

# TEMPLATE BAS - GRAPHS PIE & BAR

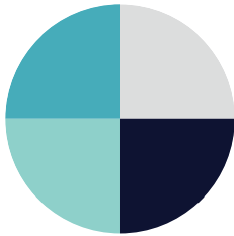
DIFFICULTÉS ★★★★★

PRÉPARATION ★★★★★

POSITION DU TEMPLATE: BOTTOM

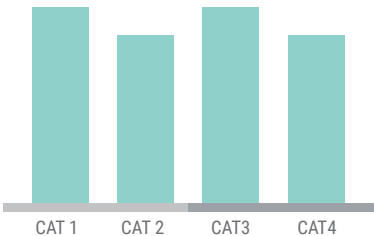
NOM DE L'ENTITÉ GÉO.

Catégorie 1 Catégorie 2 Catégorie 3 Catégorie 4



Lorem ipsum iso venat sierd ardu que  
pour l'entité sélectionnée

Personnes



Lorem ipsum iso venat sierd ardu que  
pour l'entité sélectionnée

## IMPORTANT !

Pour que le template fonctionne, pensez à paramétrer votre fichier de configuration config.xml comme ceci :

```
<layer id="xx" name="xx"
  infoformat="application/vnd.ogc.gml"
  ... >
  <template url="montemplate.mst"></template>
</layer>
```

## J'ai besoin de

### NOM DE L'ENTITÉ GÉOGRAPHIQUE

#### Titre (texte)

LIEN VERS MON TITRE ☒

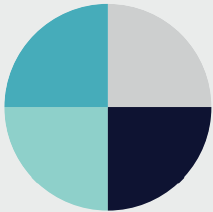
Le titre est **propre** à chaque entité géographique, quel est le champs dans ma donnée ?

**A1.** {{nom\_géographique}}

#### OPTIONS

POUR CHAQUE ÉLÉMENT, IL EST NÉCESSAIRE DE CHOISIR UNE OPTION PARMI CELLES PROPOSÉES

Catégorie 1 Catégorie 2 Catégorie 3 Catégorie 4



Lorem ipsum iso venat sierd ardu que  
pour l'entité sélectionnée

### Diagramme circulaire

LIEN VERS MES DONNÉES ☒

L'identifiant ID est propre à chaque entité comme le code INSEE pour une commune, quel est le champs dans ma donnée ?

**B1.** {{code\_geo}}

Quelles sont les données à représenter en pie au sein du graphique ?

**C1.** {{champs\_data1}}, {{champs\_data2}}

CONFIGURATION DU GRAPHIQUE ☒

Quelles sont les libellés des variables représentées dans le graphique ?

**D1.** cat1, cat2

Quelle est la légende du graphique (unité de représentation) ?

**E1.** Unité de représentation

Le titre du graphique est :

**F1.** Lorem ipsum iso venat

Personnes



Lorem ipsum iso venat sierd ardu que  
pour l'entité sélectionnée

### Diagramme en bar

LIEN VERS MES DONNÉES ☒

L'identifiant ID est propre à chaque entité comme le code INSEE pour une commune, quel est le champs dans ma donnée ?

**G1.** {{code\_geo}}

Quelles sont les données à représenter en bar au sein du graphique ?

**H1.** {{champs\_data1}}, {{champs\_data2}}, {{champs\_data3}}, {{champs\_data4}}

CONFIGURATION DU GRAPHIQUE ☒

Quelles sont les libellés des variables représentées dans le graphique en X ?

**I1.** cat1, cat2, cat3, cat4

Quelle est la légende du graphique (unité de représentation) ?

**J1.** Unité de représentation

Le titre du graphique est :

**K1.** Lorem ipsum iso venat

## Je configure mon template

• Ouvrir le fichier `template.bottomGraphs2.mst` à l'aide d'un éditeur de texte (Notepad++)

• À la ligne 5, insérez l'option **A1** pour configurer le titre : `<h3 class="title-feature ">A1</h3>`

-- CONFIGURATION DU DIAGRAMME CIRCULAIRE

• À la ligne 11, insérez l'option **B1** pour configurer l'identifiant du graphique et injecter le code géographique de la donnée :

```
<canvas class="chart" id="myChartPie1_B1" data-codegeo="B1"></canvas>
```

--> L'option B1 étant présente dans plusieurs partie du code, il est préférable d'effectuer un RECHERCHER/REPLACER en remplaçant code\_geo par B1

• À la ligne 11, insérez tous les champs définis dans l'option **C1** pour ajouter les données au graphique :

```
<canvas data-mychartPie1-1="{{C1_champsdata1}}" data-mychartPie1-2="{{C1_champsdata2}}"></canvas>
```

--> Ajoutez autant de data-mychartPie1-X que vos variables [CAT1, CAT2, CAT3, CAT4...] qui composent le graphique

• À la ligne 14, insérez l'option **F1** pour configurer le titre du graphique : `<div class="title-chart">F1</div>`

