# TÉRKÉPEK MEGJELENÍTÉSE SVG FORMÁTUMBAN A WEBEN

# MI IS AZ SVG?

- Rövidítés: Scalable Vector Graphics, azaz Skálázható Vektor Grafika
- XML formátumban definiált vektoros grafika
- Webes megjelenítésre használjuk

# SVG ÉS QGIS

• Alapértelmezetten nem lehet QGIS-ben svg formátumban elmenteni a térképeket, de letölthető egy SimpleSvg nevű plugin, aminek segítségével már elérhető ez a funkció

## SIMPLESVG PLUGIN

- Telepítés:
  - Válasszuk ki a "Plugins -> Manage and Install Plugins…" menüpontot
  - Keressünk rá arra, hogy: SimpleSvg
  - Majd az Install Plugin gombbal telepítsük
- Használat:
  - A menüsorban megjelent egy "svg <>" feliratú gomb



• Ha el akarunk menteni egy képet svg formátumban, akkor csak erre kell kattintani, ki kell választani a helyet ahova menteni szeretnénk, és meg is vagyunk

	<pre>▷ <svg:g id="plaat&lt;br&gt;&lt;! end of id/lay&lt;br&gt;▷ &lt;svg:g id=" plaat<br=""><!-- end of id/lay<br--><!-- QGIS extent I<br-->♡ &lt;svg:g id="qgisv&lt;br&gt;<svg:path id="&lt;/pre"></svg:path></svg:g></pre>	For Points it can only save a point as a small colored circle. Note: plugin uses screen (pixel) coordinates for svg coordinates. Default to ONLY use the objects that are in current map window. Uncheck checkbox below to have ALL your vector objects (at current scale). NOT working for raster data.
-		Resize Map or Set Extent
-		Only features in current mapview (uncheck to have ALL)
:		C:/Users/Abraham/Downloads/pas.svg Browse
		OK Cancel Help

 Van lehetőség átméretezni a térképünket, és kiválasztani, hogy csak a jelenleg megjelenített shape fájlok legyenek rajta az SVG képünkön, vagy az összes betöltött

# EGY TÉRKÉP EXPORTÁLÁSA SVG-BE

• Van egy térképünk 3 shape (foutak\_uj.shp, nagytavak\_uj.shp, megye\_uj.shp) fájllal, és ezt szeretnénk SVG formátumban lementeni



• Ezt meg is tesszük az előbb említett módszerrel, így kapunk is egy szép SVG fájlt, amit azonnal megnyitunk:



- Mielőtt rátérnék a formátum ismertetésére, egyből észre lehet venni pár alapvető hibát:
  - o vonalak nem jelennek meg
  - o határoló vonalak alig észrevehetőek

SVG FÁJL FORMÁTUMA

#### EGY EGYSZERŰ PÉLDA

```
<svg xmlns="http://www.w3.org/2000/svg"
    xmlns:xlink="http://www.w3.org/1999/xlink">
    <rect x="10" y="10" height="100" width="100"
    style="stroke:#ff0000; fill: #0000ff"/>
```

</svg>

- az elején megmondjuk, hogy ez egy svg fájl, és deklarálunk egy-két dolgot:
  - o xmlns="http://www.w3.org/2000/svg svg fájl
  - xmlns:xlink megengedi hogy lehessen hyperlinkeket létrehozni XML dokumentumon belül
- Ez egy téglalapot hoz létre (rect), ami a 10, 10 pontból indul, és 100 a magassága és a vastagsága is. A vonal színe: #ffo000, ami piros, és a kitöltése: #0000ff, ami pedig kék.
- Lényegében ennyi kell egy svg képhez. Lehet ezenfelül különböző alakzatokat létrehozni: kör, poligon, polilyne; lehet ezeket összefűzni: g; különféleképpen kitölteni stb.

# MI PÉLDÁNK

#### DEKLARÁCIÓ

<?xml version="1.0" standalone="no"?>

<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN" "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">

<svg xmlns="http://www.w3.org/2000/svg" xmlns:inkscape="http://www.inkscape.org/namespaces/inkscape" xmlns:xlink="http://www.w3.org/1999/xlink" viewBox = "0 0 1113 494" version = "1.1">

- Első sorban megmondjuk, hogy ez egy xml fájl lesz, nem egyedülálló lesz, hanem használni fog külső DTD-t, azaz Document Type Definition-t.
- Utána megmondjuk, hogy mi lesz a dokumentum típusa: svg
- Majd jön ugyanaz a rész, amit az előző példánkban is láttunk már. De most egy inkscape nevű szoftvert is beimportálunk, ami vektorgrafikus elemek megjelenítésére, szerkesztésére való.

