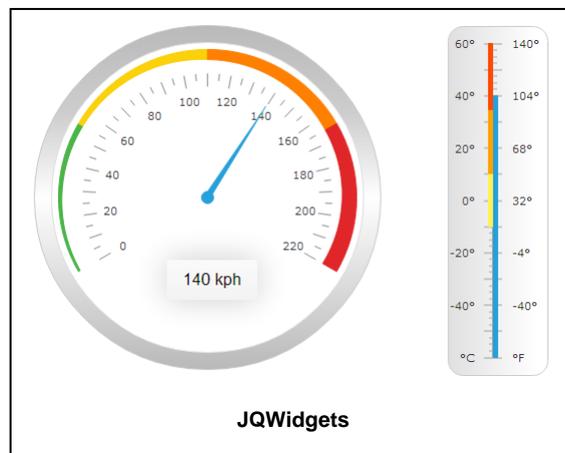


CI4 : Gestion de l'information / Structures matérielles et logicielles associées au traitement de l'information.

Tutoriel « Présentation des données dans une page web »

- TP de découverte.
- Application, mise en œuvre de savoirs/savoir-faire
- Recherche et validation de solutions.
- Evaluation.



CONNAISSANCES ABORDEES

Programmation objet.

PROBLEMATIQUE

Présenter des données dans des widgets et les intégrer à une page web.

CONDITIONS DE DEROULEMENT DE L'ACTIVITE

Phases de travail	Objectifs	Activités
1) Mise en situation	Comprendre le cahier des charges.	Lire la mise en situation.
2) Etude de la problématique	Comprendre la notion d'attribut d'un objet. Paramétrer un objet jauge (widget JQuery) pour afficher, dans une page web, la température mesurée par une station météo.	Modifier les attributs d'un objet logiciel (script JQuery).
3) Synthèse	Faire évoluer le site.	Etudier les modifications nécessaires à l'ajout d'un widget dans une page web.

SIGLES UTILISES

: Durée conseillée	: Important	: Information	: Editeur de texte (voir prof)
--------------------	-------------	---------------	--------------------------------

RESSOURCES DOCUMENTAIRES, LOGICIELS ET MATERIELS

PC + Navigateur (Edge, Chrome, Opéra, Mozilla etc..) + Editeur de texte



Table des matières

1. Mise en situation	3
1.1. Introduction	3
1.2. Lien avec le travail en projet.....	4
1.3. Cahier des charges	4
2. Etude de la problématique	5
2.1. Partie 1 : Modification du Widget JQuery « Gauge »	5
2.1.1. Analyse du code du widget à modifier	5
2.1.2. Préparation du répertoire du site	6
2.1.3. Modification des attributs de l'objet "Gauge"	9
2.1.4. Contrôle de l'objet "Gauge" avec un objet "Button"	11
2.2. Partie 2 : intégration du widget "Gauge" dans un site web.....	12
3. Synthèse	12
Annexe 1 : La documentation du widget "jqxGauge"	14
Annexe 2 : Chemins relatifs et chemins absolus.....	15

1. Mise en situation

1.1. Introduction

Les **widgets** sont particulièrement bien adaptés lorsqu'on souhaite obtenir un **effet visuel** ou effectuer un **traitement sur les données à afficher dans une page Web**.

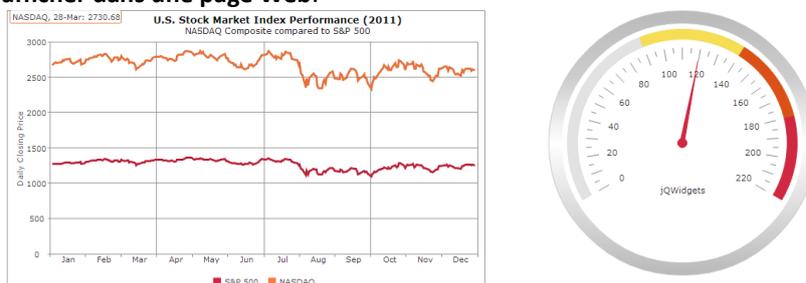


Figure 1 : Exemples de widgets (jQuery)

« En informatique, le mot **widget** recouvre deux notions distinctes en relation avec les **interfaces graphiques**. Il peut alors être considéré comme étant la contraction des termes **window** (fenêtre) et **gadget**.

Il peut désigner :

- un **composant d'interface graphique**, un élément visuel d'une interface graphique (bouton, ascenseur, liste déroulante, etc.) ;
- un **widget interactif**, un petit outil qui permet d'obtenir des informations (météo, actualité, dictionnaire, carte routière, pense-bête – en anglais post-it –, traducteur etc.). » Wikipédia



Des frameworks tels que **jqWidgets** rassemblent un grand nombre d'**outils graphiques** (gauge, chart, grid etc.)

