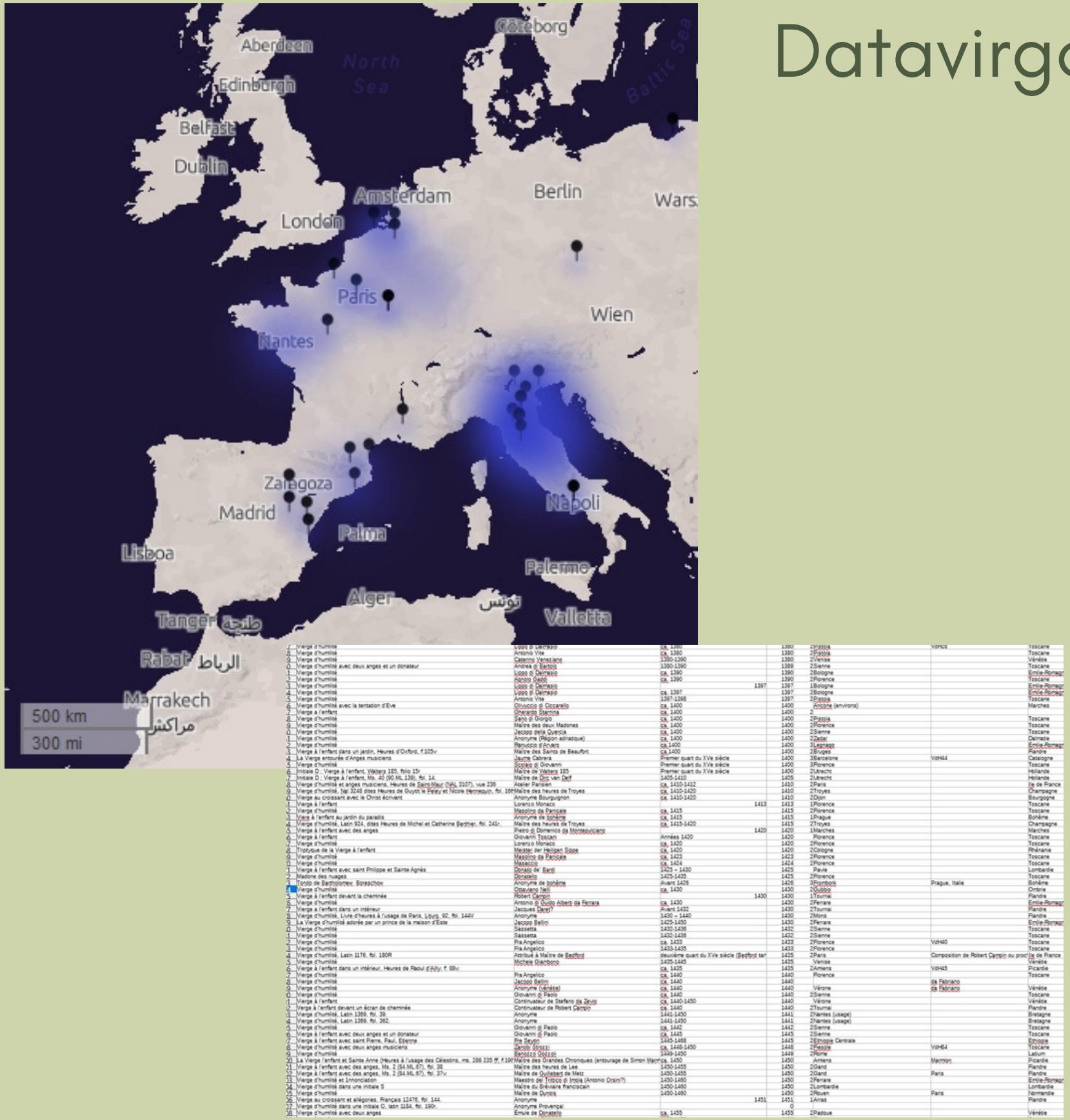


PSL - DHAI intensive week

MATSUKATA PROJECT

Introduction to interactive mapping and
programming

Quentin Bernet - 2022



Datavirgo project



Planning

11

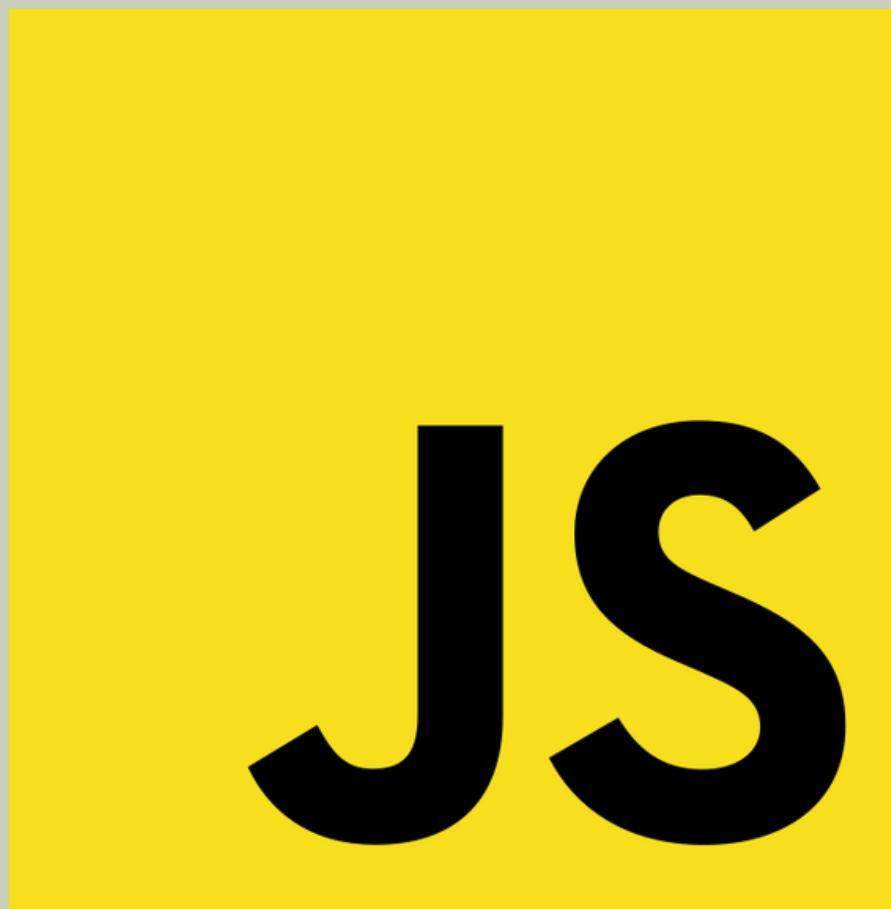
First session : Presentation, typology and basics

Second session : How to print structured data with python and first steps with leaflet

Third session : Programming interactivity with Javascript

Our two programming languages

Python and Javascript.

