

```

<!-- begin of customized social share buttons P.Q.-->
<link href='https://use.fontawesome.com/releases/v5.0.6/css/all.css'
rel='stylesheet' type='text/css' />
<!--script
src='https://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js'
type='text/javascript'></script-->
<script src='https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js'
type='text/javascript'></script>
<link
href='https://philquand.github.io/Javascript/FloatingShareBar/css/jquery-
social-share-bar.css' rel='stylesheet' type='text/css' />
<script
src='https://philquand.github.io/Javascript/FloatingShareBar/js/jquery-social-
share-bar.js' type='text/javascript'></script>

<div data-caption='' data-lang='' data-url='' id='postBodyforSocialNetworks'
style='display:none'></div>
<div id='showBodyforSocialNetworks'></div>
<div class='myShareBar' id='share-bar'></div>

<script>
$(document).ready(function() {
    $('#share-bar').share({
        'channels': ['facebook', 'twitter', 'linkedin', 'pdf', 'email']
    });
});
</script>
<style>
header.centered-top-container.sticky.animating div div nav #page_list_top
#HTML2 div #share-bar{
    display: none;
}
</style>
<!-- end of customized social share buttons P.Q.-->

<script src='https://philquand.github.io/Javascript/flagsRepository.js'
type='text/javascript' ></script>
<style>
h1 {
    display: block;
    font-size: 2em;
    margin-top: 0.67em;
    margin-bottom: 0.67em;
    margin-left: 0;
    margin-right: 0;
    font-weight: bold;
}

```

```
h2 {
  display: block;
  font-size: 1.5em;
  margin-top: 0.83em;
  margin-bottom: 0.83em;
  margin-left: 0;
  margin-right: 0;
  font-weight: bold;
}
h3 {
  display: block;
  font-size: 1.17em;
  margin-top: 1em;
  margin-bottom: 1em;
  margin-left: 0;
  margin-right: 0;
  font-weight: bold;
}
h4 {
  display: block;
  font-size: 1em;
  margin-top: 1.33em;
  margin-bottom: 1.33em;
  margin-left: 0;
  margin-right: 0;
  font-weight: bold;
}
h5 {
  display: block;
  font-size: .83em;
  margin-top: 1.67em;
  margin-bottom: 1.67em;
  margin-left: 0;
  margin-right: 0;
  font-weight: bold;
}
h6 {
  display: block;
  font-size: .67em;
  margin-top: 2.33em;
  margin-bottom: 2.33em;
  margin-left: 0;
  margin-right: 0;
  font-weight: bold;
}
.reference:before {
  display: block;
  content: " ";
}
```

```
margin-top: -140px;
height: 140px;
visibility: hidden;
pointer-events: none;
}

.reference:target div {
    background-color: #BFEFFF;
}

.incrustation {
    background-color: #e8f6f8;
    padding-top: 0.5em;
    padding-bottom: 0.5em;
    padding-left: 1em;
    padding-right: 1em;
}

blockquote {
    text-align: left;
    font-size: 20px;
    line-height: 22pt;
}

.res-title h3, .res-title h3.span, .res-title h3.a {
color:red;
padding: 10px 0 0 10px;
font-size: 24px;
line-height: 1;
//max-height: 24px;
margin:0 15px 10px 0;
display: block
}

.html2pdf-post-title, .html2pdf-post-date, .html2pdf-post-signature, .lang-in-
post {
display: none;}

.authors-in-post {
color: blue;
font-style: italic
}

.authors {
color:black;
padding: 0 0 10px 10px;
font-size: 16px;
line-height: 1;
```

```
//max-height: 16px;
margin:0 15px 10px 0;
display: block
}

.res-img {
padding: 30px 1em 0 0;
float: left
}

.myflat-button{
cursor:pointer;
//display:inline-block;
font-weight:700;
text-transform:uppercase;
border-radius:2px;
padding:8px;
margin:-8px
}
.myflat-icon-button{
background:0 0;
border:0;
margin:0;
outline:0;
padding:0;
margin:-12px;
padding:12px;
cursor:pointer;
box-sizing:content-box;
//display:inline-block;
line-height:0
}
.myflat-button.ripple:hover{
background-color:rgba(244,61,61,.12)
}
.myflat-button.ripple .splash{
background-color:rgba(244,61,61,.4)
}

.mySumFeat, .mySumReg {
line-height: 1.6;
}
.mySumFeat {
color: black;
}
.mySumReg {
color: grey;
}
```

```
.blogger, .image-attribution {
display: none
}

.legend {
padding:.5em .75em;
line-height: 18px;
color: #555;
background:white;
}

.legend i {
width: 18px;
height: 18px;
float: left;
margin-right: 8px;
//margin-right: 1em;
opacity: 0.7;
}

.leaflet-container.default-cursor-enabled {
cursor: default;
}

.navigationbar {
width: 100%; /* change the width of the navigation bar */
height: 50px; /* change the height of the navigation bar */
}

.navigationcss {
margin: 0 auto;
padding: 0;
}

.navigationcss ul {
float: none;
list-style: none;
margin: 0;
padding: 0;
overflow: visible;
}

.navigationcss li a, .navigationcss li a:link, .navigationcss li a:visited,
.navigationcss li.inactive a {
color: #F4F43E; /* change color of the main links */
display: block;
margin: 0;
padding: 10px 30px; /* change the first number for the top/bottom spacing,
and the second number for left/right spacing */
}
```

```
}

.navigationcss li.active a {
color: #ffffff; /* change color of the active link */
border-bottom: 4px solid currentColor;
}

.navigationcss li a:hover, .navigationcss li a:active {
color: #ffffff ; /* change the color of the links when hovered over */
margin: 0;
padding: 10px 30px; /* make sure these are the same as the section above! */
}

.navigationcss li {
float: none;
display: inline-block;
list-style: none;
margin: 0;
padding: 0;
}

.navigationcss li ul {
z-index: 9999;
position: absolute;
left: -999em;
height: auto;
margin: 0;
padding: 0;
}

.navigationcss li:hover ul, .navigationcss li li:hover ul, .navigationcss li
li li:hover ul, .navigationcss li.sfhover ul, .navigationcss li li.sfhover ul,
.navigationcss li li li.sfhover ul {
left: auto;
}

.dropdown-content {
display: inline-block;
//position: fixed;
position: absolute;
z-index: 9999;
//left: -999em;
height: auto;
width: 150px;
margin: 0;
padding: 0;
}
```

```

.dropdown-content li {
  float: none;
  list-style: none;
  margin: 0;
  padding: 0;
}

.dropdown-content li a, .dropdown-content li a:link, .dropdown-content li
a:visited, .dropdown-content li.inactive a {
  color: black; /* change color of the active link */
}

.dropdown-content li.active a {
  color: red; /* change color of the active link */
  border-bottom: 4px solid currentColor;
}

.dropdown-content li a:hover, .dropdown-content li a:active {
  background-color: #ddd;
  color: red;
}

.dropdown-content li a {
  display: none;
  background-color: #f1f1f1;
  min-width: 160px;
  box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
  z-index: -1;
  color: black;
  padding: 12px 16px;
  text-decoration: none;
  padding: 10px 30px; /* change the first number for the top/bottom spacing,
and the second number for left/right spacing */
  float: none;
}

.dropdown-content li ul {
  margin-left: 200px;
  margin-top: -50px;
}

</style>

<div class="PageList" id="contNavetFlags">
<!--Start Navigation -->
<div id='navigationGB' style="display: none;">
  <div class="navigationbar" lang="fr" style="display: block; text-align:
left;">

```

```

    <ul id='frnavigation' class='navigationcss'>
      <li><a href="https://www.linternationaledessavoirspourtous.org/p/la-
science-pour-le-plus-grand-nombre.html?lang=fr">Appel</a></li>
      <!--li><a href="" ontouchstart="return false">l'Équipe éditoriale</a>
<ul-->
      <li><a href="https://www.linternationaledessavoirspourtous.org/p/qui-
sommes-nous.html?lang=fr">Qui sommes-nous ?</a></li>
      <li><a
href="https://www.linternationaledessavoirspourtous.org/p/contacts.html?
lang=fr">Contacts</a></li>
      <!--/ul></li-->
      <li><a
href="https://www.linternationaledessavoirspourtous.org/p/critiques.html?
lang=fr">Critiques</a></li>
      <li><a
href="https://www.linternationaledessavoirspourtous.org/p/luttes.html?
lang=fr">Luttes</a></li>
      <li><a
href="https://www.linternationaledessavoirspourtous.org/p/alternatives.html?
lang=fr">Alternatives</a></li>
      <li><a
href="https://www.linternationaledessavoirspourtous.org/p/liens.html?
lang=fr">Liens</a></li>
      <li><a href="javascript:frDropdownGB()" data-isplus="false">Plus...</a>
</li>
    </ul>
  </div>
  <div class="navigationbar" lang="en" style="display: block; text-align:
left;">
    <ul id='ennavigation' class='navigationcss'>
      <li><a href="https://www.linternationaledessavoirspourtous.org/p/la-
science-pour-le-plus-grand-nombre.html?lang=en">Call</a></li>
      <li><a href="https://www.linternationaledessavoirspourtous.org/p/qui-
sommes-nous.html?lang=en">Who are we?</a></li>
      <li><a
href="https://www.linternationaledessavoirspourtous.org/p/contacts.html?
lang=en">Contacts</a></li>
      <li><a
href="https://www.linternationaledessavoirspourtous.org/p/critiques.html?
lang=en">Critics</a></li>
      <li><a
href="https://www.linternationaledessavoirspourtous.org/p/luttes.html?
lang=en">Struggles</a></li>
      <li><a
href="https://www.linternationaledessavoirspourtous.org/p/alternatives.html?
lang=en">Alternatives</a></li>
      <li><a
href="https://www.linternationaledessavoirspourtous.org/p/liens.html?

```



```

lang=en">Links</a></li>
    <li><a href="javascript:enDropdownGB()" data-isplus="false">More...</a>
</li>
</ul>
</div>
<div class="navigationbar" lang="de" style="display: block; text-align:
left;">
    <ul id='denavigation' class='navigationcss'>
        <li><a href="https://www.linternationaledessavoirspourtous.org/p/la-
science-pour-le-plus-grand-nombre.html?lang=de">Aufruf
</a></li>
        <li><a href="https://www.linternationaledessavoirspourtous.org/p/qui-
sommes-nous.html?lang=de">Über uns</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/contacts.html?
lang=de">Kontakt</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/critiques.html?
lang=de">Kritiken</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/luttes.html?
lang=de">Kämpfe</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/alternatives.html?
lang=de">Alternativlösungen</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/liens.html?
lang=de">Links</a></li>
        <li><a href="javascript:itDropdownGB()" data-isplus="false" >Mehr...
</a></li>
    </ul>
</div>
<div class="navigationbar" lang="it" style="display: block; text-align:
left;">
    <ul id='itnavigation' class='navigationcss'>
        <li><a href="https://www.linternationaledessavoirspourtous.org/p/la-
science-pour-le-plus-grand-nombre.html?lang=it">Appello</a></li>
        <li><a href="https://www.linternationaledessavoirspourtous.org/p/qui-
sommes-nous.html?lang=it">Chi siamo?</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/contacts.html?
lang=it">Contatti</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/critiques.html?
lang=it">Critiche</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/luttes.html?
lang=it">Lotte</a></li>
        <li><a

```

```
href="https://www.linternationaledessavoirspourtous.org/p/alternatives.html?
lang=it">Alternative</a></li>
    <li><a
href="https://www.linternationaledessavoirspourtous.org/p/liens.html?
lang=it">Links</a></li>
    <li><a href="javascript:itDropdownGB()" data-isplus="false">Di più...</a>
</li>
</ul>
</div>
<div class="navigationbar" lang="pt" style="display: block; text-align:
left;">
    <ul id='ptnavigation' class='navigationcss'>
        <li><a href="https://www.linternationaledessavoirspourtous.org/p/la-
science-pour-le-plus-grand-nombre.html?lang=pt">Chamamento</a></li>
        <li><a href="https://www.linternationaledessavoirspourtous.org/p/qui-
sommes-nous.html?lang=pt">Quem somos nós?</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/contacts.html?
lang=pt">Contactos</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/critiques.html?
lang=pt">Críticas</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/luttes.html?
lang=pt">Lutas</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/alternatives.html?
lang=pt">Alternativas</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/liens.html?
lang=pt">Links</a></li>
        <li><a href="javascript:ptDropdownGB()" data-isplus="false">Mais...</a>
</li>
    </ul>
</div>
<div class="navigationbar" lang="es" style="display: block; text-align:
left;">
    <ul id='esnavigation' class='navigationcss'>
        <li><a href="https://www.linternationaledessavoirspourtous.org/p/la-
science-pour-le-plus-grand-nombre.html?lang=es">Llamamiento</a></li>
        <li><a href="https://www.linternationaledessavoirspourtous.org/p/qui-
sommes-nous.html?lang=es">¿Quiénes somos?</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/contacts.html?
lang=es">Contactos</a></li>
        <li><a
href="https://www.linternationaledessavoirspourtous.org/p/critiques.html?
lang=es">Críticas</a></li>
```

```
<li><a
href="https://www.linternationaledessavoirspourtous.org/p/luttes.html?
lang=es">Luchas</a></li>
<li><a
href="https://www.linternationaledessavoirspourtous.org/p/alternatives.html?
lang=es">Alternativas</a></li>
<li><a
href="https://www.linternationaledessavoirspourtous.org/p/liens.html?
lang=es">Enlaces</a></li>
<li><a href="javascript:esDropdownGB()" data-isplus="false">Más...</a>
</li>
</ul>
</div>
</div>
<!--End Navigation -->
<div id="Flags" style="display: inline-block;">
<div id="frHandler" style="display: inline-block">

</div>
<div id="enHandler" style="display: inline-block">

</div>
<div id="deHandler" style="display: inline-block">

</div>
<div id="esHandler" style="display: inline-block">

</div>
<div id="itHandler" style="display: inline-block">

</div>
<div id="ptHandler" style="display: inline-block">

