

# Initiation opendata, datavisualisation et liste de données «La fréquentation du métro parisien»

## Objectifs :

- Découvrir l'Opendata
- Utiliser le format XLS et être sensibilisé aux différents formats de l'Opendata
- Initiation aux listes de données sous tableur

## Ressources :

⇒ url :

- <http://data.ratp.fr/fr/les-donnees.html>
- <http://d3js.org/>
- <https://github.com/mbostock/d3/wiki/Tutorials>
- <https://github.com/mbostock/d3/wiki/Gallery>

⇒ Il s'agit de récupérer des données ouvertes dans un tableur, de les trier, de faire des calculs et de réaliser une visualisation

## L'open data (données ouvertes)

# Où trouver des données ?

- Les données ouvertes numérisées (sous différents formats : xls, xml...)
- L'Etat
  - <http://www.data.gouv.fr/>
  - [http://www.statistique-publique.fr/index.php?php\\_action=ACCUEIL](http://www.statistique-publique.fr/index.php?php_action=ACCUEIL)
  - <http://www.google.com/publicdata/directory>
- Les départements les hauts de seine (<http://opendata.hauts-de-seine.net/>)
- Les villes (Rennes, Nantes, Montpellier...)
- Les grands organismes publics tels l'INSEE
- Les organisations internationales : OCDE, UNESCO, L'ONU
- Les portails de données ouvertes : <http://www.data-publica.com/>
- Les médias : ex le [guardian](#)
- Les entreprises : la sncf <http://data.sncf.com/> la ratp : <http://data.ratp.fr/>
- ⇒ Un annuaire à consulter : <http://datacatalogs.org/>
- Les données que l'on constitue en les saisissant dans un tableur à partir de documents, de pdf, de sites web...

# Initiation opendata, datavisualisation et liste de données «La fréquentation du métro parisien»

## Comment sont présentées ces données ouvertes ?

11

Sous formes de fichiers aux formats suivants :

- CSV
- html
- [rdf](#) (exemple les données de la [bnf](#) sur [data.gouv](#))
- [rtf](#)
- [xls](#)
- [xml](#)
- Format compressé : zip

⇒ Dans cette séance le fichier XLS va être utilisé pour effectuer des calculs de fréquentation du métro parisien

### Traiter les données

1 Télécharger le **fichier xls** correspondant aux données du trafic annuel entrant par station (2011)

à l'url suivante : <http://data.ratp.fr/fr/les-donnees.html>

The screenshot shows the RATP Open Data interface. At the top, there is a navigation menu with 'La démarche', 'Les données', 'Actualités', 'Vos créations', and 'Développeurs'. Below the menu is a grid of icons representing various data categories and services. The main content area displays the dataset 'Trafic annuel entrant par station (2011)'. It includes a description: 'Ce jeu de donnée détaille le trafic des entrants directs sur le réseau ferré RATP.' Below the description, there is a license section for 'Etalab' with the category 'Voyageurs' and a download count of '18256 fois'. There are buttons to download the data in 'XLS' and 'PDF' formats. On the right side, there is a 'Trier par' section with options for 'Popularité', 'Date de mise à jour', 'Date de mise en ligne', and 'Alphabétique'. Below that is a 'Filtres' section with checkboxes for 'Licences' (RATP, ODBL, Etalab) and 'Formats' (XLS, XML). At the bottom, there is a 'Mise à jour depuis' section with radio buttons for '1 jour', '1 semaine', and '1 mois'.

# Initiation opendata, datavisualisation et liste de données

## «La fréquentation du métro parisien»

**2 Effectuer un nettoyage des données.** En effet, Il s'agit de conserver uniquement les données concernant le **métro parisien et de sa proche banlieue** et par conséquent de supprimer toutes celles concernant le **RER**

**3 Compléter les éléments non documentés pour obtenir ultérieurement des sous-totaux cohérents :**

- ⇒ modifier **Asnières** (qui apparaît au début) en **Asnières sur Seine** dans le champ **Ville**
- ⇒ ajouter **Créteil** pour la station **CRETEIL-POINTE DU LAC** dans le champ **Ville**
- ⇒ ajouter **Paris** pour la station **CLUNY SORBONNE** dans le champ **Ville** et **05** dans le champ **Arrondissement**