Extrait de la présentation du **framework** sur le site **jqwidgets.com**

« Powerful UI widgets for jQuery developers

jqWidgets delivers everything you need to build rich client UI with JavaScript, HTML5 and jQuery. It contains more than 30 UI widgets and is one of the fastest growing JavaScript UI frameworks on the Web.

Works on PC, Touch and Mobile devices

jqWidgets is ready for a wide range of touch and mobile devices. The framework automatically detects the device type, capabilities and browser version, and adjusts the internal behavior for optimal rendering and best user experience. The framework supports all major desktop and mobile web browsers - Internet Explorer 7.0+, Firefox 2.0+, Safari 3.0+, Opera 9.0+, Google Chrome, IE Mobile, Android, Opera Mobile, Mobile Safari(iPhone, iPad). »

JQWidget est donc un framework **JQuery**.

« **jQuery** est une **bibliothèque JavaScript libre** qui porte sur l'interaction entre JavaScript et HTML, et a pour but de **simplifier** des commandes communes de JavaScript.

La première version date de janvier **2006**.



jQuery se présente comme un unique fichier JavaScript contenant toutes les fonctions de base. Il peut être inclus dans toute page web. » Wikipédia

*Vous avez utilisé JQuery pour créer un **site Web pour mobile**. (TUTO « Conception d'un site web pour mobile »). Seules les notions de balise et d'attribut ont été nécessaires lors de la réalisation de ce site. Pour mettre en œuvre les **widgets**, vous allez aborder la notion d'**objet** et quelques uns des principes de **programmation** qui lui sont associés.*

1.2. Lien avec le travail en projet

Ce TP est un préalable à la préparation de tout projet nécessitant la présentation soignée de données dans une page Web.

Ce document doit vous permettre de revoir et d'approfondir les langages **HTML** (balise **div**) et **CSS** (**sélecteur**, **classe**) abordées dans les précédents TP.

Nous introduirons la notion d'**objet logiciel** (**attribut**, **méthode**, **évènement**).

Vous appliquerez ces notions pour réaliser un **site web**. Celui-ci sera déployé "**en local**" (sur votre PC).

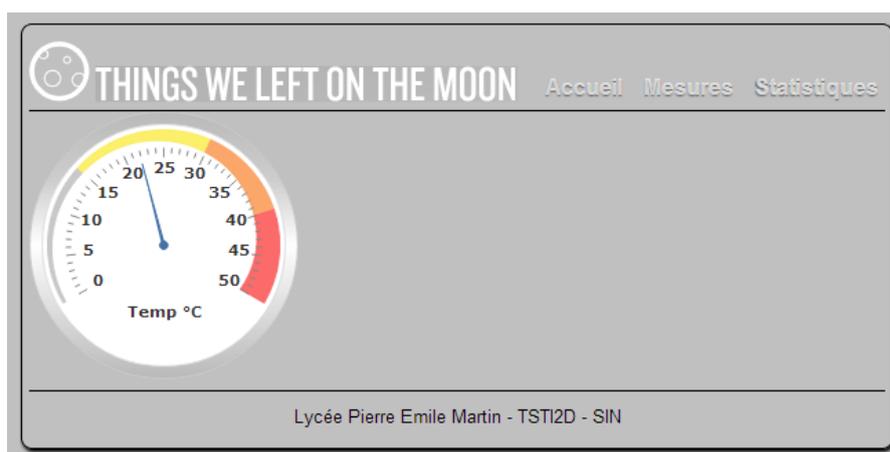
Notre objectif étant de réaliser un site soigné et fonctionnel, les animations et le rendu des pages seront obtenus en faisant appel au **framework** (**jqWidgets**) et à **CSS3**.

Les compétences que vous allez acquérir seront réinvesties dans **vos projets** lors du codage des pages de présentation et des interfaces utilisateurs (**UI**).

1.3. Cahier des charges

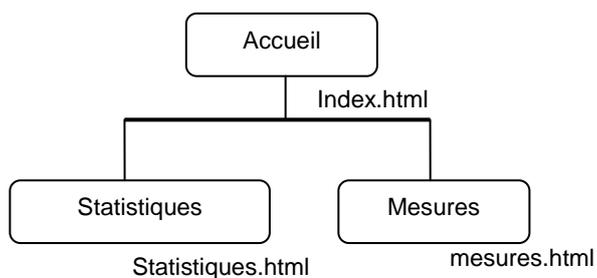
Le site à réaliser est destiné à un écran de PC. Il doit permettre de **présenter l'évolution de grandeurs physiques** dans des **diagrammes** (**charts**) et leur **valeur instantanée** dans des **jauges** (**gauges**).