```

```
</div>
```

```
</div>
```

```
</div>
```

```
<script type="text/javascript" />
```

```
function msgPostTittleFR() {
```

```
    var msgPostTittle = [{
```

```
        href: "https://www.linternationaledessavoirspourtous.org/p/la-science-  
pour-le-plus-grand-nombre.html",
```

```
        tittle: "La science pour le plus grand nombre, pas pour l'argent"
```

```
    },
```

```
    {
```

```
        href: "https://www.linternationaledessavoirspourtous.org/p/qui-sommes-  
nous.html",
```

```
        tittle: "Qui sommes-nous ?"
```

```
    },
```

```
    {
```

```
        href:
```

```
"https://www.linternationaledessavoirspourtous.org/p/contacts.html",
```

```
        tittle: "Contacts"
```

```
    },
```

```
    {
```

```
        href:
```

```
"https://www.linternationaledessavoirspourtous.org/p/critiques.html",
```

```
        tittle: "Critiques"
```

```
    },
```

```
    {
```

```
        href: "https://www.linternationaledessavoirspourtous.org/p/luttes.html",
```

```
        tittle: "Luttes"
```

```
    },
```

```
    {
```

```
        href:
```

```
"https://www.linternationaledessavoirspourtous.org/p/alternatives.html",
```

```
        tittle: "Alternatives"
```

```
    },
```

```
    {
```

```
        href: "https://www.linternationaledessavoirspourtous.org/p/liens.html",
```

```
        tittle: "Liens"
```

```
    }
```

```
];
```

```
return msgPostTittle;
```

```
}
```

```
function msgPostTittleEN() {
```

```
    var msgPostTittle = [{
```

```
        href: "https://www.linternationaledessavoirspourtous.org/p/la-science-
```

```

pour-le-plus-grand-nombre.html",
    tittle: "Science for the many, not for money"
},
{
    href: "https://www.linternationaledessavoirspourtous.org/p/qui-sommes-
nous.html",
    tittle: "Who are we?"
},
{
    href:
"https://www.linternationaledessavoirspourtous.org/p/contacts.html",
    tittle: "Contacts"
},
{
    href:
"https://www.linternationaledessavoirspourtous.org/p/critiques.html",
    tittle: "Critics"
},
{
    href: "https://www.linternationaledessavoirspourtous.org/p/luttes.html",
    tittle: "Struggles"
},
{
    href:
"https://www.linternationaledessavoirspourtous.org/p/alternatives.html",
    tittle: "Alternatives"
},
{
    href: "https://www.linternationaledessavoirspourtous.org/p/liens.html",
    tittle: "Links"
}
};
return msgPostTittle;
}

```

```

function msgPostTittleDE() {
    var msgPostTittle = [{
        href: "https://www.linternationaledessavoirspourtous.org/p/la-science-
pour-le-plus-grand-nombre.html",
        tittle: "Wissenschaft für die Vielen statt Wissenschaft für Profit"
    },
    {
        href: "https://www.linternationaledessavoirspourtous.org/p/qui-sommes-
nous.html",
        tittle: "Über uns"
    },
    {
        href:

```

```

"https://www.linternationaledessavoirspourtous.org/p/contacts.html",
    tittle: "Kontakt"
},
{
    href:
"https://www.linternationaledessavoirspourtous.org/p/critiques.html",
    tittle: "Kritik"
},
{
    href: "https://www.linternationaledessavoirspourtous.org/p/luttes.html",
    tittle: "Kämpfe"
},
{
    href:
"https://www.linternationaledessavoirspourtous.org/p/alternatives.html",
    tittle: "Alternativlösungen"
},
{
    href: "https://www.linternationaledessavoirspourtous.org/p/liens.html",
    tittle: "Links"
}
];
return msgPostTittle;
}

function msgPostTittleIT() {
    var msgPostTittle = [{
        href: "https://www.linternationaledessavoirspourtous.org/p/la-science-
pour-le-plus-grand-nombre.html",
        tittle: "La scienza per i molti, e non per i soldi"
    },
    {
        href: "https://www.linternationaledessavoirspourtous.org/p/qui-sommes-
nous.html",
        tittle: "Chi siamo?"
    },
    {
        href:
"https://www.linternationaledessavoirspourtous.org/p/contacts.html",
        tittle: "Contatti"
    },
    {
        href:
"https://www.linternationaledessavoirspourtous.org/p/critiques.html",
        tittle: "Critiche"
    },
    {
        href: "https://www.linternationaledessavoirspourtous.org/p/luttes.html",

```

```

        tittle: "Lotte"
    },
    {
        href:
"https://www.linternationaledessavoirspourtous.org/p/alternatives.html",
        tittle: "Alternative"
    },
    {
        href: "https://www.linternationaledessavoirspourtous.org/p/liens.html",
        tittle: "Links"
    }
];
return msgPostTittle;
}

function msgPostTittlePT() {
    var msgPostTittle = [{
        href: "https://www.linternationaledessavoirspourtous.org/p/la-science-
pour-le-plus-grand-nombre.html",
        tittle: "Ciência para todos, não para o dinheiro"
    },
    {
        href: "https://www.linternationaledessavoirspourtous.org/p/qui-sommes-
nous.html",
        tittle: "Quem somos nós?"
    },
    {
        href:
"https://www.linternationaledessavoirspourtous.org/p/contacts.html",
        tittle: "Contactos"
    },
    {
        href:
"https://www.linternationaledessavoirspourtous.org/p/critiques.html",
        tittle: "Críticas"
    },
    {
        href: "https://www.linternationaledessavoirspourtous.org/p/luttes.html",
        tittle: "Lutas"
    },
    {
        href:
"https://www.linternationaledessavoirspourtous.org/p/alternatives.html",
        tittle: "Alternativas"
    },
    {
        href: "https://www.linternationaledessavoirspourtous.org/p/liens.html",
        tittle: "Links"
    }
];
return msgPostTittlePT;
}

```

```

    }
    ];
    return msgPostTittle;
}

function msgPostTittleES() {
    var msgPostTittle = [{
        href: "https://www.linternationaledessavoirspourtous.org/p/la-science-
pour-le-plus-grand-nombre.html",
        tittle: "Ciencia para todos, no por dinero"
    },
    {
        href: "https://www.linternationaledessavoirspourtous.org/p/qui-sommes-
nous.html",
        tittle: "¿Quiénes somos?"
    },
    {
        href:
"https://www.linternationaledessavoirspourtous.org/p/contacts.html",
        tittle: "Contactos"
    },
    {
        href:
"https://www.linternationaledessavoirspourtous.org/p/critiques.html",
        tittle: "Críticas"
    },
    {
        href: "https://www.linternationaledessavoirspourtous.org/p/luttes.html",
        tittle: "Luchas"
    },
    {
        href:
"https://www.linternationaledessavoirspourtous.org/p/alternatives.html",
        tittle: "Alternativas"
    },
    {
        href: "https://www.linternationaledessavoirspourtous.org/p/liens.html",
        tittle: "Enlaces"
    }
    ];
    return msgPostTittle;
}

function curRootSite() {

    return 'www.linternationaledessavoirspourtous.org';

}

```



```

</script>
<script type="text/javascript" />

LangForSummary = ["fr", "en", "de", "es", "it", "pt"];
img_thumb_width = 180; // Image Thumbnail Width
img_thumb_height = 180; // Image Thumbnail Height
summary_noimg = 500; // Length of Summary Text if no image is Used
summary_img_reg = 400; // Length of Summary Text if regular post has Image
Thumbnail
summary_img_feat = 1000; // Length of Summary Text if featured post has Image
Thumbnail

callFBSDKinit();

var frHandler = document.getElementById("frHandler");
frHandler.addEventListener("click", frLangTB, false);

var enHandler = document.getElementById("enHandler");
enHandler.addEventListener("click", enLangTB, false);

var enHandler = document.getElementById("deHandler");
enHandler.addEventListener("click", deLangTB, false);

var enHandler = document.getElementById("itHandler");
enHandler.addEventListener("click", itLangTB, false);

var enHandler = document.getElementById("ptHandler");
enHandler.addEventListener("click", ptLangTB, false);

var enHandler = document.getElementById("esHandler");
enHandler.addEventListener("click", esLangTB, false);

function frLangTB() {

    var msgPostTittle = msgPostTittleFR();

    var msgLangInscription = {
        header : "L'Internationale des Savoirs pour Tous",
        btInscrire : "s'inscrire",
        loc : 'fr_FR',
        setInscription : "S'inscrire à ce blog",
        setSharingText : "Obtenir le lien",
        setGetLinkText : "Lien copié dans le presse-papier",
        setCallforEmail : 'Adresse e-mail',
        setRecMsg : 'Recevoir des notifications par e-maill',
        setJumpLink : 'Lire la suite',
        setPopularPosts : 'Posts les plus consultés de ce blog',
        //setFeaturedPost : 'à la une...',
    }
}

```

```

        setFeaturedPost : '',
        setLocforDates : 'fr-FR',
        searchInput : "Rechercher dans ce blog",
        searchAction : "rechercher",
        setOtherPosts : "Autres articles",
        setPostFilter1 : "Affichage des posts correspondant à la recherche du
terme",
        setPostFilter2 : "Tout Afficher"
    };
    setLanguageTB("fr", msgPostTittle, msgLangInscription, "frnavigation");
}

```

```

function enLangTB() {

    var msgPostTittle = msgPostTittleEN();

    var msgLangInscription = {
        header : "The International of Knowledge for All",
        btInscrire : "subscribe",
        loc : 'en_GB',
        setInscription : 'Subscribe to this blog',
        setSharingText : "Get the link",
        setGetLinkText : "Link copied to clipboard",
        setCallforEmail : 'Email Address',
        setRecMsg : 'Get email notifications',
        setJumpLink : 'Read more',
        setPopularPosts : 'Popular posts from this blog',
        //setFeaturedPost : 'feature post...',
        setFeaturedPost : '',
        setLocforDates : "en-US",
        searchInput : "Search this blog",
        searchAction : "Search",
        setOtherPosts : "More posts",
        setPostFilter1 : "Showing posts matching the search for",
        setPostFilter2 : "SHOW ALL"
    };
    setLanguageTB("en", msgPostTittle, msgLangInscription, "ennavigation");
}

```

```

function deLangTB() {

    var msgPostTittle = msgPostTittleDE();

    var msgLangInscription = {
        header : 'Internationales Netzwerk « Wissen für Alle »',
        btInscrire : "anmeldung",
        loc : 'de_DE',
        setInscription : 'Diesen Blog abonnieren',

```

```

        setSharingText : "Holen Sie sich den Link",
        setGetLinkText : "Link in die Zwischenablage kopiert",
        setCallforEmail : 'Email Adresse',
        setRecMsg : 'E-mail benachrichtigungen erhalten',
        setJumpLink : 'Weiterlesen',
        setPopularPosts : 'Beliebte Posts aus diesem Blog',
        //setFeaturedPost : 'Feature-Post...',
        setFeaturedPost : '',
        setLocforDates : "de-DE",
        searchInput : "Suche im Blog",
        searchAction : "Suche",
        setOtherPosts : "WEITERE POSTS",
        setPostFilter1 : "Posts für die Suchanfrage",
        setPostFilter2 : "ALLE ANZEIGEN"
    };
    setLanguageTB("de", msgPostTittle, msgLangInscription, "denavigation");
}

function itLangTB() {

    var msgPostTittle = msgPostTittleIT();

    var msgLangInscription = {
        header : "L'Internazionale della Conoscenza per Tutti",
        btInscrire : "iscriviti",
        loc : 'it_IT',
        setInscription : 'Iscriviti a questo blog',
        setSharingText : "Ottieni il link",
        setGetLinkText : "Link copiato negli appunti",
        setCallforEmail : 'Indirizzo email',
        setRecMsg : 'Ricevi Notifiche email',
        setJumpLink : 'Continua a leggere',
        setPopularPosts : 'Post popolari in questo blog',
        //setFeaturedPost : 'caratteristica post',
        setFeaturedPost : '',
        setLocforDates : "it-IT",
        searchInput : "Cerca nel blog",
        searchAction : "circa",
        setOtherPosts : "ALTRI POST",
        setPostFilter1 : "Visualizzazione dei post corrispondenti alla
ricerca",
        setPostFilter2 : "MOSTRA TUTTO"
    };
    setLanguageTB("it", msgPostTittle, msgLangInscription, "itnavigation");
}

function ptLangTB() {