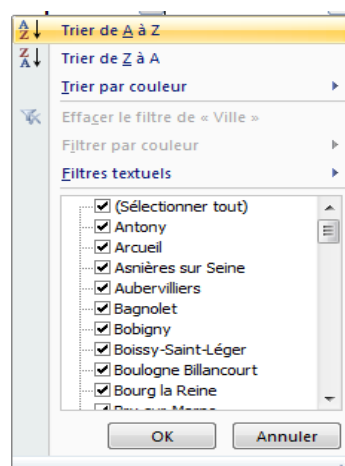
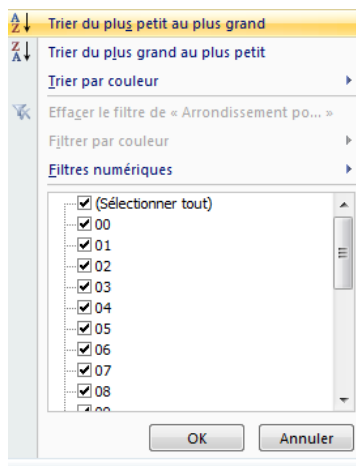
**4 Trier les données par arrondissement et par ville (Paris et la banlieue)**

⇒ Utiliser le menu **Données/Filtrer**

Afin d'obtenir les flèches

Rang	Réseau	Station	Trafic	Ligne de correspondance				Ville	Arrondissement pour Paris
1	Métro	GARE DU NORD	48 146 629	4	5			Paris	10
2	Métro	SAINT-JAZARE	46 790 941	3	9	12	13	14	08

⇒ Puis trier par **arrondissement** et par **ville**



# Initiation opendata, datavisualisation et liste de données

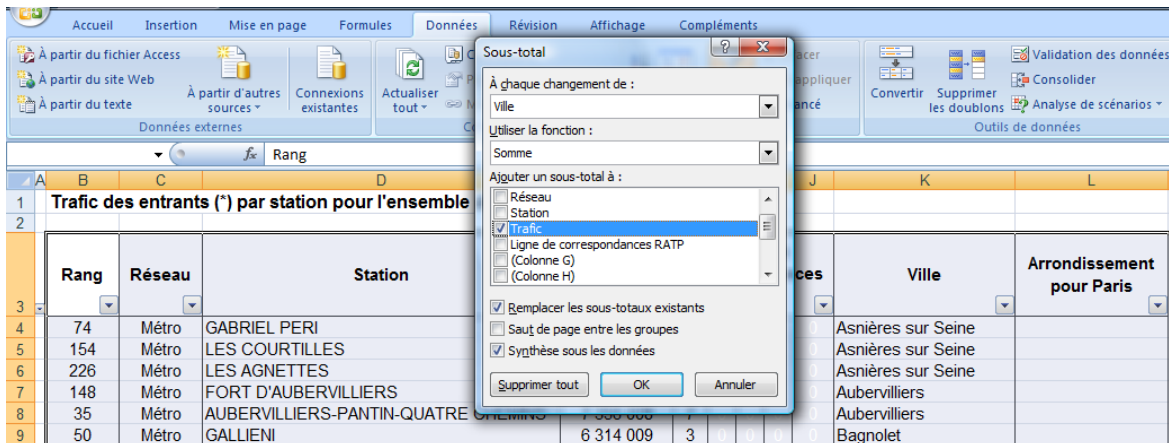
## «La fréquentation du métro parisien»