## MEGYE

- g: arra jó, hogy különböző alakzatokat csoportosítsunk össze
- **path**: arra jó, hogy bonyolultabb alakzatokat hozzunk létre pontok, vonalak, görbék segítségével
- Az első két sorban mondjuk meg, hogy ez lesz a megye\_uj layer-ünk, beállítjuk hozzá a vonal vastagságot, a kitöltési színt, és egyéb ilyen dolgokat. (Megj.: a vonal vastagság "none", ami kicsit átláthatatlanná teszi szerintem a layert, ezért ezt érdemes lesz később akár kézzel is átírni)
- Ebbe a layer-be fog beletartozni az össze alatta levő kisebb alakzat (megye\_uj\_1....megye\_uj\_20)
- Utána hozzuk létre a megye\_uj\_1 id-jű alakzatot, ami egy bonyolult alakzat lesz, path megadással:
  - M hova rakjuk le a rajzoló tollunkat először
  - L vonal, azaz hogy hova fogjuk a jelenlegi helyzetből az egyenes vonalat húzni
- Utána hozzuk létre a megye\_uj\_2-t, és ezt is ugyanígy rajzoljuk

#### NAGYTAVAK

• a Nagytavak layert a Megye layerhez nagyon hasonlóan jön létre, ezért azt nem írom le újra

# FŐ UTAK

• A Fő utak layer érdekessége, hogy az csak vonalakból áll, ezért lesz is vele egy kisebb probléma

```
<g id="foutak_uj" inkscape:groupmode="layer" inkscape:label="foutak_uj">
<g stroke="none" fill="none" stroke-linejoin="round" stroke-width="0.26">
<g id="foutak_uj_1" >
<path d="M 199,270 L 201,271 L 203,271 L 205,271
```

• Ez is nagyon hasonló az előző példákhoz, de itt az figyelhető meg, hogy mivel vonalakról van szó, ezért a fill is "none"-re van állítva, így meg sem fognak jelenni az utak. Ezért én kézzel beleírtam fájlba, és átállítottam a stroke-ot "black"-re.

# EREDETI ÉS JAVÍTOTT VERZIÓ (CHROME ALATT)



• Lényeges különbség, hogy én mindenhol ahol csak lehet kitöltöttem a stroke-ot valamilyen színnel, hogy jobban el lehessen különíteni, vagy egyáltalán meg lehessen jeleníteni az alakzatokat.



# BÖNGÉSZŐK ÖSSZEHASONLÍTÁSA

#### CHROME

- a lényegi különbség, amit felhasználói szemmel észre lehet venni a különböző böngészők között, az a nagyítás lehetősége
- Chrome esetében akármennyire próbálkozunk, nem lehet egyáltalán ránagyítani a képekre

#### MOZILLA

 Mozilla esetében egyel jobb a helyzet, mert már lehet nagyítani. Annyi a gond, hogy nem lehet kiválasztani, hogy mi legyen a fókuszban, ezért először kell nagyítani, utána pedig megkeresni a helyet, amit fel akartunk nagyítani.

#### INTERNET EXPLORER

 Meglepő módón, az Internet Explorer volt a legsikeresebb, mivel itt a nagyításnak még fókuszt is meg lehet adni.

# SVG LEHETŐSÉGEI

- Az SVG-nek számos előnye van riválisaival szemben, de az elterjedése még folyamatban van. Már jó ideje a legnagyobb böngészők támogatják, és sok egyéb szoftver is, köztük több adatbáziskezelő is. Az egyik leglényegesebb, hogy az Oracle is támogatja, és igen egyszerűen lehet már alakzatokat importálni, exportálni SVG-be.
- Az SVG egyik legnagyobb riválisa a HTML5 canvas. Míg az SVG vektor grafikus, addig a canvas raszter grafikus megjelenítést biztosít. Valójában a legtöbb helyen cavast használnak még, de az SVG is elterjedőben van.

SVG	Canvas
Vektor grafikus	Raszter grafikus
<b>Skálázható</b> : ez mostanában lesz egyre fontosabb, mivel egyre több ember használ hatalmas felbontású eszközöket	Nem skálázható
De SVG-t bárki létre tud hozni, főleg hogy különféle <b>eszközök</b> is vannak már hozzá: Inkscape, OpenOffice Draw, LibreOffice Draw stb.	Canvas egy "programozható kép". Ezért, aki Canvast akart létrehozni, annak előbb meg kell tanulnia az API-t használni.
SVG-t akár <b>szerver oldalon</b> is létre lehet hozni, és később kliens oldalon megjeleníteni, mivel csak egy egyszerű XML fájl.	Canvas-t csak kliens oldali JavaScripttel lehet szerkeszteni.

SVG fájlokat az emberek, és <b>gépek</b> is nagyon könnyen <b>megértik</b> . Ez nagyon fontos lesz mostanában, ahogy a szematikus web, a gépi tanulás, és a Big Data egyre nagyobb teret nyer magának.	
<b>DOM kezelés</b> : ugyanúgy lehet kezelni egy SVG alakzatot, mint egy HTML blokkot. Lehet hozzá eseménykezelést rendelni, vagy egyszerűen HTML-ből manipulálni.	Canvast pedig nem lehet ilyen egyszerűen kezelni. A saját API-jában kell az elemeket manipulálni, stb.
SVG támogatja a CSS-t, a különféle webes fontokat, és a JavaScript technológiákat is.	
Kevesebb, egyszerűbb elemek megjelenítésére viszont az SVG alkalmasabb.	Ha rengeteg kis apró dolog alakzatot kell megjeleníteni, és akár animálni is, akkor a Canvas sokkal gyorsabb megoldást tud nyújtani. Ezért a webes játékokhoz ez sokkal jobban illik.