Exemple d'organisation du site :



Dans ce TP, un seul widget (une jauge) sera intégré à la page "Mesures" du site.

Architecture retenue pour le site



2. Etude de la problématique

[🕒 2h]

Le travail à réaliser est organisé en **deux parties** :

- ✓ **Partie 1** : modifier le script d'un widget JQuery afin d'obtenir un objet capable d'afficher une grandeur physique.
- ✓ **Partie 2** : intégrer le widget dans une page HTML.

Appel prof

Pour télécharger le répertoire du TP

2.1. Partie 1 : Modification du Widget JQuery « Gauge »

Objectifs : créer et organiser le répertoire du projet et adapter le code JQuery du Widget « Gauge » proposé pour qu'il affiche l'information issue d'un capteur de température (copie d'écran ci-contre).

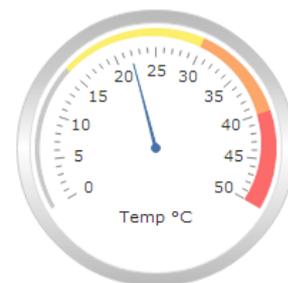


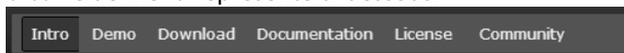
Figure 2 : Jauge à réaliser

2.1.1. Analyse du code du widget à modifier

Les informations nécessaires à la mise en œuvre des widgets JQuery sont accessibles sur le site jqwidgets.com.

Allez sur le site en entrant l'url <http://www.jqwidgets.com> dans un navigateur.

Sélectionnez **Demo** dans la barre de menu représenté ci-dessous.



Sélectionnez **jqxGauge**

Sélectionnez "Gauge Settings" et testez les différentes options proposées dans l'exemple. (Cochez ou décochez les options et observez l'effet obtenu)

Note : si l'accès à internet n'est pas disponible, les démos sont accessibles à partir du fichier **index.htm** situé dans le répertoire **jqwidgets-ver_x.x.x** (x.x.x correspond au numéro de version du framework)

Chemin : <Répertoire du TP>\Ressources\jqwidgets-ver_x.x.x\index.htm



Exemple de jauge !

Comme vous pouvez le remarquer, il est possible d'agir sur les différents **attributs (properties)** de la jauge.

Exemple : l'attribut **label** est associé à l'échelle de la jauge.

Vous allez voir **comment transformer un widget en modifiant ses attributs** dans la suite de ce document.

Sélectionnez **View Source** pour afficher le code source de la page.

Le texte est un peu "dense" ! En effet, il contient le code du widget "Gauge" et le code du formulaire "Options".

Vous allez utiliser un code plus "restreint" dans la suite du TP. Vous constaterez alors que l'utilisation d'un widget est facilitée par le **framework** JQuery.

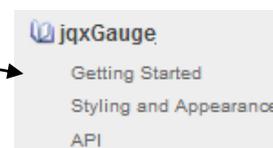


Principe à retenir : pour utiliser un widget JQuery, il faut **copier-coller** son code dans la page du site que l'on développe puis le **modifier** pour l'**adapter** aux besoins de l'application !

Pour avoir des informations sur le **code de la jauge**, sélectionnez **Documentation** dans le menu du site JQWidgets.com.

Sélectionnez **jqxGauge** dans la liste des widgets puis "**Getting Started**"

Remarque : Les noms des widgets ne sont pas classés dans l'ordre alphabétique !



La page devrait ressembler à celle reproduite en **annexe 1 de ce document**.

Le code de l'exemple !

La documentation !

L'en-tête du fichier "**testjauge.html**" **rappelé ci-dessous** contient **les liens** vers les **fichiers indispensables** au fonctionnement de la jauge.

```
<title id='Description'>jqxGauge Basic Demo</title>
<link rel="stylesheet" href="../../jqwidgets/styles/jqx.base.css" type="text/css" />
<script type="text/javascript" src="../../scripts/jquery-x.x.x.min.js"></script>
<script type="text/javascript" src="../../jqwidgets/jqxcore.js"></script>
<script type="text/javascript" src="../../jqwidgets/jqxchart.js"></script>
<script type="text/javascript" src="../../jqwidgets/jqxgauge.js"></script>
<script type="text/javascript" src="../../jqwidgets/jqxbuttons.js"></script>
```

Des liens
et
des chemins "relatifs" !

Les liens vers les fichiers, **jqx.base.css**, **jqxcore.js** etc., placés dans les balises `<script>` et `<link>`, ont l'extension **.css** lorsqu'ils font référence à une **feuille de style** et l'extension **.js** lorsqu'il s'agit de code **javascript**.



