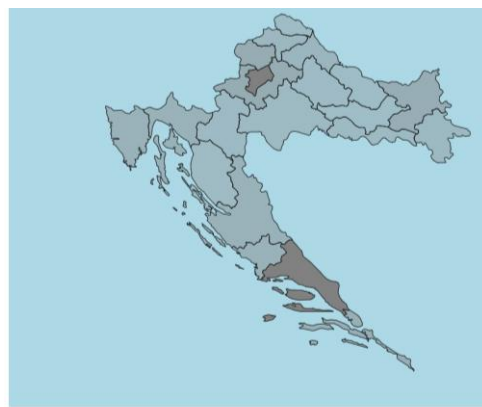
```
var svg = d3.select("body").append("svg")
    .attr("width", width)
    .attr("height", height)
    .style("background", "lightblue")
    .call(d3.behavior.zoom().on("zoom", function () {
        svg.attr("transform", "translate(" + d3.event.translate + ") " +
            "scale(" + d3.event.scale + ")")
    })))
```

Kako bi omogućili povećavanje i smanjivanje , morali smo preurediti kod iz predloška. Na SVG element smo dodali funkciju *call* kojom omogućujemo postavljanje koordinata x i y unutar SVG elementa te stupanj zumiranja nakon povećavanja ili smanjivanja.

## Zadatak 3:



Županija: Požeško-slavonska Broj zaraženih: 3



Županija: Grad Zagreb Broj zaraženih: 479

Požeško-slavonska županija , broj zaraženih 3 – Grad Zagreb, broj raženih 479

```

var opacity = d3.scale.linear()
    .domain([3, 459])
    .range([0.2, 1]);
d3.json("cro_regv3.json", function (error, cro) {
    var data = topojson.feature(cro, cro.objects.layer1);
    var states = svg.selectAll("path.county")
        .data(data.features)
        .enter()
        .append("path")
        .attr("class", "county")
        .attr("id", function (d) { return d.id; })
        .attr("d", path).style("fill", "gray")
        .style("stroke", "black")
        .style("stroke-width", 1)
        .style("stroke-opacity", 1)
        .style("fill-
opacity", function (d) { return opacity(d.properties.brZarazenih); });
    states.on("mouseover", function (d) {
        $("#zupanija").empty()
        $("#zupanija").append("<span>Zupanija: " + d.properties.name + "
</span>")
        $("#zupanija").append("<span> Broj zarazenih: " + d.properties.br
Zarazenih + "</span>")
    })
});

```

Za podatak koji ćemo prikazati na karti uz svaku županiju, odabran je podatak o broju zaraženih korona virusom. Do podataka je lako doći, preuzeti su sa stranice <https://www.24sata.hr/> . Svaki podatak za pojedinu županiju je ubačen ručno te uz ime županije dodan je i taj podatak prilikom ispisa. Kako bi bojom istaknuli broj zaraženih izradili smo linearnu skalu koju smo koristili kod prozirnosti. Domena se kreće u rasponu od najmanjeg broja zaraženih do najvećeg. Raspon odnosno vrijednost koja će se predat pozivom skale se kreće od 0.2 do 1 ovisno o broju zaraženih. Kod je jako sličan kodu iz preostalih zadataka uz male preinake kao što je vidljivo na danom kodu.

#### Zadatak 4:



```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Zadatak 4</title>

  <script src="https://d3js.org/d3.v3.min.js"></script>
  <script src="http://d3js.org/topojson.v1.min.js"></script>
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"
"></script>

  <style>
    .background {
      fill: none;
      pointer-events: all;
    }
  </style>

</head>

<body>
  <script>
    var width = 960;
```

```

var height = 700;

var projection = d3.geo.mercator()
    .center([0, 10])
    .scale(6000)
    .translate([17600, 4500])
    .rotate([-180, 0]);

var path = d3.geo.path()
    .projection(projection);

var svg = d3.select("body").append("svg")
    .attr("width", width)
    .attr("height", height)
    .style("background", "lightblue");

var opacity = d3.scale.linear()
    .domain([3, 459])
    .range([0.3, 1]);

svg.append("rect")
    .attr("class", "background")
    .attr("width", width)
    .attr("height", height);

var g = svg.append("g");

d3.json("cro_regv3.json", function (error, cro) {
    var data = topojson.feature(cro, cro.objects.layer1);
    g.selectAll("path")
        .data(data.features)
        .enter()
        .append("path")
        .attr("class", "county")
        .attr("id", function (d) { return d.id; })
        .attr("d", path).style("fill", "gray")
        .style("stroke", "black")
        .style("stroke-width", 1)
        .on("click", showCounty)
        .style("stroke-opacity", 1)
        .style("fill-
opacity", function (d) { return opacity(d.properties.brZarazeni); });
    });
    function showCounty(d) {
        var bounds = path.bounds(d),

```

```

        dx = bounds[1][0] - bounds[0][0],
        dy = bounds[1][1] - bounds[0][1],
        x = (bounds[0][0] + bounds[1][0]) / 2,
        y = (bounds[0][1] + bounds[1][1]) / 2,
        scale = 0.9 / Math.max(dx / width, dy / height),
        translate = [width / 2 - scale * x, height / 2 - scale * y];
    g.transition()
      .duration(750)
      .attr("transform", "translate(" + translate + ") " + " scale(" + scale + ")");
    g.append("text")
      .attr("dx", x)
      .attr("dy", y)
      .attr("font-size", "4")
      .attr("fill", "white")
      .attr("text-anchor", "middle")
      .text("Zupanija: " + d.properties.name + ", broj zarazenih: " + d
.properties.brZarazenih);
    g.transition()
      .duration(750)
      .delay(10000)
      .attr("transform", "")
      .select("text").remove();
  }

</script>
</body>

</html>

```

Zbog povećanja složenosti ovog zadatka u odnosu na prethodne , postaviti ćemo cijeli kod. Prvo smo dodali `g` element, unutar kojega smo ubacili sve županije te postavili metodu *showCountry* koja se pokreće na klik. Metoda prima jedan parametar tj. županiju, na kojoj je pokrenuta metoda. Zatim se „pravi“ kocka(granica) unutar koje se nalazi županija te se računaju njezina dužina i visina i središte u koordinatnom sustavu. Nakon toga se računaju položaj županije unutar `g` elementa i stupanj zumiranja prema kojima se translata. Dodali smo tekst koji prikazuje ime županije i broj zaraženih te nakon 10 sekundi vratili se u početno stanje. Moguće je prikazati i više parametara pojedine županije, ali je potrebno provjeriti ime parametra u svojstvima svake županije json dokumenta.