

# TP - Créer des graphiques avec Hightcharts

---

## 1. OBJECTIF

---

Hightcharts est une librairie javascript pour créer des vues graphiques sur une page web. Son utilisation est gratuite pour des projets personnels et payantes pour un usage commercial. La création des graphiques est paramétrable à souhait avec un objet représenté au format JSON. Cette bibliothèque permet de réaliser tous types de graphiques (pie chart, bar graph, bubble graph, timeline, ...) et possède un module d'export intégré (JPEG, PDF, SVG, PNG).

- Représentation de données sous formes de vues graphiques.
- Les données sous la forme d'objet JSON
- Les requêtes AJAX avec JQuery
- La construction en langage PHP d'objets de données au format JSON.

## 2. CONDITIONS DE RÉALISATION

---

**Travail individuel** sur PC avec NetBeans

Un serveur de base de données est accessible avec phpMyAdmin:

hostname	<b>172.18.58.5</b>
username	<b>snir</b>
password	<b>snir</b>
basename	<b>prixEssence</b>

## 3. RESSOURCES

---

La documentation complète de Hightcharts est disponible  
<https://www.highcharts.com/docs>

le code complet de Hightcharts est disponible sur github  
<https://github.com/highcharts/highcharts>

## 4. PRÉSENTATION

Vous allez représenter sous forme de graphiques des données sauvegardées dans une base de données. Pour ce faire vous utiliserez la bibliothèque Highcharts.

Nous procéderons par étapes successives :

- avec des données situées dans une variable js.
- avec des données situées dans un fichier JSON.
- avec des données au format JSON obtenues en AJAX (méthode GET)

### 4.1. Représentation des données sous forme de camembert

Dans un premier temps nous allons représenter des données contenus dans un fichier data.js

Créez un fichier index.html comme ceci :

```
2  <!--
3  To change this license header, choose License Headers in Project Properties.
4  To change this template file, choose Tools | Templates
5  and open the template in the editor.
6  -->
7  <html>
8      <head>
9          <title>TODO supply a title</title>
10         <meta charset="UTF-8">
11         <meta name="viewport" content="width=device-width, initial-scale=1.0">
12         <!-- Libs -->
13         <script src="//ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
14         <script src="//code.highcharts.com/highcharts.js"></script>
15         <script src="//code.highcharts.com/modules/exporting.js"></script>
16
17         <script type='text/javascript' src='data.js'></script>
18         <script type='text/javascript' src='main.js'></script>
19     </head>
20     <body>
21         <div class="wrapper">
22             <div id="pieChart"></div>
23         </div>
24     </body>
25 </html>
```

le fichier html charge les fichiers suivants :

- JQuery (via googleapi.com)
- Highcharts (via highcharts.com)
- data.js (Le fichier contenant les données à afficher)
- Main.js (le fichier qui va contenir votre code javascript pour lancer highcharts)

Créez un fichier data.js contenant une variable javascript de type tableau de valeurs:

```
var donnees = [
  ["ILE DE FRANCE", 2232943587],
  ["CHAMPAGNE ARDENNE", 208049172],
  ["PICARDIE", 291647915],
  ["HAUTE NORMANDIE", 329969182],
  ["CENTRE", 426281434],
  ["NORD PAS DE CALAIS", 784487750],
  ["LORRAINE", 462959889],
  ["ALSACE", 313990716],
  ["FRANCHE COMTE", 184551882],
  ["BASSE NORMANDIE", 251148990],
  ["PAYS DE LA LOIRE", 528196589],
  ["BRETAGNE", 475894603],
  ["LIMOUSIN", 142804187],
  ["AUVERGNE", 221209988],
  ["POITOU CHARENTES", 255393306],
  ["AQUITAINE", 637123470],
  ["MIDI PYRENEES", 617798467],
  ["BOURGOGNE", 277143087],
  ["RHONE ALPES", 1056935822],
  ["LANGUEDOC ROUSSILLON", 629009598],
  ["PROVENCE ALPES COTE D'AZUR", 1198241479],
  ["CORSE", 77128456]
];
```

Puis créer un script main.js

```
$(document).ready(function() {
  $.afficherGraphique(couleurs, donnees);
});

var couleurs = [
  "#1f77b4", "#aec7e8", "#ff7f0e", "#ffbb78", "#2ca02c", "#98df8a", "#d62728", "#ff9896",
  "#9467bd", "#c5b0d5", "#8c564b", "#c49c94", "#e377c2", "#f7b6d2", "#7f7f7f", "#c7c7c7",
  "#bcbd22", "#dbdb8d", "#17becf", "#9edae5"
];

$.afficherGraphique = function(couleurs, data) {
  var chart = new Highcharts.Chart({
    chart: {
      renderTo: 'pieChart',
      width: 598,
      height: 450,
      marginLeft: 25,
      marginRight: 25
    },
    colors: couleurs,
    title: {
```