```

```

var msgPostTittle = msgPostTittlePT();

var msgLangInscription = {
    header : "Internacional do Saber para Todos",
    btInscrire : "subscrever",
    loc : 'pt_PT',
    setInscription : 'Subscrever este blogue',
    setSharingText : "Obter o link",
    setGetLinkText : "Link copiado para a área de transferência",
    setCallforEmail : 'Endereço de email',
    setRecMsg : 'Obter notificações por email',
    setJumpLink : 'Ler mais',
    setPopularPosts : 'Mensagens populares deste blogue',
    //setFeaturedPost : 'postagem de recurso',
    setFeaturedPost : '',
    setLocforDates : "pt-PT",
    searchInput : "Pesquisar nest blogue",
    searchAction : "pesquisar",
    setOtherPosts : "MAIS POSTAGENS",
    setPostFilter1 : "Mostrando postagens que correspondem à pesquisa
por",
    setPostFilter2 : "MOSTRAR TUDO"
};
setLanguageTB("pt", msgPostTittle, msgLangInscription, "ptnavigation");
}

function esLangTB() {

    var msgPostTittle = msgPostTittleES();

    var msgLangInscription = {
        header : "La Internacional del Saber para Todos",
        btInscrire : "subscribirse",
        loc : 'es_ES',
        setInscription : 'Registrar en este blog',
        setSharingText : "Obtenga el enlace",
        setGetLinkText : "Enlace copiado al portapapeles",
        setCallforEmail : 'Dirección de correo electrónico',
        setRecMsg : 'Recibir notificaciones por correo electrónico',
        setJumpLink : 'Leer más',
        setPopularPosts : 'Entradas populares de este blog',
        //setFeaturedPost : 'feature post',
        setFeaturedPost : '',
        setLocforDates : "es-ES",
        searchInput : "Buscar este blog",
        searchAction : "buscar",
        setOtherPosts : "MÁS ENTRADAS",
        setPostFilter1 : "Mostrando las entradas que coinciden con la búsqueda

```

```

de",
    setPostFilter2 : "MOSTRAR TODO"
};
setLanguageTB("es", msgPostTittle, msgLangInscription, "esnavigation");
}

function setLanguageTB(Lang, msgPostTittle, msgLangInscription, navID) {

    var rootSite = curRootSite();

    setLangByTagNameTB(Lang, "div");

    navIDSauv = navID;
    sauvMenuBarTyp = { navID : 0, DrpTyp : "" }
    setNavigation();

    setDivWidth();

    setRedirectOnLang(Lang, rootSite);
    var firstPostTittleLang = setPostTitleLang(msgPostTittle);
    setDataForSummary(Lang, msgPostTittle);
    setLangInscriptionTB(Lang, msgLangInscription, rootSite,
firstPostTittleLang);
    //setHomeCustom();

function setHomeCustom() {
    if( postTyp() == "home" ) {
        var elem = document.querySelector("#Blog1");
        elem.style.display = 'none';
        var elem = document.getElementsByClassName("blog-posts");
        for( var i = 0; i < elem.length; i++ ) {
            elem[i].style.display = 'none';
        }
    }
}

function setRedirectOnLang(Lang, rootSite) {
    var myLinks = document.getElementsByTagName('a');
    for (var i = 0; i < myLinks.length; i++) {
        var isHref = myLinks[i].href;
        if (isHref.indexOf(rootSite) > -1 && isHref.indexOf('#') == -1) {
            var trgLang = 'lang=' + Lang;
            if (isHref.indexOf("lang=") > -1) {
                var debLang = isHref.indexOf("lang=");
                var finLang = isHref.indexOf("lang=") + 7;
                var initLang = isHref.substring(debLang, finLang);
                isHref = isHref.replace(initLang, trgLang);
            }
        }
    }
}

```

```

        else if(isHref.indexOf("?") > -1) {
            isHref += '&' + trgLang;
        } else {
            isHref += '?' + trgLang;
        }
        myLinks[i].href = isHref;
    }
}

function setLangInscriptionTB(Lang, msg, rootSite, firstPostTittleLang) {
    var acc = document.getElementsByName('loc');
    var btInscrire = document.getElementsByClassName('subscribe-button');
    // On cache, pour l'instant, le bouton d'inscription à la lettre
d'information
    btInscrire[0].style.display = 'none';
    var setInscription = document.getElementsByClassName('subscribe-title');
    //var setSharingText = document.getElementsByClassName('platform-sharing-
text');
    //var setGetLinkText = document.getElementsByClassName('modal-dialog-
content');
    var setCallforEmail = document.getElementsByClassName('follow-by-email-
address');

    setRedirectOnLang(Lang, rootSite);

    var setRecMsg = document.getElementsByClassName('follow-by-email-submit');
    var setJumpLink = document.getElementsByClassName('jump-link');
    var setPopularPosts = document.getElementById('PopularPosts1');
    var setFeaturedPost = document.getElementById('FeaturedPost1');
    var setOtherPosts = document.getElementById('blog-pager');
    var setPostFilter = document.getElementsByClassName('post-filter-
message');
    var options = {
        year: "numeric",
        month: "long",
        day: "numeric"
    };
    var myDates = document.getElementsByTagName('Time');

    document.getElementsByClassName('header-widget')
[0].children[0].children[0].innerHTML = msg.header;
    btInscrire[0].innerHTML = msg.btInscrire;
    acc[0].value = msg.loc;
    setInscription[0].innerHTML = msg.setInscription;
    //setSharingText[0].innerHTML = msg.setSharingText;
    //setGetLinkText[0].innerHTML = msg.setGetLinkText;
    setCallforEmail[0].placeholder = msg.setCallforEmail;

```

```

setRecMsg[0].value = msg.setRecMsg;
for (var i = 0; i < setJumpLink.length; i++) {
    if (setJumpLink[i].children.length > 0) {
        setJumpLink[i].children[0].innerHTML = msg.setJumpLink;
    }
    else {
        setJumpLink[i].innerHTML = msg.setJumpLink;
    }
}
if (setPopularPosts != null) {
    setPopularPosts.children[0].innerHTML = msg.setPopularPosts;
}
if (setFeaturedPost != null) {
    setFeaturedPost.children[0].innerHTML = msg.setFeaturedPost;
}
if ( (setOtherPosts != null) && (setOtherPosts.children.length > 0) ) {
    setOtherPosts.children[0].innerHTML = msg.setOtherPosts;
}
if( setPostFilter.length > 0 ) {
    var innerHTML = setPostFilter[0].children[0].innerHTML;
    innerHTML = msg.setPostFilter1 + ' ' +
innerHTML.substring(innerHTML.indexOf("<"), innerHTML.length);
    setPostFilter[0].children[0].innerHTML = innerHTML;
    setPostFilter[0].children[1].children[0].innerHTML = msg.setPostFilter2;
}
for (var i = 0; i < myDates.length; i++) {
    var isDate = myDates[i].dateTime;
    var date = new Date(isDate);
    //var postLang = document.getElementsByClassName('post-lang');
    if ( (i != 0) || (firstPostTittleLang != ' ' ) ||
(window.location.pathname == "/search") ) {
        myDates[i].innerHTML = date.toLocaleString(msg.setLocforDates,
options);
    } else setLocalFlags();

}
document.getElementsByClassName('search-input')
[0].children[0].setAttribute("placeholder", msg.searchInput);
document.getElementsByClassName('search-input')
[0].children[0].setAttribute("id", 'searchText');
document.getElementsByClassName('search-input')
[0].children[0].setAttribute("data-lang", Lang);
document.getElementsByClassName('search-action')[0].value =
msg.searchAction;
document.getElementsByClassName('search-action')[0].setAttribute("id",
'searchButton');
document.getElementById("searchButton").onclick = doSearch;

```

```

var mySignatures = document.getElementById("signatures-country-name");
if( mySignatures !== 'undefined') mySignatures.setAttribute("data-lang",
Lang);

function doSearch() {
var searchText = document.getElementById("searchText");
var v = searchText.value;
var lang = searchText.getAttribute('data-lang')
//window.location = "search?q=" + v + "&lang=" + lang;
window.location.href =
"https://www.linternationaledessavoirspourtous.org/search?q=" + v + "&lang=" +
lang;
return false; // not entirely necessary, but just in case
}
}

function setPostTitleLang(msgPostTittle) {
var myPosts = document.getElementsByClassName('post-title');
var firstPostTitleLang = '';
for (var i = 0; i < myPosts.length; i++) {
var pathname = window.location.pathname;
var href;
if( postTyp() == "home" || i > 0 ) {
href = myPosts[i].children[0].href.split("?")[0];
} else {
href = window.location.href.split("?")[0];
}
var postTitle =
checkPostTitleLang(href, msgPostTittle);
if (postTitle != '') {
if( postTyp() == "home" || i > 0 ) {
myPosts[i].children[0].innerHTML = postTitle;
} else {
myPosts[0].innerHTML = postTitle;
firstPostTitleLang = postTitle;
}
}
}
return firstPostTitleLang;
}

function setDataForSummary(Lang, msgPostTittle) {
var divSocNet = document.getElementById('showBodyforSocialNetworks');
if (postTyp() != "home") {
divSocNet.style.display = "block";
var dataSocNet = document.getElementById('postBodyforSocialNetworks');
var myPosts = document.getElementsByClassName('post-title');
var path = window.location.href;

```



```

var href = path.split("?")[0];
dataSocNet.dataset.url = href;
//dataSocNet.dataset.title = myPosts[0].innerHTML;
dataSocNet.dataset.title = myPosts[0].innerText;
var postTitle =
    checkPostTitleLang(href, msgPostTittle);
if (postTitle != '') dataSocNet.dataset.title = postTitle;
var img = dataSocNet.getElementsByTagName("img");
if (img.length >= 1) {
    imgtag = img[0].src;
}
else {
    imgtag = 'https://2.bp.org/-
S2Q4C1j8nBM/XAf7ErxMHVI/AAAAAAAAiA4/00GFdrlKzdsdFL2DKDkFc176NLnqCMXCwCLcBGAs/s
320/Biblioteca_dell%2527Universita_di_Bologna.jpg';
}
dataSocNet.dataset.imgtag = imgtag;
if(dataSocNet.dataset.url.indexOf('?') > -1) {
    dataSocNet.dataset.url += '&lang=' + Lang;
} else {
    dataSocNet.dataset.url += '?lang=' + Lang;
}
var result = getLinkContent(dataSocNet.dataset.url);
//dataSocNet.innerHTML = result.body.replace('PopularPosts1','');
dataSocNet.dataset.innerHTML = result.body;
/*if( path.indexOf("&lngloc") == -1 ) {
    result = compSumAndImg(dataSocNet.innerHTML, summary_noimg, Lang,
'lang="", 'yes');
} else {
    var lngloc = path.substring(path.length - 5, path.length);
    result = compSumAndImg(dataSocNet.innerHTML, summary_noimg, lngloc,
'class="local-lang ', 'yes');
    var curPostTitle = document.getElementsByClassName("html2pdf-post-
title");
    //var forLength = Math.min(currentBtn.length, curPostTitle.length);
    for (var i = 0; i < curPostTitle.length; i++) {
        if (curPostTitle[i].className.indexOf(lngloc) > -1) {
            dataSocNet.dataset.title = curPostTitle[i].innerText;
            break;
        }
    }
}
result = compSumAndImg(dataSocNet.innerHTML, summary_noimg, Lang,
'lang="", 'yes');
var summary = result.summary;
dataSocNet.dataset.caption = summary.replace(/\&nbsp\;/g, "") + '...';
var img = result.img;
dataSocNet.dataset.imgtag = img;*/

```

```

        if( postTitle != '' ) {
            result = compSumAndImg(dataSocNet.dataset.innerHTML, summary_noimg,
Lang, 'lang="", 'yes');
            var summary = result.summary;
            dataSocNet.dataset.caption = summary.replace(/\&nbsp\;/g, "") +
'...';
            var img = result.img;
            dataSocNet.dataset.imgtag = img;
        }
        dataSocNet.dataset.lang = Lang;

        /*alert('FB Lang = ' + dataSocNet.dataset.lang
+ '\n url = ' + dataSocNet.dataset.url
+ '\n title = ' + dataSocNet.dataset.title
+ '\n caption = ' + dataSocNet.dataset.caption );*/
    }
}

```

```

function checkPostTitleLang(href, msgPostTittle) {
    var hrefBas = href;
    if( href.indexOf('#') > -1) hrefBas = href.split("#")[0];
    for (var i = 0; i < msgPostTittle.length; i++) {
        if (hrefBas == msgPostTittle[i].href) {
            return msgPostTittle[i].tittle;
        }
    }
    return "";
}

```

```

function postTyp() {
    //var pathname = window.location.origin + window.location.pathname;
    var pathname = window.location.pathname;
    if ((pathname == "/" ) || (pathname == "/search")) {
        return 'home';
    }
    if (pathname.indexOf("/p/") > -1) {
        return 'page';
    }
    return 'post';
}

```

```

function setDivWidth() {

    var contNaveFlags = document.getElementById("contNavetFlags");
    var contFlags = document.getElementById("Flags");
    var contNav = document.getElementById("navigationGB");

    if (window.location.pathname != '/') {

```

```

contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.style.display = "block";

contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.children[0].style.display = "inline-block";

contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.children[1].style.display = "inline-block";
    }

/*contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.style.width='98%';

contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.style.textAlign='left';*/

contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.style.width = '98%';

contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.style.textAlign = 'left';

/*contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.style.width='98%';

contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.style.textAlign='left';

    contNaveFlags.parentNode.parentNode.parentNode.parentNode.style.width='98%';
    contNaveFlags.parentNode.parentNode.parentNode.style.width='98%';*/
    contNaveFlags.parentNode.parentNode.style.width = '98%';

    //contNaveFlags.parentNode.parentNode.parentNode.style.background='#E7C296';

//contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.style.background='Teal';

contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.style.width = '100%';

//contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.children[0].style.background='Lime';

//contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.children[1].style.background='Olive';
    //contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.style.background='Silver';
    if (window.location.pathname != '/') contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.style.marginLeft = '-2em';
    contNaveFlags.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.parentNode.style.width = '105%';