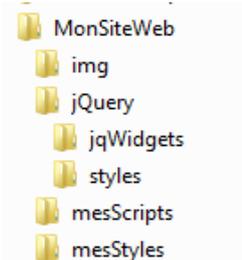
« **JavaScript** (quelques fois abrégé **JS**) est un langage de programmation de **scripts** principalement utilisé dans les pages web interactives. C'est un langage **orienté objet** ... » Wikipédia

Activité 3 : Recherchez les fichiers de la **colonne 1** ci-dessous dans les sous-répertoires de **jqwidgets-ver_x.x.x** (les chemins sont donnés ci-dessus) et **placez-les** dans les répertoires de votre projet (identifiés en **colonne 2**).

Je place les
fichiers JQuery
dans les
répertoires de mon site



1	2
Fichiers à placer dans →	le répertoire de votre projet
jquery-x.x.x.min.js x.x.x correspond au numéro de version du framework	jQuery
jqx.base.css	styles
jqxcore.js jqxchart.js jqxgauge.js jqxbuttons.js	jqWidgets



Votre projet est presque prêt, il reste à modifier les chemins dans le fichier **testjauge.html** pour qu'ils accèdent aux **fichiers nécessaires à la jauge**.



Lisez l'annexe 2 : « **Différence entre les chemins relatifs et les chemins absolus** » avant de poursuivre.

Exemple d'utilisation des chemins relatifs :

Le fichier **setting.html**, situé dans le répertoire "**jqxgauge**" (ci-contre) contient le lien :

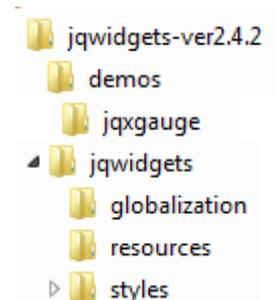
```
<link rel="stylesheet" href="../../jqwidgets/styles/jqx.base.css" type="text/css" />
```

Ce lien accède au fichier **jqx.base.css** en suivant le chemin relatif :

```
../../jqwidgets/styles
```



En effet, à partir de la position du répertoire « **jqxgauge** », il faut "remonter" deux répertoires avant de "redescendre" dans le répertoire « **styles** » !





Activité 4 : Modifiez les chemins relatifs dans l'en-tête de "testjauge.html" placé dans votre projet pour qu'ils accèdent aux différents fichiers nécessaires à votre jauge.

Je modifie les chemins relatifs pour accéder aux fichiers nécessaires à mon widget.



Get value Set value

Ouvrez la page "testjauge.html" dans un navigateur.

Si les fichiers sont dans les bons répertoires de votre projet et si les chemins relatifs sont justes, l'affichage doit ressembler à la copie d'écran ci-contre !

⚠ Remarque importante

Contrairement aux langages HTML, le Javascript ne tolère aucune erreur !

La sanction est immédiate : rien ne fonctionne !

A titre d'exemple, vous allez ajouter une parenthèse comme ci-dessous.

```
$('#setValueButton').click(function () {
    $('#gaugeContainer').jqxGauge('value', 220));
});
```

Sauvegarder le fichier *testjauge.html* et **rafraîchissez** la page dans le navigateur.

Seul le texte : **Get value Set value** s'affiche !



Vous devez donc **faire très attention** lorsque vous faites un copier-coller du code.

Supprimez la parenthèse pour retrouver la jauge !

Vous allez maintenant modifier le script JQuery de la jauge ci-dessus afin d'obtenir celle à réaliser (figure 6).

Au préalable, vous allez encore **simplifier** le code de la page fournie en supprimant celui correspondant aux deux boutons.

Get value Set value

Transformez le code du script ci-dessous en un **commentaire en plaçant : /* et */**

```

/*
 * $('#setValueButton').jqxButton({ width: 70, theme: 'classic' });
 * $('#getValueButton').jqxButton({ width: 70, theme: 'classic' });
 * $('#getValueButton').click(function () {
 *     alert(Math.round($('#gaugeContainer').jqxGauge('value')));
 * });
 * $('#setValueButton').click(function () {
 *     $('#gaugeContainer').jqxGauge('value', 220);
 * });
 */

```



Transformez également le code HTML ci-dessous en un **commentaire en plaçant : <!-- et -->**.

```

<br /><br />
<!-- <div id="getValueButton" style="display: inline;">Get value</div>
<div id="setValueButton" style="display: inline;">Set value</div> -->

```

Remarque: Synthaxe de commentaires javascript sur plusieurs lignes : `/* */`
 Synthaxe de commentaires HTML sur plusieurs lignes : `<!-- -->`

Rappel de l'objectif du TP : Présenter des données dans des widgets et les intégrer à une page web.