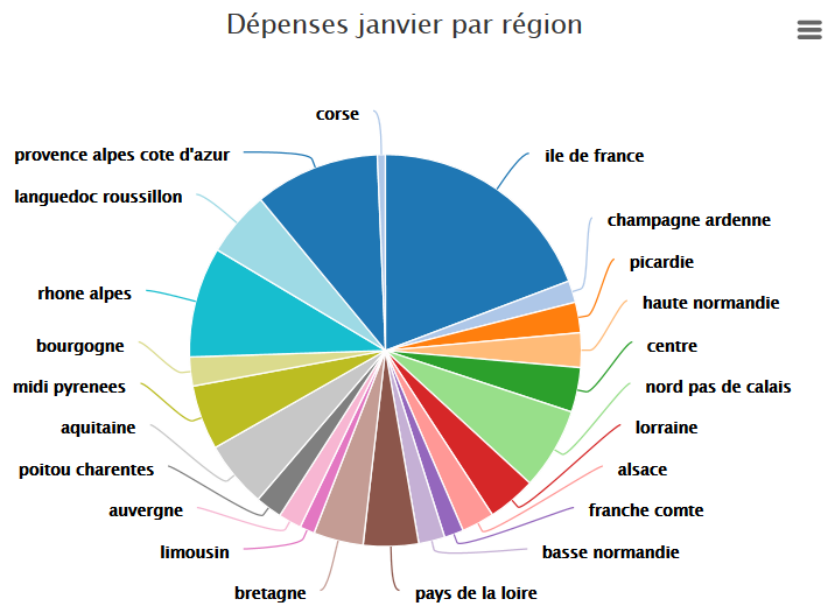
```
    text: "Dépenses par région",
    margin: 10
  },
  tooltip: {
    formatter: function() {
      return "Dépense : " + this.y + " €" ;
    }
  },
  plotOptions: {
    pie: {
      allowPointSelect: true,
      cursor: 'pointer',
      dataLabels: {
        enabled: true,
        formatter: function() {
          return "" + this.point.name.toLowerCase() + "";
        }
      }
    }
  },
  series: {
    dataLabels: {
      enabled: true,
      color: 'black',
      fontSize: 3
    }
  },
  series: [{
    type: 'pie',
    data: data
  }]
});
};
```

Dans ce script , la fonction \$.afficherGraphique(couleurs, donnees); affiche un graphique sous la forme d'un camembert.

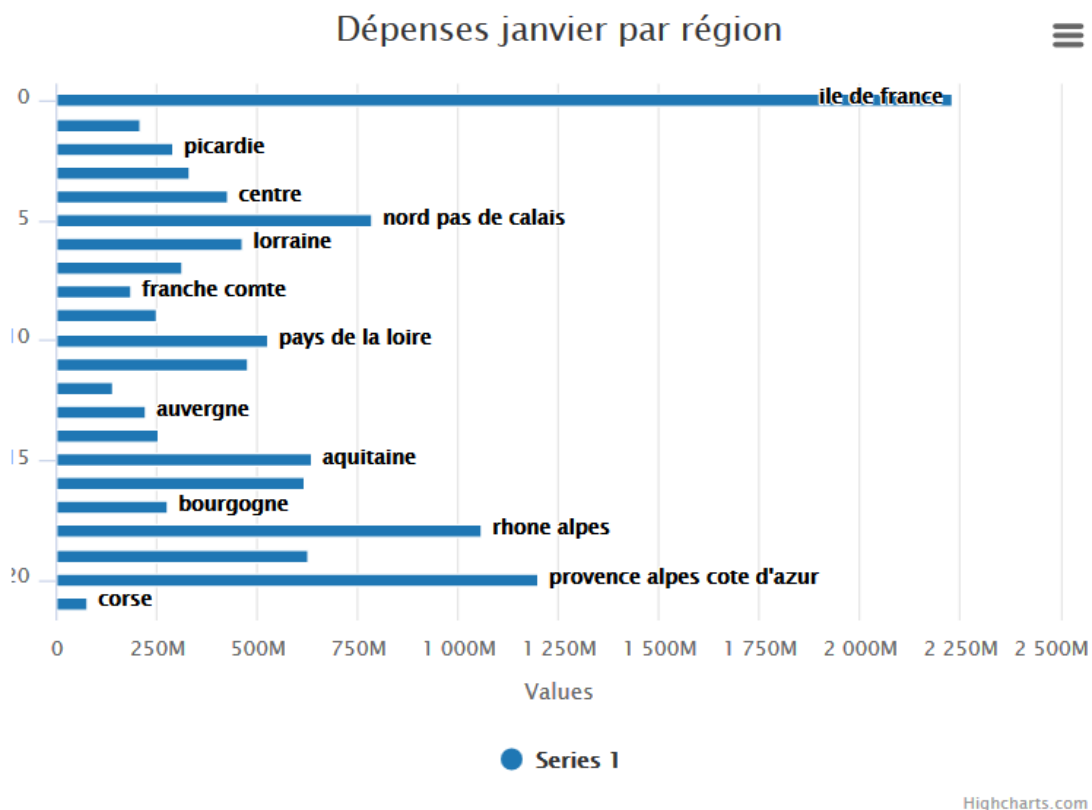
La variable array **couleurs** représente les couleurs que l'on va utiliser pour le graphique.