```

```

//contFlags.style.background='Purple';
//contNav.style.background='Maroon';
if (sauvMenuBarTyp.DrpTyp == "false") {
    contNav.style.width = "77%";
    contFlags.style.width = "20fi%";
}
else {
    contNav.style.width = "65%";
    contFlags.style.width = "33%";
}

/*var widthContNaveFlags = contNaveFlags.offsetWidth;
var widthFlags = contFlags.offsetWidth;
var widthNavNum = (widthContNaveFlags - widthFlags)-10;
var widthNavStr = widthNavNum + "px";
document.getElementById("navigationGB").style.setProperty('width',
widthNavStr);
//document.getElementById("navigationGB").style.setProperty('width',
'1000px');
var widthNav = document.getElementById("navigationGB").style.width;
alert("widthContNaveFlags = " + widthContNaveFlags
+ "\n widthNavNum = " + widthNavNum
+ "\n widthNavStr = " + widthNavStr
+ "\n widthNav = " + widthNav
+ "\n widthFlags = " + widthFlags );*/

}

function setLangByTagNameTB(Lang, TagName) {
    var myElems = document.getElementsByTagName(TagName);
    for (var i=0; i < myElems.length; i++) {
        var isLang = myElems[i].hasAttribute("lang");
        if ( isLang ) {
            if ( myElems[i].lang == Lang ) {
                myElems[i].style.display = 'block';
            } else {
                myElems[i].style.display = 'none';
            }
        }
    }
}

}

function setNavigation() {

    var navID = navIDSauv;

    var newMenuBarTyp = { navID : 0, DrpTyp : "" }

```

```

newMenuBarTyp.navID = navID;
newMenuBarTyp.DrpTyp = setMenuBarType();

//if( (sauvMenuBarTyp.navID == newMenuBarTyp.navID)
//&& (sauvMenuBarTyp.DrpTyp == newMenuBarTyp.DrpTyp) ) return;
sauvMenuBarTyp.navID = newMenuBarTyp.navID;
sauvMenuBarTyp.DrpTyp = newMenuBarTyp.DrpTyp;

var navbar = document.getElementById("navigationGB");
var href;
var path = window.location.href;
var children = document.getElementById(navID).children;
var isMatched = false;
var isPlus = false;
for (var i = 0; i < children.length; i++) {
    var tableChild = children[i];
    var myA = tableChild.children[0];
    var href = myA.getAttribute("href");
    if (!myA.getAttribute('data-isplus')) {
        //if (href != "") {
            if (tableChild.children.length == 1) {
                if ( (path.split("?")[0] == href.split("?")[0]) ||
(path.split("search?")[0] == href.split("?")[0]) ) {
                    tableChild.setAttribute('class', 'active');
                    isMatched = true;
                }
            }
            else {
                tableChild.setAttribute('class', 'inactive');
            }
        }
        else {
            var isNewMatched = setMenu(tableChild, path);
            isMatched = isMatched || isNewMatched;
        }
        if (isPlus) myA.style.display = 'none';
        else myA.style.display = 'inline-block';
    }
    else {
        if (myA.dataset.isplus == "true") isPlus = true;
        if (isPlus) myA.parentNode.style.display = 'inline-block';
        else myA.parentNode.style.display = 'none';
    }
}
if (isMatched) {
    //tableChild.style.color = '#ffffff';
    navbar.style.display = "inline-block";
}
else {

```

```

        //tableChild.style.color = '#F4F43E';
    }

    function setMenu(tableChild, path) {
        var subchildren = tableChild.children[1].children;
        tableChild.setAttribute('class', 'inactive');
        var isMatched = false;
        for (var j = 0; j < subchildren.length; j++) {
            var tableSubChild = subchildren[j];
            var href = tableSubChild.children[0].getAttribute("href");
            //if (href != "") {
            if (tableSubChild.children.length == 1) {
                if ( (path.split("?")[0] == href.split("?")[0]) ||
(path.split("search?")[0] == href.split("?")[0]) ) {
                    tableSubChild.setAttribute('class', 'active');
                    isMatched = true;
                }
            }
            else {
                tableSubChild.setAttribute('class', 'inactive');
            }
        }
        else {
            var isNewMatched = setMenu(tableSubChild, path);
            isMatched = isMatched || isNewMatched;
            //tableSubChild.children[1].style.setProperty("margin-left", "171px");
            //tableSubChild.children[1].style.setProperty("margin-top", "-1.7em");
        }
    }
    if (isMatched) {
        tableChild.children[0].style.color = '#ffffff';
    }
    else {
        //tableChild.children[0].style.color = '#F4F43E';
    }
    return isMatched;
}

function setMenuBarType() {
    var w = window.innerWidth;
    /*if (window.innerWidth > 1600) {
        setDropdownGB(-6);
        //alert('windowWidth is 6 ' + window.innerWidth);
        return "false";
    } else if (window.innerWidth > 1450) {
        setDropdownGB(5);
        //alert('windowWidth is 5 ' + window.innerWidth);
        return "true";*/

```

```

if (window.innerWidth > 1450) {
    setDropdownGB(-5);
    //alert('windowWidth is 5 ' + window.innerWidth);
    return "false";
} else if (window.innerWidth > 1350) {
    setDropdownGB(4);
    //alert('windowWidth is 4 ' + window.innerWidth);
    return "true";
} else if (window.innerWidth > 1200) {
    setDropdownGB(3);
    //alert('windowWidth is 3 ' + window.innerWidth);
    return "true";
} else if (window.innerWidth > 1100) {
    setDropdownGB(2);
    //alert('windowWidth is 2 ' + window.innerWidth);
    return "true";
} else {
    setDropdownGB(1);
    //alert('windowWidth is 1 ' + window.innerWidth);
    return "true";
}
}

function getUrlVarsTB() {
    var vars = {};
    var parts = window.location.href.replace(/[?&]+([^\&]+)=([^\&]*)/gi,
function(m,key,value) {
        vars[key] = value;
    });
    return vars;
}

function getUrlParamTB(parameter, defaultvalue){
    var urlparameter = defaultvalue;
    if(window.location.href.indexOf(parameter) > -1){
        urlparameter = getUrlVarsTB()[parameter];
    }
    return urlparameter;
}

function callFBSDKinit() {
    window.fbAsyncInit = function() {
        FB.init({
            appId: '2258091117847688',
            //appId: '2198362993711491',
            autoLogAppEvents: true,
            xfbml: true,
            version: 'v3.2'
        });
    };
}

```

```

    });
};

(function(d, s, id) {
    var js, fjs = d.getElementsByTagName(s)[0];
    if (d.getElementById(id)) {
        return;
    }
    js = d.createElement(s);
    js.id = id;
    js.src = "https://connect.facebook.net/en_US/sdk.js";
    fjs.parentNode.insertBefore(js, fjs);
})(document, 'script', 'facebook-jssdk');
}

function fbSummarize() {
    var dataSocNet = document.getElementById('postBodyforSocialNetworks');
    var encodedURL = encodeURIComponent(dataSocNet.dataset.url + '&test');
    /*FB.ui({

        //display: 'popup',
        method: 'share_open_graph',
        action_type: 'og.shares',
        action_properties: JSON.stringify({
            object: {
                'og:url': encodedURL, // your url to share
                'og:title': 'test facebook', // dataSocNet.dataset.title,
                'og:description': 'ceci est un test...', //
dataSocNet.dataset.caption,
                'og:image': dataSocNet.dataset.imgtag,
                'fb:app_id': '2258091117847688'
            }
        })
    },
    // callback
    function(response) {
        if (response && !response.error_message) {
            alert('Posting completed.');
```



```

method: 'share_open_graph',
action_type: 'og.shares',
action_properties: JSON.stringify({
  object: {
    'og:url': encodedURL, // your url to share
    'og:title': dataSocNet.dataset.title,
    'og:description': dataSocNet.dataset.caption,
    'og:image': dataSocNet.dataset.imgtag,
  }
})

/*method: 'share',
href: dataSocNet.dataset.url,
message: 'getting educated about Facebook Connect',
caption: dataSocNet.dataset.caption,
picture: dataSocNet.dataset.imgtag,
redirect_uri: encodedURL*/

},
// callback
function(response) {
  if (response && !response.error_message) {
    //alert('Posting completed.');
```

```

function enDropdownGB() {
    var myNavigation = document.getElementById('ennavigation');
    showDropdownGB(myNavigation);
}

function itDropdownGB() {
    var myNavigation = document.getElementById('itnavigation');
    showDropdownGB(myNavigation);
}

function ptDropdownGB() {
    var myNavigation = document.getElementById('ptnavigation');
    showDropdownGB(myNavigation);
}

function esDropdownGB() {
    var myNavigation = document.getElementById('esnavigation');
    showDropdownGB(myNavigation);
}

function showDropdownGB(id) {

    //typeDropdownGB( true, 'navigationcss' );
    swtIdClass( id, 'navigationcss', 'dropdown-content' );

    typeDropdownGB( true, 'dropdown-content' );

}

function setDropdownGB(isDropdownGB) {

    var isPlus;
    if (isDropdownGB > 0) isPlus = isDropdownGB;
    else {
        isPlus = -isDropdownGB;
        stopDropdownGB();
    }

    var allDropDown = document.getElementsByClassName('navigationcss');
    for (var i = 0; i < allDropDown.length; i++) {
        var allLi = allDropDown[i].children;
        for (var j = 0; j < allLi.length; j++) {
            var allA = allLi[j].children;
            var myA = allA[0];
            if (myA.getAttribute('data-isplus')) {
                if (j < isPlus) {
                    for (var k = j; k < isPlus; k++) {

```

```

        var kk = k;
        allLi[kk + 1].parentNode.insertBefore(allLi[kk + 1], allLi[kk]);
    }
}
else if (j > isPlus) {
    for (var k = j; k > isPlus; k--) {
        var kk = k;
        allLi[kk].parentNode.insertBefore(allLi[kk], allLi[kk - 1]);
    }
}
}
}
if (typeof allLi[isPlus] !== 'undefined') {
    var myA = allLi[isPlus].getElementsByTagName('a')[0];
    if (isDropDownGB > 0) {
        myA.dataset.isplus = 'true';
    }
    else {
        myA.dataset.isplus = 'false';
    }
}
}
}
/*function setDropDownGB(isDropDownGB) {

    var isPlus;
    if (isDropDownGB > 0) isPlus = isDropDownGB;
    else {
        isPlus = -isDropDownGB;
        stopDropDownGB();
    }

    var allDropDown = document.getElementsByClassName('navigationcss');
    for (var i = 0; i < allDropDown.length; i++) {
        var allLi = allDropDown[i].getElementsByTagName('li');
        for (var j = 0; j < allLi.length; j++) {
            var allA = allLi[j].getElementsByTagName('a');
            var myA = allA[0];
            if (myA.getAttribute('data-isplus')) {
                if (j < isPlus) {
                    var k = j;
                    while(k < isPlus) {
                        var kk = k, kknew=k+1;
                        while(
!allLi[kk].parentNode.isEqualNode(allLi[kknew].parentNode)) kknew++;
                        allLi[kk].parentNode.insertBefore(allLi[kknew], allLi[kk]);
                        k = kknew;
                    }
                }
            }
        }
    }
}

```

```

    }
    else if (j > isPlus) {
        var k = j;
        while(k > isPlus) {
            var kk = k, kknew=k-1;
            while(
!allLi[kk].parentNode.isEqualNode(allLi[kknew].parentNode)) kknew--;
            allLi[kk].parentNode.insertBefore(allLi[kk], allLi[kknew]);
            k = kknew;
        }
    }
}
}
if (typeof allLi[isPlus] !== 'undefined') {
    var myA = allLi[isPlus].getElementsByTagName('a')[0];
    if (isDropDownGB > 0) {
        myA.dataset.isplus = 'true';
    }
    else {
        myA.dataset.isplus = 'false';
    }
}
}
}*/

```

```

function typeDropDownGB(fullNav, navClassName) {

    var allDropDown = document.getElementsByClassName(navClassName);
    for (var i = 0; i < allDropDown.length; i++) {
        var allLi = allDropDown[i].getElementsByTagName('li');
        var isPlus = false;
        for (var j = 0; j < allLi.length; j++) {
            if(allLi[j].children[0].style.color == 'rgb(255, 255, 255)') {
                allLi[j].children[0].style.color = 'red';
            }
            var allA = allLi[j].getElementsByTagName('a');
            for (var k = 0; k < allA.length; k++) {
                var myA = allA[k];
                if ( myA.dataset.isplus == "true" ) {
                    isPlus = true;
                    if ( fullNav ) myA.parentNode.style.display = 'none';
                    else myA.parentNode.style.display = 'inline-block';
                } else {
                    if( isPlus ) {
                        if ( fullNav ) myA.style.display = 'inline-block';
                        else myA.style.display = 'none';
                    }
                }
            }
        }
    }
}