Figure 5 : Jauge fournie sur le site jqWidget

Transformation

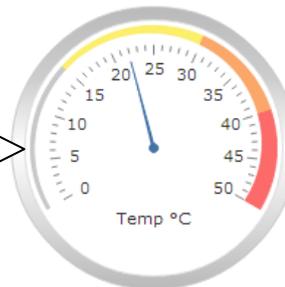


Figure 6 : Jauge à réaliser et à planter dans une page web

2.1.3. Modification des attributs de l'objet "Gauge"

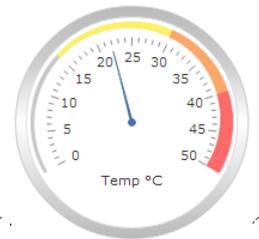
Objectif : adapter le code JQuery du Widget « Gauge » proposé sur le site dédié au framework pour qu'il affiche la température issue d'un capteur.

Pour intégrer du code JavaScript dans une page HTML, on peut le placer dans une balise `<script>`.

Le code de la jauge à modifier est donné dans l'encadré ci-dessous:

```
<script type="text/javascript">
  $(document).ready(function () {
    $('#gaugeContainer').jqxGauge({
      ranges: [{ startValue: 0, endValue: 55, style: { fill: '#C9C9C9', stroke: '#C9C9C9' }, endWidth: 5, startWidth: 1 },
        { startValue: 55, endValue: 110, style: { fill: '#FCF06A', stroke: '#FCF06A' }, endWidth: 10, startWidth: 5 },
        { startValue: 110, endValue: 165, style: { fill: '#FCA76A', stroke: '#FCA76A' }, endWidth: 15, startWidth: 10 },
        { startValue: 165, endValue: 220, style: { fill: '#FC6A6A', stroke: '#FC6A6A' }, endWidth: 20, startWidth: 15}],
      ticksMinor: { interval: 5, size: '5%' },
      ticksMajor: { interval: 10, size: '9%' },
      value: 0,
      colorScheme: 'scheme03',
      animationDuration: 1200
    });
    $('#gaugeContainer').jqxGauge('value', 140);
  });
</script>
```

Code de
la jauge fournie



A expérimenter pour comprendre l'effet obtenu sur l'OBJET "Gauge" lorsqu'on modifie ses ATTRIBUTS

- Remplacez `endValue: 220`, par `endValue: 200` dans le code ci-dessus, Sauvegarder le fichier et rafraîchissez la page dans le navigateur pour voir l'effet obtenu.
- Remplacez maintenant `endWidth: 20, startWidth: 15` par `endWidth: 15, startWidth: 20` Sauvegarder le fichier et rafraîchissez la page dans le navigateur pour voir l'effet obtenu.



Tout est dans la DOCUMENTATION !

Les données décrivant la structure interne d'un objet sont appelées ses attributs.

Le mot "ranges" correspond à un **ATTRIBUT (propriété en anglais)** de l'**OBJET** "Gauge" (notions approfondies en cours). Les attributs des widgets jQuery sont décrits dans la documentation du framework jqWidgets.



Sur le site jqwidgets.com ([Demo](#) → JqxGauge → Gauge Settings, ouvrez l'onglet [API Reference](#))
La page donne accès à l'ensemble des attributs de l'objet "Gauge". Vous venez d'agir sur l'attribut "ranges". Celui-ci est décrit par le texte ci-dessous.

Name	Type	Default
ranges	Array	[]

This property is array of objects. Each object is different range. The range is colored area with specified size. Here is sample initialization of the jqxGauge with the ranges array:

Attribut (Property)

Pour chaque attribut, la documentation donne le **TYPE** de la donnée (vous connaissez déjà les types string, integer...) et sa **valeur par défaut** (ici aucune valeur).

La **valeur par défaut** est la valeur prise par l'attribut de l'objet, en l'absence d'information, lorsque cet objet est **créé**.

Exemple : par défaut, la largeur (width) de la jauge est de 350px !



Activité 5 : La jauge à réaliser doit avoir un diamètre de 200 pixels. Sachant que vous devez utiliser l'attribut "label" pour modifier l'étendu des valeurs qu'elle peut afficher et que d'autres attributs (à identifier dans la documentation) doivent être rajoutés dans le code, **modifiez-le** pour obtenir la figure ci-contre.

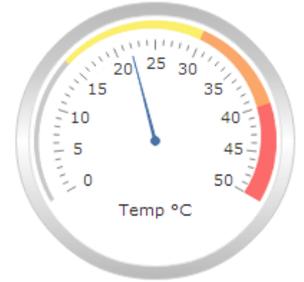
Je cherche

*les ATTRIBUTS à ajouter
et les modifications*

à apporter au code existant !

Appel prof

Pour faire vérifier votre travail.



Vous venez de modifier l'apparence de la jauge en modifiant ses attributs. Il faut maintenant la placer dans la page. Ceci est réalisé par les lignes de code ci-dessous.