## DÉVELOPPEMENT

la fonction `$.afficherGraphique` est appelé quand le document est ready en lui passant 2 paramètres, la variable `data` (représente le dataset contenu dans `data.js`) et la variable `array couleur`.



Écrire une deuxième fonction pour afficher les mêmes données sous forme de graphique de type barres.



## 4.2. Représentation des données sous forme de courbe

Nous allons maintenant représenter des données contenu dans un fichier **prixEssence.json** au format json.

Créer un nouveau projet

```

1  <!DOCTYPE html>
2
3  <html>
4  <head>
5      <title>Prix de l'essence</title>
6      <meta charset="UTF-8">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <!-- Libs -->
9      <script src="//ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
10     <script src="//code.highcharts.com/highcharts.js"></script>
11     <script src="//code.highcharts.com/modules/exporting.js"></script>
12
13     <script type='text/javascript' src='main.js'></script>
14 </head>
15 <body>
16     <div>Graphique avec Hightcharts</div>
17     <div>
18         <div id="essenceChart" style="padding: 50px;"></div>
19
20     </div>
21 </body>
22 </html>
23

```

le script main.js

le script ci dessous construit un objet javascript contenant les options qui vont servir à générer le graphique, il suffit donc de déclarer un objet **options** et de lui attribuer les options de bases :

```

var chart;
var couleurs = ["#1f77b4", "#aec7e8", "#ff7f0e", "#ffbb78", "#2ca02c"];
var options = {};

options.chart = {
    renderTo: 'essenceChart',
    height: 500,
    marginTop: 70,
    marginLeft: 100,
    marginRight: 100,
    backgroundColor: '#F9F9F9',
    type: 'line'
};

options.credits = {
    enabled: false
};

options.colors = couleurs;

options.title = {
    text: "Évolution du prix de l'essence et du baryl de pétrole",

```

```
margin: 10
};

options.tooltip = {
  formatter: function() {
    return "Année : " + this.x + " Prix : " + this.y + " €" ;
  }
};

// axes des Y
// axes des X
};
```

### 4.3. Créer 2 axes Y

Créer deux axes Y va permettre de comparer 2 valeurs d'ordre différent. Dans notre exemple, on va comparer le prix de l'essence en euro autour de 1,5€ avec le prix du baril de pétrole en \$ autour de 50\$.

```
options.yAxis = [
  {
    title : {
      text: "Gazoil & Super 95 98 (€)"
    },
    labels: {
      formatter: function() {
        return this.value + ' €';
      },
      style: {
        color: '#000'
      }
    }
  },
  {
    title: {
      text: "Baryl brent ($US)"
    },
    labels: {
      formatter: function() {
        return this.value + ' $';
      },
      style: {
        color: '#8C564B'
      }
    },
    opposite: true
  }
];
```

## 4.4. Mettre en place l'axe des X

```
options.xAxis = {
  categories: [],
  crosshair: true,
  labels: {
    rotation: -45,
    y: 20
  }
}
```

On peut remarquer dans le tableau `categorie[]` reste pour le moment vide.

## 4.5. Insérer plusieurs dataset dans un graphique

```
options.series = [
  {
    name: 'Gazoil (€)',
    data: []
  },
  {
    name: 'Super 95 (€)',
    data: []
  },
  {
    name: 'Super 98 (€)',
    data: []
  },
  {
    name: 'Baryl de pétrole ($US)',
    yAxis: 1,
    data: []
  }
];
```

On peut remarquer dans le tableau séries que les variables data sont vides. Il est donc nécessaire de les charger, c'est le rôle de la fonction `getJSON`.