### 5 Afficher les **sous-totaux** par **ville** et par **arrondissement**

⇒ Placer le curseur dans la base de données

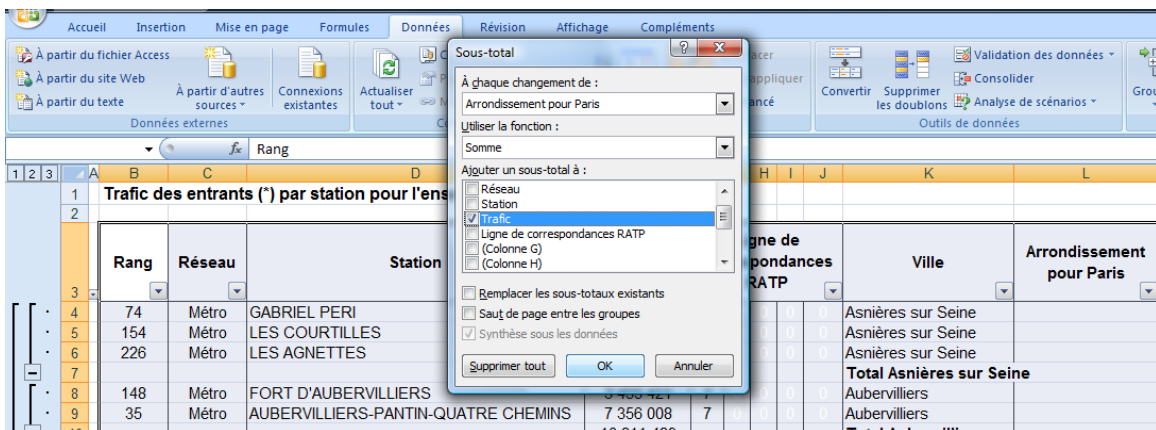
⇒ Sélectionner **Données/Sous-total** à chaque changement de **Ville**

Utiliser la **fonction Somme** et ajouter un sous-total à **Trafic**  
Et valider avec **OK**



⇒ Puis sélectionner **Données/Sous-total** à chaque changement d'**arrondissement**

Attention de décocher **Remplacer les sous-totaux existants** pour maintenir les sous-totaux relatifs aux villes



### 6 Insérer une colonne M avec en en-tête l'intitulé : **% de fréquentation par ville et par ardt** Déterminer la **part relative (en %)** de chaque arrondissement et ville en fonction du trafic total

# Initiation opendata, datavisualisation et liste de données

## «La fréquentation du métro parisien»

- 7 Puis faire apparaître uniquement les sous-totaux en cliquant sur les signes – situés à gauche de la fenêtre de façon à obtenir :

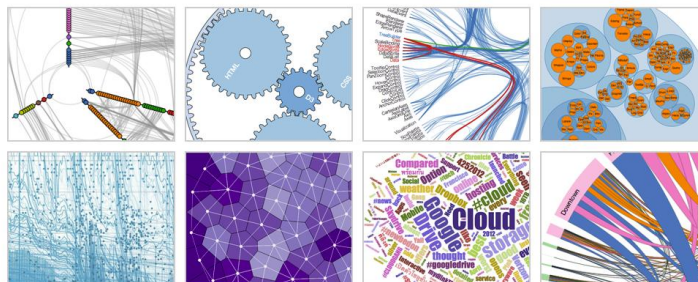
Rang	Réseau	Station	Trafic	Ligne de correspondances RATP	Ville	Arrondissement pour Paris	% de fréquentation par ville et par ardt
7			11 122 613		Total Asnières sur Seine	00	0.80%
10			10 811 429		Total Aubervilliers	00	0.77%
12			6 314 009		Total Bagnolet	00	0.45%
14			6 722 182		Total Bobigny	00	0.48%
20			19 460 129		Total Boulogne Billancourt	00	1.39%
23			5 583 162		Total Charenton	00	0.40%
25			6 540 563		Total Chatillon	00	0.47%
27			6 892 451		Total Clichy	00	0.49%
32			10 481 318		Total Créteil	00	0.75%
35			7 579 032		Total Issy Les Moulineaux	00	0.54%
38			4 390 660		Total Ivry	00	0.31%
40			4 492 364		Total Kremlin-Bicêtre	00	0.32%
42			5 707 118		Total La Courneuve	00	0.41%
44			4 602 681		Total Les Lilas	00	0.33%
48			12 028 697		Total Levallois-Perret	00	0.86%
52			6 870 940		Total Maisons -Alfort	00	0.49%
55			5 758 697		Total Malakoff	00	0.41%
59			15 596 239		Total Montreuil	00	1.12%
62			12 224 490		Total Neuilly Sur Seine	00	0.88%
66			11 101 582		Total Pantin	00	0.79%
74			49 915 457		Total 01	07	3.57%
79			13 716 326		Total 02	08	0.98%
86			20 872 451		Total 03	18	1.49%
93			26 206 239		Total 04	24	1.88%
99			20 195 948		Total 05	25	1.45%
109			28 572 200		Total 06	54	2.05%
121			32 621 655		Total 07	77	2.34%
137			116 195 216		Total 08	120	8.32%
151			69 062 991		Total 09	117	4.95%
162			104 522 353		Total 10	100	7.48%
179			62 666 986		Total 11	176	4.49%
199			108 918 672		Total 12	228	7.80%
218			82 736 076		Total 13	234	5.92%