• À la ligne 159, insérez les libellés des variables définis dans l'option **D1** :

```
var data = {myChartPie1_attr: {labels:["D1 CAT1","D1 CAT2"]}}; --> Ajoutez autant de libellés entre "" que vos variables [CAT1, CAT2...] qui composent le graphique
```

• À la ligne 159, insérez dans les datasets les données des variables définies dans l'option **C1** :

```
var data = {myChartPie1_attr: data: {datasets:[[parseFloat($("#myChartPie1_{{code_geo}}").attr("data-mychartPie1-1")), parseFloat($("#myChartPie1_{{code_geo}}").attr("data-mychartPie1-2"))]]}}};
```

--> Ajoutez autant de parseFloat(\$("#myChartPie1\_{{code\_geo}}").attr("data-mychartPie1-X")) séparé par une virgule que vos variables [CAT1, CAT2, CAT3, CAT4...] qui composent le graphique

• À la ligne 174, insérez l'option **E1** pour configurer la légende du graphique : `label: "E1",`

-- CONFIGURATION DU DIAGRAMME EN BAR

• À la ligne 26, insérez l'option **G1** pour configurer l'identifiant du graphique et injecter le code géographique de la donnée :

```
<canvas class="chart" id="myChartBar1_G1" data-codegeo="G1"></canvas>
```

--> L'option G1 étant présente dans plusieurs partie du code, il est préférable d'effectuer un RECHERCHER/REPLACER en remplaçant code\_geo par G1

• À la ligne 26, insérez tous les champs définis dans l'option **H1** pour ajouter les données au graphique :

```
<canvas data-mychartBar1-1="{{H1_champsdata1}}" data-mychartBar1-2="{{H1_champsdata2}}" data-mychartBar1-3="{{H1_champsdata3}}" data-mychartBar1-4="{{H1_champsdata4}}"></canvas>
```

--> Ajoutez autant de data-mychartBar1-X que vos variables [CAT1, CAT2, CAT3, CAT4...] qui composent le graphique

• À la ligne 29, insérez l'option **K1** pour configurer le titre du graphique : `<div class="title-chart">K1</div>`

• À la ligne 220, insérez les libellés des variables définis dans l'option **I1** :

```
var data = {myChartBar1_attr: {labels:["I1 CAT1","I1 CAT2","I1 CAT3","I1 CAT4"]}};
```

--> Ajoutez autant de libellés entre "" que vos variables [CAT1, CAT2, CAT3, CAT4...] qui composent le graphique

• À la ligne 107, insérez dans les datasets les données des variables définies dans l'option **H1** :

```
var data = {myChartPie1_attr: data: {datasets:[[parseFloat($("#myChartBar1_{{code_geo}}").attr("data-mychartBar1-1")),parseFloat($("#myChartBar1_{{code_geo}}").attr("data-mychartBar1-2")),parseFloat($("#myChartBar1_{{code_geo}}").attr("data-mychartBar1-3")),parseFloat($("#myChartBar1_{{code_geo}}").attr("data-mychartBar1-4"))]]}}};
```

--> Ajoutez autant de parseFloat(\$("#myChartBar1\_{{code\_geo}}").attr("data-mychartBar-X")) séparé par une virgule que vos variables [CAT1, CAT2, CAT3, CAT4...] qui composent le graphique

• À la ligne 122, insérez l'option **J1** pour configurer la légende du graphique : `label: "J1",`

-- PERSONNALISATION DES COULEURS

• Pour personnaliser la couleur du template (titre), insérez le code #HEX de votre couleur à la ligne 46 :

```
:root {
  --mycolor: #1D1B35;
}
```

• Pour personnaliser la couleur du graphique en bar, insérez le code RGBA de votre couleur aux lignes 123, 124, 127, 128 :

```
datasets: [{
  backgroundColor: "rgba(130,202,200,0.9)",
  borderColor: "rgba(130,202,200,0.9)",
  hoverBackgroundColor: "rgba(130,202,200,1)",
  hoverBorderColor: "rgba(130,202,200,1)",
}]
```

• Pour personnaliser la couleur du graphique circulaire, insérez les codes RGBA de vos couleurs à partir de la ligne 177 :

```
datasets: [{
  backgroundColor: [
    "rgba(130,202,200,0.9)", //COULEUR DE LA CATÉGORIE 1 --> Ajoutez autant de couleurs que vos variables [CAT1, CAT2, CAT3, CAT4...]
    "rgba(29,28,51,0.9)" //COULEUR DE LA CATÉGORIE 2
  ],
  hoverBackgroundColor: [
    "rgba(130,202,200,1)", //COULEUR DE LA CATÉGORIE 1 AU SURVOL
    "rgba(29,28,51,1)" //COULEUR DE LA CATÉGORIE 2 AU SURVOL
  ],
}]
```

LES ÉLÉMENTS DOIVENT ÊTRE PRÉSENTS DANS LA COUCHE GÉOGRAPHIQUE UTILISÉE EN LIEN DANS LE MVIEWER