## 4.6. Chargement des données dans les variables data[]

Pour charger les données en ajax, on utilise la méthode `getJSON` de JQuery pour lire le fichier de données **prixEssence.json**. Il suffit ensuite de manipuler les données dans la fonction de callback pour aller nourrir les 'array' de data de chacune des séries. On en profite pour créer les catégories (les labels) qui apparaîtront sur l'axe des X.

```
$.getJSON('prixEssence.json', function(prixEssence) {

  $.each(prixEssence, function(annee,prix){
    options.series[0].data.push(prix.gazoil);
    options.series[1].data.push(prix.super95);
    options.series[2].data.push(prix.super98);
    options.series[3].data.push(prix.brent);
    options.xAxis.categories.push(annee);

  });
  chart = new Highcharts.Chart(options);
});
```



#### 4.7. La base de données

Les informations concernant la base de données sont les suivantes:

Nom d'hôte	<b>172.18.58.5</b>
Nom de la base	<b>snirEssence</b>
utilisateur	<b>snir</b>
Mot de passe	<b>snir</b>

La base de données "**snirEssence**" contient une table **prixEssence** avec des données. Ces données correspondent aux prix moyens annuels des carburants pour l'automobile et du baryl de pétrole depuis 1995.

#### 4.8. Création du fichier JSON en php

**Le script PHP va requêter la base de données** avec la requêtes SQL appropriée, puis créer un objet JSON.

La fonction **json\_encode** convertit un objet de données en chaîne au format JSON.

Lien vers la documentation json-encode() :

<http://www.lephpfacile.com/manuel-php/function.json-encode.php>

```
<?php

require_once("config.inc.php");

function connexionBD() {
    try
    {
        $bdd = new PDO('mysql:host=' . SERVEURBD . ';dbname=' . NOMDELABASE,
        LOGIN, MOTDEPASSE);
    }
    catch (Exception $ex)
    {
        die('<br />Pb connexion serveur BD : ' . $ex->getMessage());
    }
    return $bdd;
}

function obtenirValeur() {

    $bdd = connexionBD();

    $requete = $bdd->prepare("SELECT * FROM `prixEssence`");
    $requete->execute() or die (print_r($requete->errorInfo()));

    $valeurs = array();
    $prix = array();

    while ($ligne = $requete->fetch()) {
```

```

        $prix["gazoil"] = floatval($ligne['gazoil']);
        $prix["super95"] = floatval($ligne['super95']);
        $prix["super98"] = floatval($ligne['super98']);
        $prix["brent"] = floatval($ligne['brent']);
        $valeurs[$ligne['annee']] = $prix;

    }
    echo json_encode($valeurs, JSON_FORCE_OBJECT);
}

header("Access-Control-Allow-Origin: *");
header('Content-type: application/json');
obtenirValeur();

?>

```

Ecrire un nouveau script php permettant d'obtenir la chaîne JSON suivante :

```

{
  "pointStart": 1995,
  "series": [{
    "name": "gazoil (L)",
    "tooltip": {
      "valueSuffix": " \u20ac"
    },
    "data": [0.5869, 0.6533, 0.6764, 0.6424, 0.69, 0.8468, 0.798, 0.7724, 0.7935,
0.8847, 1.0268, 1.0775, 1.0949, 1.2671, 1.0024, 1.1467, 1.3354, 1.4317, 1.3502, 1.2856, 1.1494,
1.1055]
  }, {
    "name": "super95 (L)",
    "tooltip": {
      "valueSuffix": " \u20ac"
    },
    "data": [0.857, 0.9091, 0.9491, 0.9186, 0.955, 1.0917, 1.0329, 1.0146, 1.0163,
1.0603, 1.1659, 1.2368, 1.2765, 1.3538, 1.2092, 1.3464, 1.4995, 1.6518, 1.5367, 1.4846, 1.3531,
1.3039]
  }, {
    "name": "super98 (L)",
    "tooltip": {
      "valueSuffix": " \u20ac"
    },
    "data": [0.8631, 0.9171, 0.9503, 0.9278, 0.9636, 1.1093, 1.057, 1.0365, 1.0366,
1.0827, 1.2053, 1.2743, 1.3086, 1.3929, 1.2431, 1.3821, 1.5374, 1.701, 1.5943, 1.5448, 1.415,
1.3624]
  }, {
    "name": "brent (Baryl)",
    "tooltip": {
      "valueSuffix": " $"
    },
    "data": [17.04, 20.65, 19.11, 12.78, 17.92, 28.52, 24.44, 24.95, 28.89, 38.24, 54.41,
65.14, 72.45, 96.99, 61.48, 79.44, 111.22, 118.83, 108.63, 99.02, 52.35, 43.54],
    "yAxis": 1
  }
]}

```

Écrire le script permettant d'afficher le graphique des données à partir de ce fichier JSON.