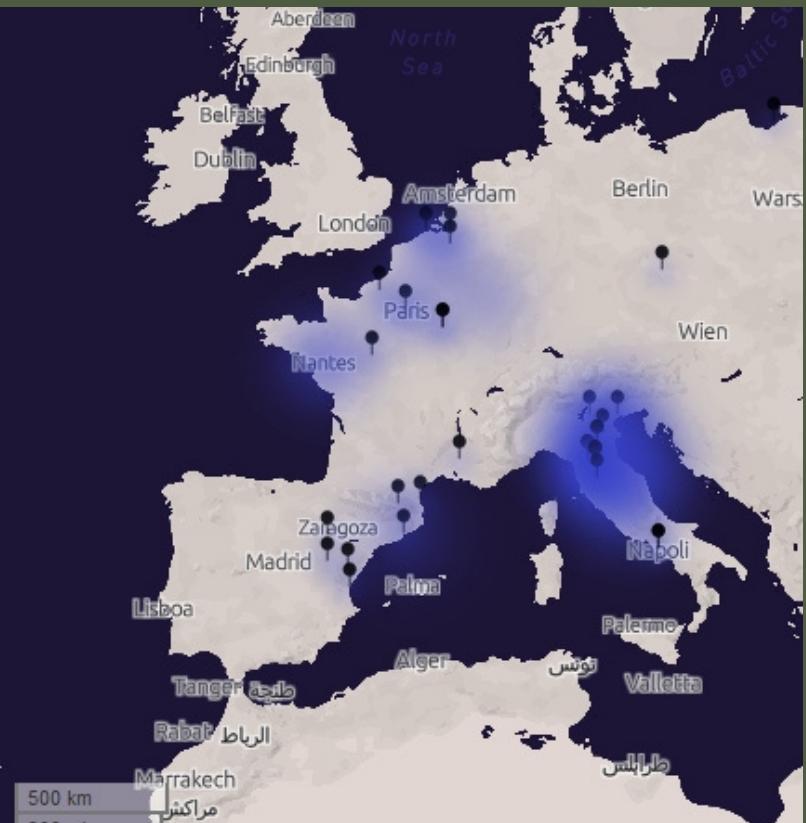


	A	B	C
153	ID_E_32	Lebourg, Albert	La Seine à La Bouille
154	ID_E_32	Lépine, Stanislas Victor Edouard	La Seine à Paris
155	ID_E_32	Marquet, Albert	Les Sables d'Olonne
156	ID_E_32	Monet, Claude	Vétheuil
157	ID_E_32	Monet, Claude	Femmes en barque
158	ID_E_32	Monet, Claude	Neige à Argenteuil
159	ID_E_32	Monet, Claude	Les nymphéas
160	ID_E_32	Monticelli, Adolphe Joseph Thomas	Les Martigues
161	ID_E_32	Renoir, Pierre Auguste	Sous-bois
162	ID_E_32	Sisley, Paul	Hortfeur
163	ID_E_4	Gauguin, Paul	La famiglia Schuffenecker
164	ID_E_4	Gauguin, Paul	Piccole bretone in piedi in riva al mare
165	ID_E_4	Gauguin, Paul	Mulino in Bretagna
166	ID_E_4	Gauguin, Paul	"Vairumati"
167	ID_E_4	Manet, Edouard	La "Servouse de bocks" della birreria di Reichshoffen
168	ID_E_4	Monet, Claude	Giovani donne in barca
169	ID_E_4	Monet, Claude	Effetti di neve ad Argenteuil
170	ID_E_4	Pissarro, Camille	La strada lungo la ferrovia
171	ID_E_4	Renoir, Pierre Auguste	Le Parigine travestite da Algerine
172	ID_E_4	Van Gogh, Vincent	La camera dell'artista ad Arles
173	ID_E_5	Gauguin, Paul	Stilleben mit Fächer, 1889 ?
174	ID_E_5	Gauguin, Paul	Die Familie Schuffenecker
175	ID_E_6	Degas, Edgar	Femme se coiffant
176	ID_E_7	Courbet, Gustave	Bordende fra Flagey vender hjem fra merkedet
177	ID_E_8	Gauguin, Paul	Paysage de Bretagne
178	ID_E_8	Gauguin, Paul	Natura morta à l'éventail
179	ID_E_8	Gauguin, Paul	Portrait du peintre Slevinsk
180	ID_E_8	Gauguin, Paul	La famiglia Schuffenecker
181	ID_E_8	Gauguin, Paul	Petite Bretonnes debutt abord du bord de la mer
182	ID_E_8	Gauguin, Paul	Paysage de Bretagne (Le Moulin)
183	ID_E_8	Gauguin, Paul	Vairumati
184	ID_E_9	Gauguin, Paul	Landscape in Brittany
185	ID_E_9	Gauguin, Paul	Breton Girls by the Sea
186	ID_E_9	Gauguin, Paul	Mill in Brittany
187	ID_E_9	Lépine, Stanislas Victor Edouard	The Banks of the Seine
188	ID_E_9	Monet, Claude	Snow Scene at Argenteuil
189	ID_E_9	Monet, Claude	Charing Cross Bridge, London
190	ID_E_9	Monet, Claude	The Avenue
191	ID_E_9	Monet, Claude	Water-lillies (Nymphéas)
192	ID_E_9	Monticelli, Adolphe Joseph Thomas	Les Martigues
193	ID_E_9	Pissarro, Camille	The Road by the Railway
194	ID_E_9	Puyvis de Chavannes, Pierre	The Poor Fisherman
			Le Pauvre P

```

"img":<a href="https://collection.nmwa.go.
"markerHtmlStyles":"background-color: rgb(
},
    {"loc":[45.766606, 4.834343],
"label":"Picasso, Pablo, La liseuse.",
"title":"Picasso, Pablo, La liseuse, Picas
"artiste":"Picasso, Pablo",
"expo":"Picasso. Festival de Lyon-Charbonn
"start":"1953-01-1T00:00:00Z",
"end":"1953-12-31T00:00:00Z",
"color": "rgb(200,25,1)",
"img":<a href="https://www.centreopempidou.
"markerHtmlStyles":"background-color: rgb(
},
    {"loc":[48.8425186, 2.3435374],
"label":"Gauguin, Paul, Petit paysage bret
"title":"Gauguin, Paul, Petit paysage bret
"artiste":"Gauguin, Paul",
"expo":"Pérennité de l'art gaulois",
"start":"1955-02-1T00:00:00Z",