```

```

    }
    }
}

function swtIdClass(d, oldClass, newClass) {
    //var d = document.getElementById(id);
    // removing class navigationcss of id='frnavigation'
    d.className = d.className.replace(' ' + oldClass, ''); //must keep a space
before class name
    // adding class dropdown-content of id='frnavigation'
    var myClassName = ' ' + newClass; //must keep a space before class name
    d.className = d.className.replace(myClassName, ""); // first remove the
class name if that already exists
    d.className = d.className + myClassName; // adding new class name
}

// When the user clicks anywhere outside of the modal menubar, close it
window.onclick = function(event) {
    var myNavigation;
    var myTarget = event.target;
    if (myTarget.dataset.isplus != "true") {
        stopDropDownGB();
    }
}

function stopDropDownGB() {
    var allDropDown = document.getElementsByClassName('dropdown-content');
    for (var i = 0; i < allDropDown.length; i++) {
        var isPlus = false;
        var allLi = allDropDown[i].getElementsByTagName('li');
        for (var j = 0; j < allLi.length; j++) {
            var allA = allLi[j].getElementsByTagName('a');
            for (var k = 0; k < allA.length; k++) {
                var myA = allA[k];
                if (myA.dataset.isplus == "true") {
                    isPlus = true;
                    myA.parentNode.style.display = 'inline-block';
                    var myNavigation = myA.parentNode.parentNode;
                }
                else {
                    if (isPlus) myA.style.display = 'none';
                }
            }
        }
        if (isPlus) swtIdClass(myNavigation, 'dropdown-content', 'navigationcss');
    }
}

```

```
// window.resize callback function
timeout = false, // holder for timeout id
delay = 250, // delay after event is "complete" to run callback
calls = 0;

function getDimensions() {
    calls += 1;
    alert("Width = " + window.innerWidth + "\n Height = " + window.innerHeight
+ "\n calls = " + calls )
}

// window.resize event listener
window.addEventListener('resize', function() {
    // clear the timeout
    clearTimeout(timeout);
    // start timing for event "completion"
    //timeout = setTimeout(getDimensions, delay);
    timeout = setTimeout(setNavigation, delay);
});

window.addEventListener('DOMContentLoaded', function(){

// Hamburger Menu is hidden
    var myHamburgerMenu = document.getElementsByClassName('hamburger-menu');
    if( myHamburgerMenu.length == 1) {
        myHamburgerMenu[0].style.display = "none";
    }

var myLang = getUrlParamTB('lang','fr');
    switch (myLang) {
        case "fr":
            frLangTB();
            break;
        case "en":
            enLangTB();
            break;
        case "de":
            deLangTB();
            break;
        case "it":
            itLangTB();
            break;
        case "pt":
            ptLangTB();
            break;
        case "es":
            esLangTB();
    }
}
```

```

        break;
    default:
        frLangTB();
    }
});

function setArticle(article, href, lang, title, authors, sumImg, sumContent) {

    var divArticle = document.getElementById(article);

    //var onclickTagname =
divArticle.children[0].children[0].children[0].children[0].tagName;
    var onclickTagname = divArticle.children[0].children[0].children[0].tagName;

    if (sumContent == '') {
        var div = document.getElementById("testTranslate");
        var result = getLinkContent(href);
        div.innerHTML = result.lang;
        lang = div.innerHTML;
        div.innerHTML = result.title;
        title = div.innerHTML;
        div.innerHTML = result.authors;
        authors = div.innerHTML;
        div.innerHTML = result.body;
        result = removeHtmlTagforAllLangs(div.innerHTML, 400, lang.split(' '),
'class="local-lang ', 'no');
        div.innerHTML = '';
        // remplace le code ASCII de l'espace insécable par son code HTML
        sumContent = result.summary.replace(/\&nbsp\;/g, "&#160;");
        sumImg = result.img;
    }

    if ((onclickTagname == "A")) {
        divArticle.children[0].children[0].children[0].href = href;

divArticle.children[0].children[0].children[0].setAttribute("onclick","return
" + gotoOnClick + "(this);");
    }
    //if ((onclickTagname == "A"))
divArticle.children[0].children[0].children[0].href = href + '?
lang=en&lngloc=fr-FR';
        divArticle.children[0].children[0].children[0].innerHTML = title;
        //divArticle.children[0].children[0].children[1].src =
getFlagfromLang(lang);
        divArticle.children[0].children[1].innerHTML = title;
        if (authors != "") divArticle.children[0].children[2].innerHTML = authors;

        var sumImgCont = divArticle.children[1].children[0].children[0].children[0];

```

```

if (sumImg == '') {
    sumImgCont.dataset.img = 'no';
}
else {
    sumImgCont.src = sumImg;
}
divArticle.children[1].children[0].children[1].innerHTML = sumContent;

divFlags = divArticle.children[0].children[0];
createFlags('', lang, divFlags, "setSumFlagsOnClick");

}

function gotoOnClick(e) {
    //alert(e.href + '&lngloc=' + e.getAttribute("data-lngloc"));
    window.location = e.href + '&lngloc=' + e.getAttribute("data-lngloc");
    return false;
}
function createFlags(lngloc, strLang, divFlag, foncOnClick) {
    strLang = strLang.split(' ');
    if(lngloc == '') lngloc = strLang[0];
    var elem = [];
    //for (i = 0; i < strLang.length; i++) {
    for( var i = strLang.length-1; i > -1; i-- ) {
        var src;
        switch (strLang[i]) {
            case "fr-FR":
                src = flagsRepository.France.img;
                break;
            case "en-EN":
                src = flagsRepository.RoyaumesUnis.img;
                break;
            case "en-US":
                src = flagsRepository.USA.img;
                break;
            case "de-DE":
                src = flagsRepository.Allemagne.img;
                break;
            case "it-IT":
                src = flagsRepository.Italie.img;
                break;
            case "pt-PT":
                src = flagsRepository.Portugal.img;
                break;
            case "es-ES":
                src = flagsRepository.Espagne.img;
                break;
        }
    }
}

```



```

    elem[i] = createFlag(src, strLang[i], divFlag, foncOnClick);
}
for( var i = strLang.length-1; i > -1; i-- ) {
    if ((lngloc == strLang[i]) && (foncOnClick != '')) {
        elem[i].click();
        break;
    }
}

function createFlag(src, lang, divFlag, foncOnClick) {
    var x = document.createElement("IMG");
    x.setAttribute("src", src);
    x.setAttribute("width", "32");
    x.setAttribute("height", "32");
    x.setAttribute("alt", lang);
    x.setAttribute("border", "0");
    x.setAttribute("style", "cursor:pointer; margin-left: 5px; float:
right;");
    if( foncOnClick != '' ) x.setAttribute("onclick", "return " + foncOnClick +
"(this);");
    x.setAttribute("class", "local-flags " + lang);

    divFlag.appendChild(x);
    return x;
}

function setLocalFlags() {
    var divFlag = document.getElementById("local-flags");
    var path = window.location.href;
    var isHref = document.getElementsByTagName("html2pdf-post-title");
    if( isHref.length == 0 ) return;
    var href = isHref[0].children[0].href;
    document.getElementById("contNavetFlags").style.display = 'none';
    var postLang = document.getElementsByClassName("post-lang")[0];
    var postLangHTML = postLang.innerHTML;
    var lngloc = '';
    if (path.indexOf('&lngloc=') > -1) {
        lngloc = path.split('&lngloc=')[1].substring(0,5);
    }
    else if (path.indexOf('?lang=') > -1) {
        var lang = path.split('?lang=')[1].substring(0,2);
        var strlang = postLangHTML;
        strlang = strlang.split(' ');
        for (var i = 0; i < strlang.length; i++) {
            if (lang == strlang[i].substring(0, 2)) {
                lngloc = strlang[i];
                break;
            }
        }
    }
}

```

```

    }
    }
}
createFlags(lngloc, postLangHTML, divFlag, "setLocalFlagsOnClick");
}

function setLocalFlagsOnClick(e) {
    var currentBtn = e.parentNode.children;
    for (var i = 0; i < currentBtn.length; i++) {
        currentBtn[i].classList.remove("flag-active");
    }
    var localLang = e.className.split("local-flags ")[1];
    e.classList.add("flag-active");
    var pathname = window.location.pathname;
    if (pathname != "/") {
        // traduction du titre
        var curPostTitle = document.getElementsByClassName("html2pdf-post-title");
        var forLength = Math.min(currentBtn.length, curPostTitle.length);
        for (var i = 0; i < forLength; i++) {
            if (forLength == 1 || curPostTitle[i].className.indexOf(localLang) > -1)
        {
            document.getElementsByClassName('post-title')[0].innerHTML =
curPostTitle[i].innerHTML;
            // update Facebook information
            var dataSocNet = document.getElementById('postBodyforSocialNetworks');
            dataSocNet.dataset.title = curPostTitle[i].innerText;
            result = compSumAndImg(dataSocNet.dataset.innerHTML, summary_noimg,
localLang, 'class="local-lang ', 'yes');
            var summary = result.summary;
            dataSocNet.dataset.caption = summary.replace(/\&nbsp\;/g, "") +
'...';
            var img = result.img;
            dataSocNet.dataset.imgtag = img;
            if (dataSocNet.dataset.url.indexOf('&lngloc=') == -1) {
                dataSocNet.dataset.url += '&lngloc=' + localLang;
            } else {
                var lngloc = dataSocNet.dataset.url.split('&lngloc=')
[1].substring(0,5);
                dataSocNet.dataset.url =
dataSocNet.dataset.url.replace(lngloc,localLang);
            }
            break;
        }
    }
}
// traduction des auteurs
var curPostAuth = document.getElementsByClassName("post-authors");
var forLength = Math.min(currentBtn.length, curPostAuth.length);

```

```

    for (var i = 0; i < forLength; i++) {
        if (curPostAuth[i].className.indexOf(localLang) > -1)
curPostAuth[i].style.display = 'block';
        else curPostAuth[i].style.display = 'none';
    }
    // traduction de la date document.getElementsByClassName("post-lang")
[0].className = 'lang-in-post post-lang ' + localLang;
    var myDates = document.getElementsByTagName('Time');
    var isDate = myDates[0].dateTime;
    var date = new Date(isDate);
    myDates[0].innerHTML = date.toLocaleString(localLang, {
        year: "numeric",
        month: "long",
        day: "numeric"
    });

    var currentDiv =
e.parentNode.parentNode.parentNode.getElementsByTagName("div");
    for (var i = 0; i < currentDiv.length; i++) {
        if (currentDiv[i].className.indexOf('local-lang') > -1) {
            currentDiv[i].style.display = 'none';
        }
    }
    for (var i = 0; i < currentDiv.length; i++) {
        if (currentDiv[i].className.indexOf('local-lang') > -1) {
            if (currentDiv[i].className.split(' ')[1] == localLang)
currentDiv[i].style.display = 'block ';
        }
    }

    var pdfRef = document.getElementById("pdf-ref");
    if (pdfRef != null) {
        var href = pdfRef.href;
        switch (localLang) {
            case "fr-FR":
                if ("fr" in pdfRef.dataset) href = pdfRef.dataset.fr;
                break;
            case "en-EN":
                if ("en" in pdfRef.dataset) href = pdfRef.dataset.en;
                break;
            case "en-US":
                if ("en" in pdfRef.dataset) href = pdfRef.dataset.en;
                break;
            case "de-DE":
                if ("de" in pdfRef.dataset) href = pdfRef.dataset.de;
                break;
            case "it-IT":
                if ("it" in pdfRef.dataset) href = pdfRef.dataset.it;

```

```

        break;
    case "pt-PT":
        if ("pt" in pdfRef.dataset) href = pdfRef.dataset.pt;
        break;
    case "es-ES":
        if ("es" in pdfRef.dataset) href = pdfRef.dataset.es;
        break;
    }
    pdfRef.href = href;
}
return false;
}

function setSumFlagsOnClick(e) {
    var localLang = e.className.split("local-flags ")[1];
    /*var href = e.parentNode.parentNode.children[0].href;
    href = href.substring(0, href.length - 5) + localLang;
    e.parentNode.parentNode.children[0].href = href;*/
    e.parentNode.children[0].setAttribute("data-lngloc", localLang);
    var currentDiv =
e.parentNode.parentNode.parentNode.getElementsByTagName("div");
    for (var i = 0; i < currentDiv.length; i++) {
        if (currentDiv[i].className.indexOf('local-lang') > -1) {
            currentDiv[i].style.display = 'none';
        }
    }
    for (var i = 0; i < currentDiv.length; i++) {
        if (currentDiv[i].className.indexOf('local-lang') > -1) {
            if (currentDiv[i].className.split(' ')[1] == localLang)
currentDiv[i].style.display = 'block ';
        }
    }
    return false;
}

function getFlagfromLang( lang ) {
    var flag;
    switch (lang) {
        case "fr-FR":
            flag = flagsRepository.France.img;
            break;
        case "en-EN":
            flag = flagsRepository.RoyaumesUnis.img;
            break;
        case "en-US":
            src = flagsRepository.USA.img;
            break;
        case "de-DE":