Code dans le script :

La fonction **\$()** recherche la balise identifiée par gaugeContainer dans la page Web (# ⇔ **sélecteur**).

```
$("#gaugeContainer").jqxGauge({
  // ... liste des attributs
});
```

La méthode de construction de l'objet "Gauge" jqxGauge est appliquée (**un point ⇔ appliquer**) à cette balise avec les attributs placés entre accolade.

Ce code peut se traduire par :

*Hé, l'élément avec l'ID
gaugeContainer...*

...j'ai créé une jauge. Affiche-la

```
$("#gaugeContainer").jqxGauge({
  // ... liste des attributs
});
```



*... avec les valeurs des attributs par
défaut ou celles présentes entre les
accolades !*

Code dans le corps de la page HTML :

Pour que le navigateur affiche cet objet, il faut lui préciser : "**sa position dans la page HTML**".

C'est le rôle du code ci-dessous :

```
<div id="gaugeContainer"></div>
```



Conclusion : Pour qu'un objet jqWidget s'affiche dans une page HTML, il faut le décrire dans un script écrit en Javascript et le placer dans une balise <div> unique.

Maintenant que vous savez comment afficher une jauge dans une page HTML, vous aller la contrôler avec un widget "Button".

2.1.4. Contrôle de l'objet "Gauge" avec un objet "Button"

Objectif : régler la valeur affichée par l'objet « Gauge » avec un objet "Button".

Set value

Enlevez les caractères (*/* et */*) placés précédemment dans le script et supprimez le code ci-dessous.

```

//
//      $('#getValueButton').jqxButton({ width: 70, theme: 'classic' });
//      $('#getValueButton').click(function () {
//          alert(Math.round($('#gaugeContainer').jqxGauge('value')));
//      });
//

```

Enlevez les caractères (*<!-- et -->*) placés précédemment et supprimez le code ci-dessous.

```

//
//      <div id="getValueButton" style="display: inline;">Get value</div>
//

```

Activité 6 : En vous inspirant de l'exemple donné page précédente, **expliquez** ce que fait le code contenu dans les deux encadrés ci-dessous. **Complétez** le DR3.

```

//
//      $('#setValueButton').jqxButton({ width: 70, theme: 'classic' });
//
//      $('#setValueButton').click(function () {
//          $('#gaugeContainer').jqxGauge('value', 22);
//      });
//
//      <div id="setValueButton1" style="display: inline;">22°C</div>
//

```



Activité 7 : Rajoutez un bouton permettant de positionner l'aiguille sur la valeur 12.

Appel prof

Pour faire vérifier votre travail.

Vous allez maintenant sauvegarder le **code de votre jauge (et UNIQUEMENT LUI)** dans un fichier **javascript externe** à la page HTML. Vous pourrez ainsi le **réutiliser** dans plusieurs pages.

Supprimez le code des boutons dans le script et dans le corps de la page HTML.

Activité 8 : Placez le code du script (à l'intérieur des balises script) sur une nouvelle page, enregistrez-la sous le nom « gauge.js » dans le répertoire mesScripts et écrivez le lien vers ce nouveau fichier dans l'en-tête de la page HTML.

La page HTML testjauge.html doit ressembler à la copie d'écran ci-dessous.

```

<!DOCTYPE html>
<html lang="en">
<head>
  <title id='Description'>jqxGauge Basic Demo</title>
  <link rel="stylesheet" href="jqwidgets/styles/jqx.base.css" type="text/css" />
  <script type="text/javascript" src="jqwidgets/scripts/jquery-1.7.2.min.js"></script>
  <script type="text/javascript" src="jqwidgets/jqxcore.js"></script>
  <script type="text/javascript" src="jqwidgets/jqxchart.js"></script>
  <script type="text/javascript" src="jqwidgets/jqxgauge.js"></script>
  <script type="text/javascript" src="jqwidgets/jqxbuttons.js"></script>
  <script type="text/javascript" src="mesScripts/gauge.js"></script>
</head>
<body style="background:white;">
  <div id="gaugeContainer"></div>
</body>
</html>

```

Vous êtes maintenant prêt à **intégrer votre jauge** dans les pages d'un site web !

2.2. Partie 2 : intégration du widget "Gauge" dans un site web

Objectif : intégrer le widget "Gauge" dans la page *mesures.html*.

Le sous répertoire **Partie_2_Site** situé dans le répertoire **Problématique** de votre TP contient la page *mesures.html*. Celle-ci ressemble à la copie d'écran ci-dessous. (A tester).



Figure 5: Pages du site

La page est prête à recevoir votre widget "Gauge".