```



Python algorithm(CSV) → JS code → HTML document

What do we need to build a statement?

print("hello!")

action+object

if x<5 :

print("hello!")

operator + action + object

Primary objects in programming

String (str) : "Hello"

Integer (int) : 12

Float (float) : 12.5

Boolean (bool) : true/false

Variable :

x = "hello"

Python

var x = "hello"

Javascript

—

Groups, lists and datasets

List :

```
list = [2, "hello", 34.6]
```

Tuple :

```
tuple = (1, 5, 6, 8, "hello")
```

Array :

```
array = ["hello", "flower", "capricorn"]
```

Dictionary :

```
thisdict = {
```

```
    "brand": "Ford",
```

```
    "model": "Mustang",
```

```
    "year": 1964
```

```
}
```

Basic operators

Arithmetic operators : + / - * %

Assignment operators : += -= /=

Comparison operators : == != > <

Logical operators : and/or/not

Identity operators : is / is not

Membership operators : in/ not in

Comparative operators :

& And

| Or

~ Not

Iterables

For :

```
for i in my_csv : ...
```

```
for i in my_csv[2:] : ...
```

```
for i in range (18,2) : ...
```

While :

```
while x<7 :
```

```
x = +1
```

```
print(x)
```

Switch : *Javascript*

Conditional statements

If/else :

if 8 < 6 :

print "hello"

elif my_list[2] == "cheval":

print "hello" *Python*

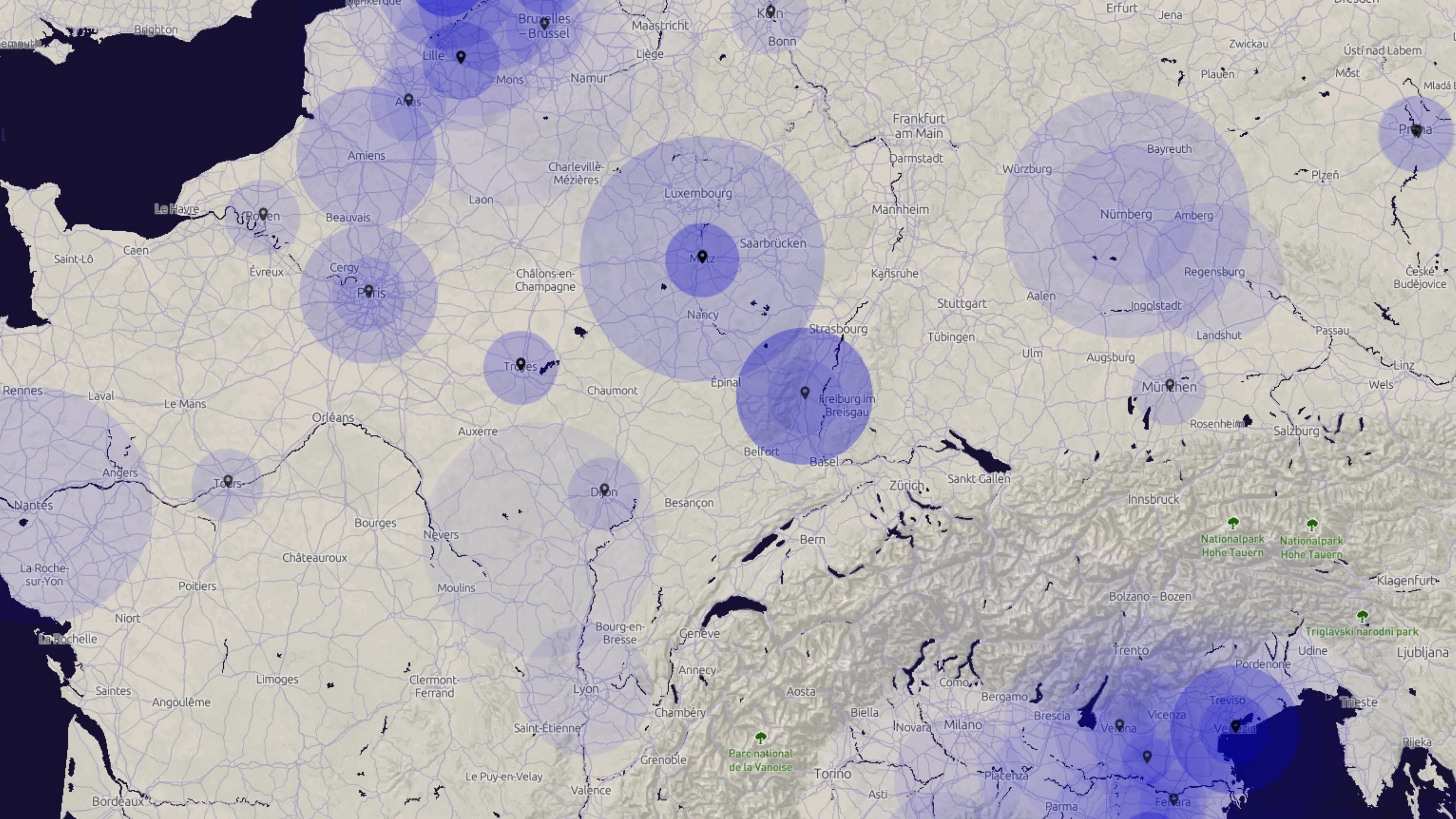
else if (x == 2)