### Initialisation à la datavisualisation

Jusqu'à présent, pour visualiser les données en % un graphique circulaire ou en bâtons était réalisé, aujourd'hui des outils de visualisation des données comme **D3** à l'url <http://d3js.org/> (utilisé par la presse tels le NYTimes, le Guardian...) permettent de faire des cartographies dynamiques sous forme de pages web.

Overview Examples Documentation Source

## Data-Driven Documents



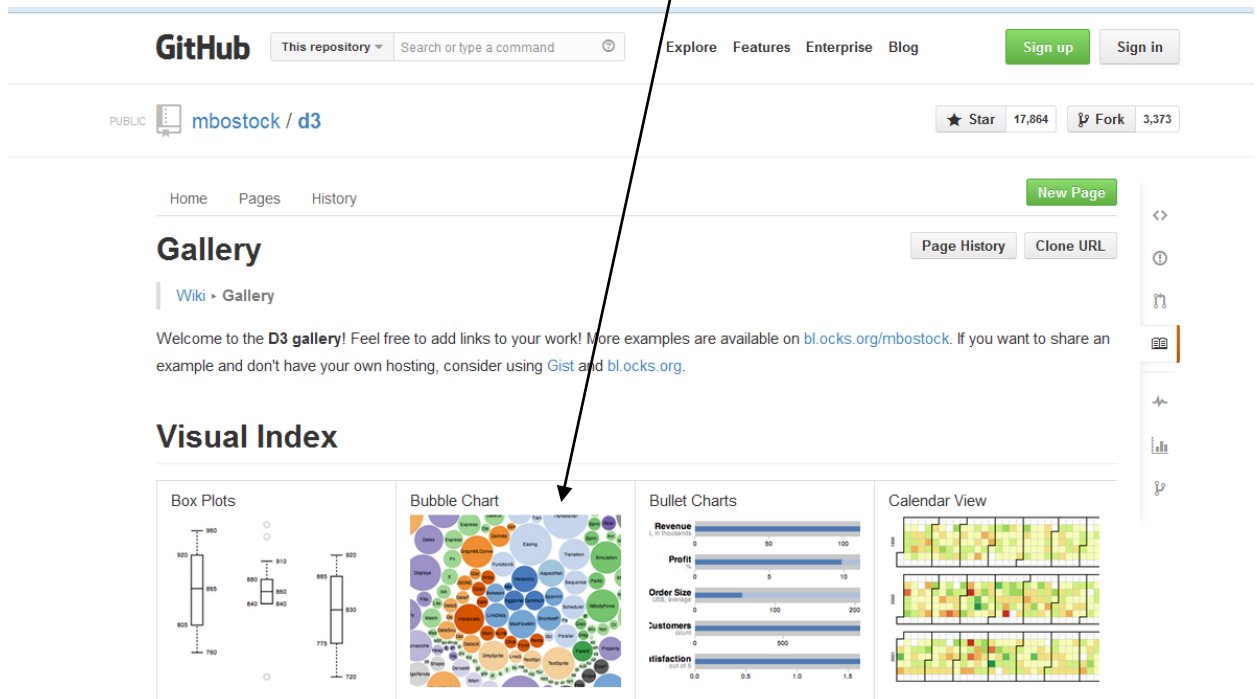
D3.js is a JavaScript library for manipulating documents based on data. D3 helps you bring data to life using HTML, SVG and CSS. D3's emphasis on web standards gives you the full capabilities of modern browsers without tying yourself to a proprietary framework, combining powerful visualization components and a data-driven approach to DOM manipulation.