On donne ci-dessous un extrait du contenu du fichier *mesures.html*. Ouvrez-le dans un navigateur.



```
<head>
  <title> Présentation des données avec des Widgets JQuery </title>
  <!-- A compléter -->
<!------->
  <link rel="stylesheet" type="text/css" href="mesStyles/CSS3TP5.css"/>
</head>
<div class="main">
  <div class="content">
  <!-- A compléter -->
  </div> <!--Content-->
</div><!-- /.main -->
```

Activité 9 : Complétez les parties du code ci-dessus, identifiées par <!-- A compléter -->, pour que le widget "Gauge" apparaisse dans la page. Imprimez la page *mesures.html* et joignez-la au document réponse.

3. Synthèse

[🕒 20mn]

Objectif : Identifier les modifications nécessaires à l'ajout d'un nouveau widget.

Dans le cadre d'un projet, on souhaite afficher l'évolution d'une température ambiante dans un widget "**jqxChart Line Series**".

Un exemple (ci-contre) est proposé sur le site jqwidget.com.

Activité 10 : Etablissez la liste des modifications à apporter à cet exemple pour afficher l'évolution de la température.

Identifiez quelques-uns des attributs (et leur type) concernés par vos modifications.

Répondez sur le DR5

jqxChart Line Series Example



Figure 6: Chart jQuery

Glossaire

Framework :

En programmation informatique, un **framework** est un *kit* de [composants logiciels](#)...(Wikipédia)

Template

Un **gabarit**, souvent nommé à tort avec les anglicismes **template** ou **layout**, est un patron de mise en page où l'on place images et textes. Aussi, il est souvent utilisé de manière répétitive pour créer des documents présentant une même structure. (Wikipédia)

UTF-8 (*UCS transformation format 8 bits*) est un [codage de caractères informatique](#) conçu pour coder l'ensemble des caractères internationaux d'[Unicode](#) en restant compatible avec la norme [ASCII](#) limitée à l'anglais, mais très largement répandue depuis des décennies. (Wikipédia)

Bibliographie –

- Comprendre les propriétés de CSS - Utiliser CSS pour le Web

WebDesign Hors-série n°15

- CSS3 POUR LES WEB DESIGNERS

Dan Cederholm

EYROLLES

Webographie

- Webmastering - Conception d'un site web
- Exemple de page réalisée avec des CSS3
- JQWidgets
- Autres bibliothèques de Widgets

<http://goo.gl/iF0Jdn>

<http://CSS3exp.com/code>

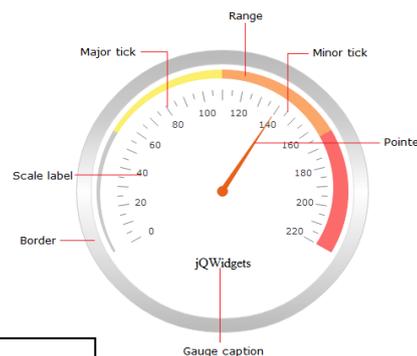
<http://www.jqwidgets.com/>

<http://goo.gl/35rlhw>

Annexe 1 : La documentation du widget "jqxGauge"

Introduction

jqxGauge displays an indicator within a range of values. Gauges can be used in a table or matrix to show the relative value of a field in a range of values in the data region, for example, as a KPI. It supports SVG and VML rendering.



Getting Started

Include all required JavaScript & CSS files: jqxGauge uses jQuery for basic JavaScript tasks like elements selection and events handling.

You need to include the jQuery JavaScript file, the jQWidgets core framework file - jqxcore.js, the main jqxChart plug-in file - jqxchart.js file, the jqxGauge plugin file jqxgauge.js and the base jQWidgets stylesheet - jqx.base.css:

① Liste des fichiers **INDISPENSABLES** à la mise en œuvre de la jauge.

```
<link rel="stylesheet" href="../../jqwidgets/styles/jqx.base.css" type="text/css" />
<script type="text/javascript" src="../../scripts/jquery-1.7.2.min.js"></script>
<script type="text/javascript" src="../../jqwidgets/jqxcore.js"></script>
<script type="text/javascript" src="../../jqwidgets/jqxchart.js"></script>
<script type="text/javascript" src="../../jqwidgets/jqxgauge.js"></script>
```

Create a DIV element inside the body of your page. You need to define an element which will serve as a container for the gauge. DIV elements are convenient because you can easily position them. Give the DIV element an ID like 'gaugeContainer' or something that you like.

```
<div id="gaugeContainer"></div>
```

② Une ligne suffit pour intégrer la jauge dans une page HTML !

Insert the jQuery document ready code inside the head section of your page:

```
$(document).ready(function() {
    // jqxGauge initialization code goes here.
})
```

③ Code JQuery : Attends le chargement de la page HTML avant d'appliquer le script.