{print "hello"} *Javascript*

else :

print "hello"

```
switch (objects_category) {  
    case 'monnaie','monnaies':  
        Monnaies.addLayer(marker);  
        break;  
    case 'gravure','gravures':  
        Gravures.addLayer(marker);  
        break;  
    case 'livre','livres':  
        Livres.addLayer(marker);  
        break;  
    case 'document','documents':  
        Documents.addLayer(marker);  
        break;  
    default :  
        var none = none;  
}
```



The Algorithm



```
2
3     <!DOCTYPE html>
4     <html lang="en">
5         <head>
6             <meta charset="utf-8">
7             <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
8             <meta name="description" content="Label for Jean Tremblot - Photographies, 1925.">
9             <meta name="keywords" content="label, Jean Tremblot, photographies, 1925, documents, autre, M, 4.2">
```

```
23522     {"loc": [48.459579,1.563352],
23523     "title":"Jean Tremblot - Photographies , 1925.",
23524     "date":"1925",
23525     "categorie":"documents",
23526     "classe":"autre",
23527     "genre":"M",
23528     "indice":"4.2"
23529 },
23530     {"loc": [48.459579,1.563352],
23531     "title":"Jean Tremblot - Médailles , 1925.",
23532     "date":"1925",
23533     "categorie":"monnaies",
23534     "classe":"autre",
23535     "genre":"M",
23536     "indice":"4.2"
23537 },
23538     {"loc": [48.459579,1.563352],
```

```
91088     Etoffes.eachLayer(function (layer) {
91089
91090         if(layer.options.properties.date>=parseFloat(e[0])
91091             Etoffes.addLayer(layer);
91092         else {map.removeLayer(layer)})}
91093     })
91094
91095     map.removeLayer(searchLayer);
91096
91097
91098
91099     </script>
91100
91101     </body>
```



```
2
3     <!DOCTYPE html>
4     <html lang="en">
5         <head>
6             <meta charset="utf-8">
7             <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
8             <meta name="description" content="Django template system documentation">
9             <meta name="keywords" content="label, template, system, documentation">
```



```
91088     Etoffes.eachLayer(function (layer) {
91089         if(layer.options.properties.date>=parseFloat(e[0].date))
91090             Etoffes.addLayer(layer);
91091         else {map.removeLayer(layer)}}
91092     })
91093
91094     map.removeLayer(searchLayer);
91095
91096     map.removeLayer(searchLayer);
91097
91098
91099 </script>
91100
91101 </body>
```

HTML file - interactive map