Download the latest version here:

- d3.v3.zip

# Initiation opendata, datavisualisation et liste de données «La fréquentation du métro parisien»

- 8 Télécharger [d3.v3.zip](#) sur le site <http://d3js.org/>
- 9 Décompresser le fichier zip qui correspond à la librairie nécessaire à la réalisation de la visualisation
- 10 Puis réaliser la visualisation **Bubble Chart** dont un exemple est présent sur le site <https://github.com/mbostock/d3/wiki/Gallery>



- ⇒ Utiliser à cette fin les fichiers à télécharger sur le réseau ou sur <http://slenader.free.fr>
- **opendata.json**
  - **opendata.html**

Le fichier **json** est à compléter avec les données **% de fréquentation par ville et par ardt** calculées avec le tableur

- ⇒ saisir la donnée dans le champ **size** en fonction de l'arrondissement ou de la ville  
**(pour être visible toutes les données sont multipliées par 1000)**  
**3,57 % est saisi pour 3570**

```
1 {
2   "name": "opendata",
3   "children": [
4
5     {
6       "name": "metro RATP",
7       "children": [
8
9
10
11         {
12           "name": "Paris arrondissements"
13           "children": [
14             {"name": "01", "size": 3570},
15             {"name": "02", "size": },
16             {"name": "03", "size": },
17             {"name": "04", "size": },
18             {"name": "05", "size": },
19             {"name": "06", "size": },
20             {"name": "07", "size": },
```

**Attention de ne pas supprimer des virgules dans le fichier json**



# Initiation opendata, datavisualisation et liste de données «La fréquentation du métro parisien»

- ⇒ le fichier **opendata.html** est du **HTML 5**, il fait appel à la **librairie D3**
- ⇒ Il contient également un **script en javascript** qui récupère les données contenues dans le fichier **opendata.json**

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4
5 <title>Opendata</title>
6
7 <link rel="stylesheet" type="text/css" href="" />
8 <script src="d3.v3.min.js" charset="utf-8"></script>
9 </head>
10 <body>
11 <script>
12 var diameter = 960,
13     format = d3.format(",d"),
14     color = d3.scale.category20c();
15
16 var bubble = d3.layout.pack()
17     .sort(null)
18     .size([diameter, diameter])
19     .padding(1.5);
20
21 var svg = d3.select("body").append("svg")
22     .attr("width", diameter)
23     .attr("height", diameter)
24     .attr("class", "bubble");
25
26 d3.json("opendata.json", function(error, root) {
27     var node = svg.selectAll(".node")
28         .data(bubble.nodes(classes(root)))
29         .filter(function(d) { return !d.children; })
30         .enter().append("g")
31         .attr("class", "node")
32         .attr("transform", function(d) { return "translate(" + d.x + "," + d.y + ")"; });
33
34     node.append("title")
35         .text(function(d) { return d.className + ": " + format(d.value); });
36
37     node.append("circle")
38         .attr("r", function(d) { return d.r; })
39         .style("fill", function(d) { return color(d.packageName); });
40
41     node.append("text")
42         .attr("dy", ".3em")
43         .style("text-anchor", "middle")
44         .text(function(d) { return d.className.substring(0, d.r / 3); });
45 });
46
47 // Returns a flattened hierarchy containing all leaf nodes under the root.
48 function classes(root) {
49     var nodes = root.children ? root.children.map(function(c) { return classes(c); }) : [];
50     return root.className ? nodes.concat([root]) : nodes;
51 }
```

- 11 **Expliquer les raisons** qui expliquent la fréquentation plus élevée du métro parisien dans certains arrondissements et villes de banlieue

## Quelques liens

### **Data visualisation :**

- <http://www.informationisbeautiful.net/>
- <http://datavisualization.ch/>
- <http://www.datavisualization.fr/>
- <http://www.excelcharts.com/blog/>
- <http://flowingdata.com/>

### **pour les développeurs :**

- <http://www.highcharts.com/>
- <http://d3js.org/>
- <https://developers.google.com/maps/?hl=fr>