```

```

        flag = flagsRepository.Allemagne.img;
        break;
    case "it-IT":
        flag = flagsRepository.Italie.img;
        break;
    case "pt-PT":
        flag = flagsRepository.Portugal.img;
        break;
    case "es-ES":
        flag = flagsRepository.Espagne.img;
        break;
    default:
        flag = flagsRepository.France.img;
    }
    return flag;
}

function showSummary(el) {

    var onclickTagname = el.tagName;
    if (onclickTagname == "IMG") {
        var eyeOpen = "https://lh3.googleusercontent.com/-Wv-aBi5_sbW/XCvi-
zlsFiI/AAAAAAAAiHw/wmVW3KBeJE0_V0toQkUSqLqcb1F60nS6gCHMYCw/s0/eye-open.png";
        var eyeClosed = "https://lh3.googleusercontent.com/-
q3AVmPQD_gY/XCvbMfSSkjI/AAAAAAAAiHk/V_N9sg6HpNEUSfK_5YtSeDPcGfJpej0ggCHMYCw/s0
/eye-1103592_960_720.png";
        var divEye = el.parentNode;
        //var divTitle = divEye.parentNode;
        //var divArticle = divTitle.parentNode;
        var divArticle = divEye.parentNode;
        var divSum = divArticle.children[1];

        var spanImgLink = divSum.children[0].children[0];

        if (divSum.style.display == 'none') {
            el.src = eyeClosed;
            divSum.style.display = 'block';
            //divTitle.children[0].children[0].style.display = 'none';
            //divTitle.children[0].children[1].style.display = 'block';
            divEye.children[0].children[0].style.display = 'none';
            divEye.children[0].children[1].style.display = 'block';
            if (spanImgLink.children[0].dataset.img == 'no') {
                spanImgLink.style.display = 'none';
            }
        }
    }
    else {
        el.src = eyeOpen;
    }
}

```

```

        divSum.style.display = 'none';
        //divTitle.children[0].children[0].style.display = 'block';
        //divTitle.children[0].children[1].style.display = 'none';
        divEye.children[0].children[0].style.display = 'block';
        divEye.children[0].children[1].style.display = 'none';
    }
}
else {
    var eyeOpen = "https://lh3.googleusercontent.com/-Wv-aBi5_sbW/XCvi-
zlsFiI/AAAAAAAAiHw/wmVW3KBeJE0_V0toQkUSqLqcb1F60nS6gCHMYCw/s0/eye-open.png";
    var eyeClosed = "https://lh3.googleusercontent.com/-
q3AVmPQD_gY/XCvbMfSSkjI/AAAAAAAAiHk/V_N9sg6HpNEUSfK_5YtSeDPcGfJpej0ggCHMYCw/s0
/eye-1103592_960_720.png";
    var divLink = el.parentNode.parentNode;
    var divArticle = divLink.parentNode;
    var divSum = divArticle.children[1];

    var spanImgLink = divSum.children[0].children[0];

    if( divSum.style.display == 'none') {
        divSum.style.display = 'block';
        if( spanImgLink.children[0].dataset.img == 'no' ) {
            spanImgLink.style.display = 'none';
        }
    } else {
        divSum.style.display = 'none';
    }
}
}

function getLinkContent(theURL) {
    var xmlhttp = false;
    loadXMLDoc(theURL);
    if (xmlhttp == false) {
        //alert("timeout");
        return '';
    }
    else {
        var allText = xmlhttp.responseText;
        return getBody(allText);
    }
}

function getBody(content) {
    var élément; var result = {title: '', authors: '', body: ''};
    var élément = '// Fin des scripts, à ne pas supprimer (voir) getBody';
    content = getPart(content, élément, 1, 'no') ;
    élément = 'post-lang';
    result.lang = getPart(content, élément, 0, 'no').split("<")[0];
}

```

```

var langs = result.lang.split(' ');
/*élément = 'post-title entry-title';
result.title = getPart(content, élément, 0, 'no').split("<")[0];
élément = 'post-authors';
result.authors = getPart(content, élément, 0, 'yes').split("<")[0];*/
var title = '';
var authors = '';
if( langs.length == 1) {
    élément = 'post-title entry-title';
    title = getPart(content, élément, 0, 'no').split("<")[0];
    élément = 'post-authors';
    authors = getPart(content, élément, 0, 'no').split("<")[0];
} else {
    for( var i=0; i < langs.length; i++ ) {
        élément = 'html2pdf-post-title ' + langs[i];
        //title.push( getPart(content, élément, 0, 'no').split("<")
[1].split(">")[1] );
        title += '<div class="local-lang ' + langs[i] + '">' + getPart(content,
élément, 0, 'no').split("<")[1].split(">")[1] + '</div>';
        élément = 'authors-in-post post-authors ' + langs[i];
        //authors.push( getPart(content, élément, 0, 'no').split("<")[0] );
        authors += '<div class="local-lang ' + langs[i] + '">' +
getPart(content, élément, 0, 'no').split("<")[0] + '</div>';
    }
}
result.title = title;
result.authors = authors;
var élément = 'post-body entry-content float-container';
result.body = getPart(content, élément, 0, 'no') ;
return result;
}

function getPart(content, élément, iIndex, iShow) {
    var indices = [];
    var idx = content.indexOf(élément);
    while (idx != -1) {
        indices.push(idx);
        idx = content.indexOf(élément, idx + 1);
    }
    if( iShow == 'yes' ) {
        for (var i = 0; i < indices.length; i++) {
            var x = indices[i];
            x = content.indexOf(">", x + 1);
            var y = content.lastIndexOf("</body>");
            var z = content.slice(x + 1, y);
            alert('getBody for ' + élément + ' :\ni = ' + i + '\n' + z);
        }
    }
}

```

```

    var x = indices[iIndex];
    x = content.indexOf(">", x + 1);
    var y = content.lastIndexOf("</body>");
    return content.slice(x + 1, y);
}

function loadXMLDoc(theURL) {
    if (window.XMLHttpRequest) { // code for IE7+, Firefox, Chrome, Opera,
Safari, SeaMonkey
        xmlhttp = new XMLHttpRequest();
    }
    else { // code for IE6, IE5
        xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
    }
    xmlhttp.onreadystatechange = function() {
        if (xmlhttp.readyState == 4 && xmlhttp.status == 200) {
            //alert(xmlhttp.responseText);
        }
    }
    xmlhttp.open("GET", theURL, false);
    xmlhttp.send();
}

function removeHtmlTagforAllLangs(strxInput, chopInput, LangForSummary,
preLang, isFB) {
    var strxOutput = '';
    var img = '';
    for (var iLang = 0; iLang < LangForSummary.length; iLang++) {
        var Lang = LangForSummary[iLang];
        var resul = compSumAndImg(strxInput, chopInput, Lang, preLang, isFB);
        strxOutput += '<div ' + preLang + Lang + '>' + resul.summary + '...
</div>';
        if( img == '' ) img = resul.img;
    }
    var result = {
        img: img,
        summary: strxOutput
    };
    return result;
}

function compSumAndImg(strx, chop, Lang, preLang, isFB) {
    var result = {
        img: '',
        summary: ''
    };
    var strxOutput = '';

```



```

//for (var iLang = 0; iLang < LangForSummary.length; iLang++) {
//var Lang = LangForSummary[iLang];
var goOn = true;
/*var goOn = false;
if (Lang == '') {
    goOn = true;
}
else {
    var langHandler = document.getElementById(Lang + 'Handler');
    if (langHandler.style.display != "none") {
        goOn = true;
    }
}*/
if (goOn) {
    var chopIn = chop;
    var chopCount = 0;
    msg = '';
    if (strx.indexOf("<") != -1) {
        var s = strx.split("<");
        var news = [];
        var isToDelete = 0;
        var isFinished = false;
        for (var i = 0; i < s.length; i++) {
            if (result.img == '') {
                var myTag = s[i].substring(0, s[i].indexOf(">"));
                if ((myTag.indexOf('img') != -1) && (myTag.indexOf('src=') != -1)) {
                    var mySrc = myTag.split('src');
                    var myDelim = mySrc[1].substring(mySrc[1].indexOf("http") - 1,
mySrc[1].indexOf("http"));
                    var myHref = mySrc[1].split(myDelim);
                    result.img = myHref[1];
                }
            }
            var isdebDiv = debDiv(s[i]);
            if (isdebDiv) {
                var debToDelete = debDivToDelete(s[i], Lang, preLang);
                if ((isToDelete != 0) || debToDelete) {
                    isToDelete++;
                }
            }
            if (s[i].indexOf(">") != -1) {
                var isendDiv = endDiv(s[i]);
                if ((isToDelete > 0) && isendDiv) {
                    isToDelete--;
                }
                /*if ( !(isToDelete > 0) ) {
                    news.push(s[i].substring(s[i].indexOf(">") + 1, s[i].length));
                }*/

```

```

        if (!(isToDelete > 0)) {
            var curS = '';
            if (isFB == 'no' && isTagtoSum(s[i])) {
                //if(chopCount < chopIn) chop += s[i].indexOf(">") + 1
                curS = '<' + s[i];
                if (chopCount < chopIn) {
                    chop += s[i].indexOf(">") + 2;
                }
                else isFinished = true;
            }
            else {
                curS = s[i].substring(s[i].indexOf(">") + 1, s[i].length);
            }
            chopCount += (s[i].length - s[i].indexOf(">") - 1);
            // insert two line breaks after authors
            if ((i > 1) && (s[i - 2] == '/span>') && (s[i - 1] == '/strong>')
&& (s[i].indexOf('/p>') == 0)) {
                news.push('<br/>');
                chop += 5;
            }
        }
        news.push(curS);
    }
    //if( isFinished ) i = s.length;
}
strx = news.join("");

// begin of substitution of coded typo characters
//strx = strx.replace(/\&\#8217\;/g, "'");
//strx = strx.replace(/\&\#171\;/g, "«");
//strx = strx.replace(/\&\#187\;/g, "H");
//strx = strx.replace(/\&\#8239\;/g, " ");
//strx = strx.replace(/\&\#160\;/g, " ");
// end of substitution of coded typo characters

    if (msg != '') alert('news.length : ' + news.length + '\ns.length : ' +
s.length + '\nmsg : \n' + msg);
    chop = (chop < strx.length - 1) ? chop : strx.length - 1;
    if(chop != strx.length - 1) while (strx.charAt(chop) != ' ') chop--;
    strxOutput = strx.substring(0, chop);
    //alert('lang = ' + Lang + 'strx : \n' + strxOutput);
}
result.summary = strxOutput;
return result;
}

/*function SummaryOff(s) {

```

```

var myTag = s.substring(0, s.indexOf(">"));
if ((myTag.indexOf('div') > -1) && (myTag.indexOf('data-summary="no"') > -
1)) {
    return true;
}
return false;
}

function SummaryOn(s) {
    var myTag = s.substring(0, s.indexOf(">"));
    if ((myTag.indexOf('div') > -1) && (myTag.indexOf('data-summary="yes"') > -
1)) {
        return true;
    }
    return false;
}*/

function isTagtoSum(s) {
    if( (s[0] == 'a') || (s.indexOf('/a>') > -1) ) {
        return true;
    }
    if( s.indexOf('strong>') > -1 ) {
        return true;
    }
    if( s.indexOf('em>') > -1 ) {
        return true;
    }
    return false;
}

function debDivToDelete(s, Lang, preLang) {
    var myTag = s.substring(0, s.indexOf(">"));
    if ((myTag.indexOf('div') > -1) && (myTag.indexOf(preLang) > -1) && !
(myTag.indexOf(preLang + Lang + '"') > -1)) {
        return true;
    }
    if ((myTag.indexOf('div') > -1) && (myTag.indexOf('data-summary="no"') > -
1)) {
        return true;
    }
    if ( (myTag.indexOf('script') == 0) ) {
        return true;
    }
    if ( myTag.indexOf('style') == 0 ) {
        return true;
    }
    return false;
}

```

```

function debDiv(s) {
    var myTag = s.substring(0, s.indexOf(">"));
    if (myTag.indexOf('div') > -1 && !(myTag.indexOf('/') > -1)) {
        return true;
    }
    if ( (myTag.indexOf('script') == 0) && !(myTag.indexOf('/script') > -1) ||
(myTag[myTag.length-1] == '/')) {
        return true;
    }
    if ( (myTag.indexOf('style') == 0) && !(myTag.indexOf('/style') > -1) ||
(myTag[myTag.length-1] == '/')) {
        return true;
    }
    return false;
}

```

```

function endDiv(s) {
    var myTag = s.substring(0, s.indexOf(">"));
    if (myTag.indexOf('div') > -1 && myTag.indexOf('/') > -1) {
        return true;
    }
    if (myTag.indexOf('/script') == 0 ) {
        return true;
    }
    if (myTag.indexOf('/style') == 0 ) {
        return true;
    }
    return false;
}

```

```

function createSummaryAndThumb(pID, isRegular) {
    var div = document.getElementById(pID);
    var imgtag = "";
    var img = div.getElementsByTagName("img");
    var summ = summary_noimg;
    if (img.length >= 1) {
        if (isRegular) {
            //imgtag = '<span style="overflow:hidden; border: none; margin:0 15px
10px 0; float:left; max-width:' + img_thumb_width + 'px;height:auto;">
</span>';
            imgtag = '<span style="overflow:hidden; border: none; margin:0 15px 10px
0; float:left; max-width:' + img_thumb_width + 'px;height:auto;"></span>';
            //imgtag = '<span style="overflow:hidden; border: none; margin:0 15px
10px 0; float:left; max-height:' + img_thumb_height + 'px;width:auto;">