Prepare the gauge display settings and initialize the gauge.

```
$('#gaugeContainer').jqxGauge({
    ranges: [{ startValue: 0, endValue: 55, style: { fill: '#C9C9C9', stroke: '#C9C9C9' },
    endWidth: 5, startWidth: 1 },
            { startValue: 55, endValue: 110, style: { fill: '#FCF06A', stroke: '#FCF06A' },
    endWidth: 10, startWidth: 5 },
            { startValue: 110, endValue: 165, style: { fill: '#FCA76A', stroke: '#FCA76A' },
    endWidth: 15, startWidth: 10 },
            { startValue: 165, endValue: 220, style: { fill: '#FC6A6A', stroke: '#FC6A6A' },
    endWidth: 20, startWidth: 15}],
    ticksMinor: { interval: 5, size: '5%' },
    ticksMajor: { interval: 10, size: '9%' },
    value: 0,
    colorScheme: 'scheme03',
    animationDuration: 1200
});
```

④ Code JQuery. Paramétrage de l'**OBJET** "Gauge" : Ce que vous allez devoir adapter au besoin de l'application!

The code above specifies the gauge's ranges, value and color scheme, animationDuration, ticksMajor and ticksMinor size and interval.

At last we will set jqxGauge's value:

```
$('#gaugeContainer').jqxGauge('value', 140);
```

⑤ Exemple de Code JQuery permettant de changer la valeur affichée par la jauge.

If you want to get gauge's value use the following code:

```
var value = $('#gaugeContainer').jqxGauge('value');
```

⑥ Exemple de Code JQuery permettant de connaître la valeur affichée par la jauge.

Annexe 2 : Chemins relatifs et chemins absolus

« Souvent source d'erreurs, de **pages introuvables**, **d'images manquantes** ou de **styles non appliqués**, le format d'écriture des liens est très important. Et pourtant, on ne prend pas forcément garde à la façon dont on fait pointer ces derniers vers les différents fichiers cible. Suivant la convention d'écriture utilisée, de nombreux problèmes peuvent survenir. En effet, deux types d'écritures cohabitent, chacune avec leur spécificité propre :

- le chemin **relatif** (absence de slash initial)
- le chemin **absolu** (slash en préfixe du chemin)

Chemin relatif

`contenu/page2.html`

La cible (`page2.html`) est cherchée par le navigateur dans le sous-répertoire "contenu".

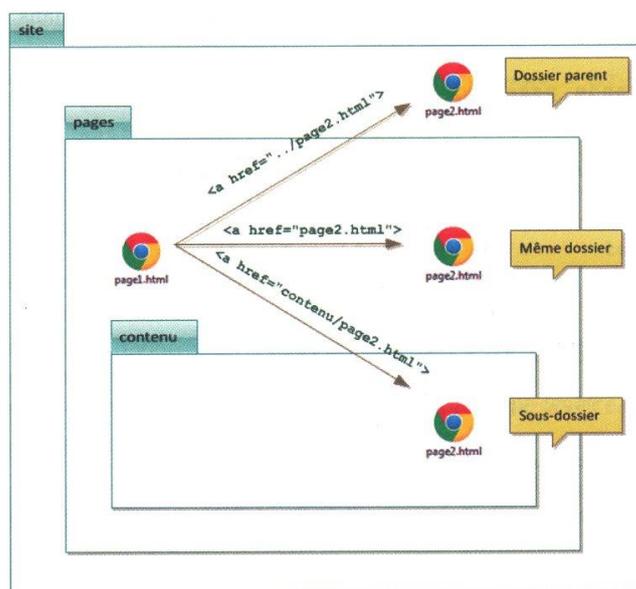
`./page2.html`

`./` : signifie que la page est cherchée à partir du répertoire courant (peut être omis).

`../page2.html`

`../` : signifie que la page est cherchée à partir du répertoire parent (on remonte d'un niveau).

Il est tout à fait possible de remonter plusieurs répertoires. Pour cela il faudra cumuler les `../` et ainsi pour remonter de deux niveaux, il sera possible d'écrire `../../dossier/page.html`.



Chemin absolu

`/dossier/page.html`

Le slash initial précise que l'on ne se réfère plus à l'emplacement courant mais que l'on remonte directement à la racine pour ensuite préciser le chemin complet.

Dans bien des cas, le chemin absolu est la convention d'écriture la plus sûre, mais aussi la moins souple si l'arborescence est amenée à changer. La page cible sera cherchée par le navigateur à partir de la racine du site.

Dans notre exemple, le fichier `page.html` sera recherché directement à partir de la racine du domaine : `http://www.mondomaine.fr/dossier/page.html`. »

Source : alscreations.com

Table des illustrations

Figure 1 : Exemples de widgets (jQuery)3
Figure 2 : Jauge à réaliser5
Figure 3 : Répertoires du TP.....6
Figure 4 : Répertoires du site6
Figure 5: Pages du site.....12
Figure 6: Chart jQuery12