```

```

</span>';
    summ = summary_img_reg;
}
else {
    imgtag = '<center></center>';
    summ = summary_img_feat;
}
}
//var summary = imgtag + '<div>' + removeHtmlTagforAllLangs(div.innerHTML,
summ, LangForSummary) + '</div>';
var summary;
if (isRegular) summary = imgtag + '<div class="mySumReg">' +
removeHtmlTagforAllLangs(div.innerHTML, summ, LangForSummary, 'lang="",
'no').summary + '</div>';
else summary = imgtag + '<div class="mySumFeat">' +
removeHtmlTagforAllLangs(div.innerHTML, summ, LangForSummary, 'lang="",
'no').summary + '</div>';
div.innerHTML = summary;
}
// Fin des scripts, à ne pas supprimer (voir) getBody
</script>
<script>
// Ressources pour cartes LeafLet
var load = (function() {
    // Function which returns a function: https://davidwalsh.name/javascript-
functions
    function _load(tag) {
        return function(url) {
            // This promise will be used by Promise.all to determine success or
failure
            return new Promise(function(resolve, reject) {
                var element = document.createElement(tag);
                var parent = 'body';
                var attr = 'src';

                // Important success and error for the promise
                element.onload = function() {
                    resolve(url);
                };
                element.onerror = function() {
                    reject(url);
                };

                // Need to set different attributes depending on tag type
                switch (tag) {
                    case 'script':
                        element.async = true;

```

```

        break;
    case 'link':
        element.type = 'text/css';
        element.rel = 'stylesheet';
        attr = 'href';
        parent = 'head';
    }

    // Inject into document to kick off loading
    element[attr] = url;
    document[parent].appendChild(element);
});
};
}

return {
    css: _load('link'),
    js: _load('script'),
    img: _load('img')
}
})();
// Constructeur de la carte Leaflet
function loadMap(idMap) {
    L.map(idMap).remove();
    var bounds = new L.LatLngBounds(new L.LatLng(80, -180), new L.LatLng(-70,
200));
    var map = L.map(idMap, {
        scrollWheelZoom: false,
        maxBounds: bounds,
        maxBoundsViscosity: 1.0
    });

    map.setMinZoom( map.getBoundsZoom( map.options.maxBounds ) );

    map.createPane('labels');

    // This pane is above markers but below popups
    map.getPane('labels').style.zIndex = 650;

    // Layers in this pane are non-interactive and do not obscure mouse/touch
events
    map.getPane('labels').style.pointerEvents = 'none';

    /*var cartodbAttribution = '&copy; <a
href="https://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors,
&copy; <a href="http://cartodb.com/attributions">CartoDB</a>';

```

```

var positron =
L.tileLayer('http://{s}.basemaps.cartocdn.com/light_nolabels/{z}/{x}/{y}.png',
{
    attribution: cartodbAttribution
}).addTo(map);

var positronLabels =
L.tileLayer('http://{s}.basemaps.cartocdn.com/light_only_labels/{z}/{x}/{y}.png', {
    attribution: cartodbAttribution,
    pane: 'labels'
}).addTo(map);*/

return map;

};

function add2Layer(myLayer, inputCollection, setCallBack, iconSize) {

    // définition de la classe FlagIcon
    var FlagIcon = L.Icon.extend({
        options: {
            iconSize: iconSize, // size of the icon
            iconAnchor: [15, 15], // point of the icon which will correspond to
marker's location
            popupAnchor: [-3, -15] // point from which the popup should open
relative to the iconAnchor
        }
    });

    for (var i = 0; i < inputCollection.length; i++) {
        var input = inputCollection[i];

        Object.getOwnPropertyNames(myLayer).forEach(
            function(val, idx, array) {
                if (val == 'Links') {
                    // définition des lieux (icons)
                    var myIconLinks = new FlagIcon({
                        iconUrl: input.flag
                    });
                    // définition des markers associés au groupe météo des luttes
                    var markerLinks = L.marker([input.lat, input.lng], {
                        icon: myIconLinks
                    }).addTo(myLayer.Links);

                    markerLinks.on('mouseover', function(event) {
                        idHref(event);
                    });
                }
            }
        );
    }
}

```

```

    }
    //else if ( (val == 'options') || (val == 'PopUps')) {
    else {
        // définition des lieux (icons)
        var myIconPopUp = new FlagIcon({
            iconUrl: input.flag
        });
        // définition des markers associés au groupe météo des luttes
        var markerPopUp = L.marker([input.lat, input.lng], {
            icon: myIconPopUp
        });
        if (val == 'PopUps') markerPopUp.addTo(myLayer.PopUps);
        else markerPopUp.addTo(myLayer);

        markerPopUp.bindPopup(input.doc, {
            maxWidth: 600,
            minWidth: 250,
            maxHeight: 160,
            autoPan: true,
            closeButton: true,
            autoPanPadding: [5, 5]
        }).openPopup();

        markerPopUp.getPopup().on('remove', function() {
            setCallBack();
        });
    }
}
)
}

function idHref(event) {
    //var clickedMarker = event.layer;
    //alert(this.getLatLng());
    for (var i = 0; i < inputCollection.length; i++) {
        var input = inputCollection[i];
        if ((event.latlng.lng == input.lng) && (event.latlng.lat == input.lat))
location.href = '#' + input.id;
    }
}

};

function zoomRemove( map ) {
    //remove zoom functions
    map.removeControl(map.zoomControl);
    map.touchZoom.disable();
    map.doubleClickZoom.disable();
    map.scrollWheelZoom.disable();

```



```

map.boxZoom.disable();
map.keyboard.disable();
// Assumes your Leaflet map variable is 'map'..
L.DomUtil.addClass(map._container, 'default-cursor-enabled');
}

// Copyright (c) 2013 Ryan Clark
// https://gist.github.com/rclark/5779673
L.TopoJSON = L.GeoJSON.extend({
  addData: function(jsonData) {
    if (jsonData.type === "Topology") {
      for (key in jsonData.objects) {
        geojson = topojson.feature(jsonData, jsonData.objects[key]);
        L.GeoJSON.prototype.addData.call(this, geojson);
      }
    }
    else {
      L.GeoJSON.prototype.addData.call(this, jsonData);
    }
  }
});

// Constructeur de la carte TopoJSON avec pays pour Leaflet
function loadMapTopoJSON(map, colorDeb, colorFin, colorStyle) {
  // Usage: Load different file types with one callback
  Promise.all([
    //load.css('lib/highlighter.css'),
    //load.img('images/logo.png')
    //load.js('lib/main.js'),

    // Début ressources utilisées pour construire la carte mondiale colorée
    TopoJSON
    // ressources chroma.js utilisées pour construire la carte TopoJSON
    colorée
    //load.js('https://drive.google.com/uc?
export=download&id=1vAhVcT9intE5QEkcuu3UQOBEutYjKtaO'),
    load.js('https://philquand.github.io/JavaScript/Leaflet/chroma.min.js'),
    // ressources utilisées pour construire la carte TopoJSON
    load.js('https://d3js.org/topojson.v1.min.js'),
    // données de la carte TopoJSON
    //load.js('https://drive.google.com/uc?
export=download&id=1WZyHrCgJmR3mCAdQMhicqT0yjkASiJpE')

    load.js('https://philquand.github.io/JavaScript/Leaflet/countries.topo.json')
    // Fint ressources utilisées pour construire la carte mondiale colorée
    TopoJSON

  ]).then(function() {

```

```

console.log('Everything has loaded!');

topoLayer = new L.TopoJSON(),
  colorScale = chroma
  //.scale(['#D5E3FF', '#003171'])
  .scale([colorDeb, colorFin])
  .domain([0, 1]);

addTopoData(worldCountries);

function addTopoData(topoData) {
  topoLayer.addData(topoData);
  topoLayer.addTo(map);
  topoLayer.eachLayer(handleLayer);
}

function handleLayer(layer) {
  var randomValue = Math.random(),
      fillColor = colorScale(randomValue).hex();

  layer.setStyle({
    fillColor: fillColor,
    fillOpacity: 1,
    color: colorStyle,
    weight: 1,
    opacity: .5
  });
}
}).catch(function() {
  console.log('Oh no, epic failure!');
});

// Constructeur de la planisphère openstreetmap pour Leaflet
function loadOpenStreetWorld(map) {

  var cartodbAttribution = '&copy; <a
href="https://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors,
&copy; <a href="http://cartodb.com/attributions">CartoDB</a>';

  var positron =
L.tileLayer('http://{s}.basemaps.cartocdn.com/light_nolabels/{z}/{x}/{y}.png',
{
  attribution: cartodbAttribution
}).addTo(map);

  var positronLabels =

```

```

L.tileLayer('http://{s}.basemaps.cartocdn.com/light_only_labels/{z}/{x}/{y}.png', {
  attribution: cartodbAttribution,
  pane: 'labels'
}).addTo(map);

};

// Constructeur de la carte Mapbox pour Leaflet
function loadMapBox(map) {
  var mapboxAccessToken =
'pk.eyJ1IjoicGhpbnBF1YW5kIiwiaSI6ImNqcmZlOWFwMjBuMnQ0NWlqN3VhczNhMW4ifQ.OSyAGC_
JQQUISRiSuc8ghg';

  L.tileLayer('https://api.tiles.mapbox.com/v4/{id}/{z}/{x}/{y}.png?
access_token={accessToken}', {
    attribution: 'Map data &copy; <a
href="https://www.openstreetmap.org/">OpenStreetMap</a> contributors, <a
href="https://creativecommons.org/licenses/by-sa/2.0/">CC-BY-SA</a>, Imagery ©
<a href="https://www.mapbox.com/">Mapbox</a>',
    maxZoom: 18,
    id: 'mapbox.streets',
    //id: 'mapbox.light',
    accessToken: mapboxAccessToken
  }).addTo(map);

};

// Constructeur de la carte GeoJSON des régions françaises pour Leaflet
function loadRegFranceJSONN(map, getColor) {
  // Usage: Load different file types with one callback
  Promise.all([
    // GeoJSON des régions françaises
    //load.js('https://drive.google.com/uc?
export=download&id=1hrJGwENimzW4gmf812yAZQk1z1KivZSU')

load.js('https://philquand.github.io/Javascript/Leaflet/regionsFranceData.js')
  ]).then(function() {
    console.log('Everything has loaded!');

    function styleReg(feature) {
      return {
        fillColor: getColor(feature.properties.nom),
        weight: 2,
        opacity: 1,
        color: 'white',
        fillOpacity: 0.7
      };
    };
  });

```

```

    }

    L.geoJson(regionsFranceData, {
        style: styleReg
    }).addTo(map);

}).catch(function() {
    console.log('Oh no, epic failure!');
});

};

// Constructeur de la carte GeoJSON des départements françaises pour Leaflet
function loadDeptFranceJSONN(map, getColor, getFilter) {
    // Usage: Load different file types with one callback
    Promise.all([
        // GeoJSON des régions françaises
        //load.js('https://drive.google.com/uc?
export=download&id=1D_ajtPoOkMWqaz8o8LkKPYjRd-LuxjJw')

load.js('https://philquand.github.io/Javascript/Leaflet/deptsFranceData.js')
    ]).then(function() {
        console.log('Everything has been loaded!');

        function styleReg(feature) {
            return {
                fillColor: getColor(feature.properties.nom),
                weight: 2,
                opacity: 1,
                color: 'white',
                fillOpacity: 0.5
            };
        }

        function filterReg(feature, layer) {
            return getFilter(feature.properties.nom);
        }

        L.geoJson(deptsFranceData, {
            style: styleReg,
            filter: filterReg
        }).addTo(map);

    }).catch(function() {
        console.log('Oh no, epic failure!');
    });

};

// Constructeur de la carte TopoJSON des signatures de l'Internationale des
Savoirs pour Tous pour Leaflet

```

```

function loadMapTopoJSONSignatures(map, grades, colorDeb, colorFin,
colorStyle, divCountryName, tableSignatures, setCallBackLarge,
setCallBackSmall) {
    var lang = divCountryName.getAttribute("data-lang");
    var divFR = document.createElement("div"); divFR.lang = 'fr';
    var divEN = document.createElement("div"); divEN.lang = 'en';
    var divDE = document.createElement("div"); divDE.lang = 'de';
    var divES = document.createElement("div"); divES.lang = 'es';
    var divIT = document.createElement("div"); divIT.lang = 'it';
    var divPT = document.createElement("div"); divPT.lang = 'pt';
    divCountryName.appendChild(divFR);
    divCountryName.appendChild(divEN);
    divCountryName.appendChild(divDE);
    divCountryName.appendChild(divES);
    divCountryName.appendChild(divIT);
    divCountryName.appendChild(divPT);
    var allDiv = { divFR : divFR, divEN : divEN, divDE : divDE, divES : divES,
divIT : divIT, divPT : divPT };
    // Usage: Load different file types with one callback
    Promise.all([
        // ressources TableTop pour lire les données Google Sheet utilisées pour
construire la carte TopoJSON colorée
        //load.js('https://drive.google.com/uc?
export=download&id=1wYZfkDO3ecEYW3kCVWbWk6W1gWWg0HnZ'),
        load.js('https://philquand.github.io/JavaScript/tabletop.min.js'),

//load.js('https://cdnjs.cloudflare.com/ajax/libs/tabletop.js/1.5.1/tabletop.m
in.js'),
        // Début ressources utilisées pour construire la carte mondiale colorée
TopoJSON
        // ressources chroma.js utilisées pour construire la carte TopoJSON
colorée
        //load.js('https://drive.google.com/uc?
export=download&id=1vAhVcT9intE5QEkcuu3UQOBEutYjKtaO'),
        load.js('https://philquand.github.io/JavaScript/Leaflet/chroma.min.js'),
        // ressources utilisées pour construire la carte TopoJSON
        load.js('https://d3js.org/topojson.v1.min.js'),
        // données de la carte TopoJSON
        //load.js('https://drive.google.com/uc?
export=download&id=1WZyHrCgJmR3mCAdQMhicqT0yjkASiJpE')

load.js('https://philquand.github.io/JavaScript/Leaflet/countries.topo.json'),
        // Fint ressources utilisées pour construire la carte mondiale colorée
TopoJSON

    ]).then(function() {
        console.log('Everything has loaded!');
    });
}

```

```

var code = "lw6dczYR7VN2YBkx8t5AHdxRzoEv6eozL6qERuvnOwWM";
//var countries, map2dataCountries;
// loop through spreadsheet with Tabletop
Tabletop.init({
  key: code,
  callback: function(sheet, tabletop) {

    var tableInfos = tableSignatures;
    if (typeof tableInfos !== 'undefined') writeTables(sheet, tableInfos);

    function writeTables(sheet, tableInfos) {

      if (tableInfos.divFR !== '') {
        var table = [];
        //table.push(sheet.français.column_names);
        table.push(['Nom, prénom', 'Activité', 'Pays']);
        for (i = 0; i < sheet.français.elements.length; i++) {
          var rowTable = [];
          rowTable.push(sheet.français.elements[i].nomprénom);
          rowTable.push(sheet.français.elements[i].activité);
          rowTable.push(sheet.français.elements[i].pays);
          table.push(rowTable);
        }
        var tableID = document.getElementById(tableInfos.divFR);
        writeTable(tableID, table, 'tableSignatures');
      }

      if (tableInfos.divEN !== '') {
        var table = [];
        //table.push(sheet.anglais.column_names);
        table.push(['Last name, first name', 'Activity', 'Country']);
        for (i = 0; i < sheet.anglais.elements.length; i++) {
          var rowTable = [];
          rowTable.push(sheet.anglais.elements[i].lastnamefirstname);
          rowTable.push(sheet.anglais.elements[i].activity);
          rowTable.push(sheet.anglais.elements[i].country);
          table.push(rowTable);
        }
        var tableID = document.getElementById(tableInfos.divEN);
        writeTable(tableID, table, 'tableSignatures');
      }

      if (tableInfos.divDE !== '') {
        var table = [];
        //table.push(sheet.allemand.column_names);
        table.push(['Name, vorname', 'Beruf', 'Land']);
        for (i = 0; i < sheet.allemand.elements.length; i++) {
          var rowTable = [];

```

```

        rowTable.push(sheet.allemand.elements[i].namevorname);
        rowTable.push(sheet.allemand.elements[i].beruf);
        rowTable.push(sheet.allemand.elements[i].land);
        table.push(rowTable);
    }
    var tableID = document.getElementById(tableInfos.divDE);
    writeTable(tableID, table, 'tableSignatures');
}

if (tableInfos.divPT != '') {
    var table = [];
    //table.push(sheet.portugais.column_names);
    table.push(['Sobrenome, nome', 'Actividade', 'País']);
    for (i = 0; i < sheet.portugais.elements.length; i++) {
        var rowTable = [];
        rowTable.push(sheet.portugais.elements[i].sobrenomenome);
        rowTable.push(sheet.portugais.elements[i].atividade);
        rowTable.push(sheet.portugais.elements[i].país);
        table.push(rowTable);
    }
    var tableID = document.getElementById(tableInfos.divPT);
    writeTable(tableID, table, 'tableSignatures');
}

if (tableInfos.divES != '') {
    var table = [];
    //table.push(sheet.espanhol.column_names);
    table.push(['Apellido, nombre', 'Actividad', 'País']);
    for (i = 0; i < sheet.espanhol.elements.length; i++) {
        var rowTable = [];
        rowTable.push(sheet.espanhol.elements[i].apellidonombre);
        rowTable.push(sheet.espanhol.elements[i].actividad);
        rowTable.push(sheet.espanhol.elements[i].país);
        table.push(rowTable);
    }
    var tableID = document.getElementById(tableInfos.divES);
    writeTable(tableID, table, 'tableSignatures');
}

if (tableInfos.divIT != '') {
    var table = [];
    //table.push(sheet.italien.column_names);
    table.push(['Cognome, nome', 'Attività', 'Paese']);
    for (i = 0; i < sheet.italien.elements.length; i++) {
        var rowTable = [];
        rowTable.push(sheet.italien.elements[i].cognomenome);
        rowTable.push(sheet.italien.elements[i].attività);
        rowTable.push(sheet.italien.elements[i].paese);
    }
}

```

```

        table.push(rowTable);
    }
    var tableID = document.getElementById(tableInfos.divIT);
    writeTable(tableID, table, 'tableSignatures');
}
}

function writeTable(parent, data, classTable) {
    if (parent.children.length > 0) deleteChild(parent);
    var TBthead = document.createElement("table");
    TBthead.classList.add(classTable);
    parent.appendChild(TBthead);
    var thead = document.createElement("thead");
    TBthead.appendChild(thead);
    drawRowth(thead, data[0]);

    if(data.length > 1 ) {
        var TBtbody = document.createElement("table");
        TBtbody.classList.add(classTable);
        parent.appendChild(TBtbody);
        var tbody = document.createElement("tbody");
        TBtbody.appendChild(tbody);

        for (var r = 1; r < data.length; r++) {
            drawRow(tbody, data[r]);
        }
    }

    function drawRow(table, rowData) {
        if (rowData == null) return null;
        if (rowData.length == 0) return null;
        //var tr = $("<tr style='width: 100%;' />");
        var tr = document.createElement("tr");
        table.appendChild(tr);
        for (var c = 0; c < rowData.length; c++) {
            drawCell(tr, rowData[c]);
        }
        return tr;

        function drawCell(tr, val) {
            var td = document.createElement("td");
            tr.appendChild(td);
            td.innerHTML = val;
            return td;
        }
    }

    function drawRowth(table, rowData) {
        if (rowData == null) return null;

```



```

        if (rowData.length == 0) return null;
        for (var c = 0; c < rowData.length; c++) {
            drawCellth(table, rowData[c]);
        }
        return table;

        function drawCellth(table, val) {
            var th = document.createElement("th");
            table.appendChild(th);
            th.innerHTML = val;
            return th;
        }
    }
}

function deleteChild(parent) {
    if(parent.children.length == 0) return;
    var child = parent.lastElementChild;
    while (child) {
        parent.removeChild(child);
        child = parent.lastElementChild;
    }
}

getSigCountFromSheet(sheet);
var countries = getCountriesFromSheet(sheet);
var map2dataCountries = getmap2dataCountriesFromSheet(sheet);

function thisColor(d) {
    var nbMax = grades.length;
    for (var i = nbMax-1; i >= 0; i--) {
        if (d > grades[i]) return colorScale(i / nbMax).hex();
    }
    return 'white';
}

function getSigCountFromSheet(sheet) {
    var sigCount = sheet.français.elements.length;
    var lang = divCountryName.getAttribute("data-lang");
    var btnSignatures =
document.getElementsByTagName("butSignatures");
    if( btnSignatures.length == 6) {
        btnSignatures[0].innerHTML = "Voir les " + sigCount + "
signataires"
        btnSignatures[1].innerHTML = "See the " + sigCount + "
signatories"
        btnSignatures[2].innerHTML = "Die " + sigCount + " Unterzeichner"
        btnSignatures[3].innerHTML = "Vedere i " + sigCount + "
firmatari"
        btnSignatures[4].innerHTML = "Ver os " + sigCount + "

```

```

signatários"
        btnSignatures[5].innerHTML = "Ver a los " + sigCount + "
signatarios"
    };
}

function getCountriesFromSheet(sheet) {
    var countries = [];
    for (var i in sheet.français.elements) {
        var data = sheet.français.elements[i];
        var idx = getCountryIndex(data.pays, countries);
        if (idx == -1) {
            countries.push({});
            idx = countries.length - 1;
            countries[idx].signatures = [];
        }
        countries[idx].name = data.pays;
        var jdx = getSignatureIndex(data.nomprénom,
countries[idx].signatures);
        if (jdx == -1) {
            countries[idx].signatures.push({});
            jdx = countries[idx].signatures.length - 1;
        }
        countries[idx].signatures[jdx].nomprénom = data.nomprénom;
        countries[idx].signatures[jdx].fract = data.activité;
    }
    for (var i in sheet.anglais.elements) {
        var data = sheet.anglais.elements[i];
        var idx = getCountryIndex(data.country, countries);
        if (idx == -1) {
            countries.push({});
            idx = countries.length - 1;
            countries[idx].signatures = [];
        }
        countries[idx].name = data.country;
        var jdx = getSignatureIndex(data.lastnamefirstname,
countries[idx].signatures);
        if (jdx == -1) {
            countries[idx].signatures.push({});
            jdx = countries[idx].signatures.length - 1;
        }
        countries[idx].signatures[jdx].nomprénom =
data.lastnamefirstname;
        countries[idx].signatures[jdx].enact = data.activity;
    }
    for (var i in sheet.allemand.elements) {
        var data = sheet.allemand.elements[i];
        var idx = getCountryIndex(data.land, countries);

```

```

        if (idx == -1) {
            countries.push({});
            idx = countries.length - 1;
            countries[idx].signatures = [];
        }
        countries[idx].name = data.land;
        var jdx = getSignatureIndex(data.namevorname,
countries[idx].signatures);
        if (jdx == -1) {
            countries[idx].signatures.push({});
            jdx = countries[idx].signatures.length - 1;
        }
        countries[idx].signatures[jdx].nomprénom = data.namevorname;
        countries[idx].signatures[jdx].deact = data.beruf;
    }
    for (var i in sheet.portugais.elements) {
        var data = sheet.portugais.elements[i];
        var idx = getCountryIndex(data.país, countries);
        if (idx == -1) {
            countries.push({});
            idx = countries.length - 1;
            countries[idx].signatures = [];
        }
        countries[idx].name = data.país;
        var jdx = getSignatureIndex(data.sobrenomenome,
countries[idx].signatures);
        if (jdx == -1) {
            countries[idx].signatures.push({});
            jdx = countries[idx].signatures.length - 1;
        }
        countries[idx].signatures[jdx].nomprénom = data.sobrenomenome;
        countries[idx].signatures[jdx].ptact = data.actividade;
    }
    for (var i in sheet.espanhol.elements) {
        var data = sheet.espanhol.elements[i];
        var idx = getCountryIndex(data.país, countries);
        if (idx == -1) {
            countries.push({});
            idx = countries.length - 1;
            countries[idx].signatures = [];
        }
        countries[idx].name = data.país;
        var jdx = getSignatureIndex(data.apellidonombre,
countries[idx].signatures);
        if (jdx == -1) {
            countries[idx].signatures.push({});
            jdx = countries[idx].signatures.length - 1;
        }
    }

```

```

        countries[idx].signatures[jdx].nomprénom = data.apellidonombre;
        countries[idx].signatures[jdx].esact = data.actividad;
    }
    for (var i in sheet.italien.elements) {
        var data = sheet.italien.elements[i];
        var idx = getCountryIndex(data.paese, countries);
        if (idx == -1) {
            countries.push({});
            idx = countries.length - 1;
            countries[idx].signatures = [];
        }
        countries[idx].name = data.paese;
        var jdx = getSignatureIndex(data.cognomenome,
countries[idx].signatures);
        if (jdx == -1) {
            countries[idx].signatures.push({});
            jdx = countries[idx].signatures.length - 1;
        }
        countries[idx].signatures[jdx].nomprénom = data.cognomenome;
        countries[idx].signatures[jdx].itact = data.attività;
    }
    return countries;
}

function getmap2dataCountriesFromSheet(sheet) {
    var map2dataCountries = [];
    for (var i in sheet.BackOffice.elements) {
        var data = sheet.BackOffice.elements[i];
        map2dataCountries.push({
            map: data.topojsoncountry,
            data: data.datacountry
        })
    }
    return map2dataCountries;
}

function getCountryIndex(country, countries) {
    for (var idx in countries) {
        if (country == countries[idx].name) return idx;
    }
    return -1;
}

function getSignatureIndex(signature, signatures) {
    for (var idx in signatures) {
        if (signature == signatures[idx].nomprénom) return idx;
    }
    return -1;
}

```

```

    }

    var tableID = document.getElementById(tableInfos.divFR);
    topoLayer = new L.TopoJSON();
    //$("#" + idCountryName) = $(". " + idCountryName),
    colorScale = chroma
    //.scale(['#D5E3FF', '#003171'])
    .scale([colorDeb, colorFin])
    .domain([0, 1]);

    addTopoData(worldCountries);

    function addTopoData(topoData) {
        topoLayer.addData(topoData);
        topoLayer.addTo(map);
        topoLayer.eachLayer(handleLayer);
    }

    function addLegend() {
        var legend = L.control({
            position: 'bottomright'
        });

        legend.onAdd = function(map) {

            var div = L.DomUtil.create('div', 'info legend'),
                labels = [];

            // loop through our density intervals and generate a label with a
            colored square for each interval
            for (var i = grades.length - 1; i >= 0; i--) {
                //for (var i = 0; i < grades.length; i++) {
                div.innerHTML +=
                    '<i style="background:' + thisColor(grades[i] + 1) + '"></i>' +
                    grades[i] + (grades[i + 1] ? '&ndash;' + grades[i + 1] +
                    '<br>' : '+<br>');
                }

            return div;
        };

        legend.addTo(map);
    }

    function handleLayer(layer) {
        //var randomValue = Math.random(),

```

```

var fillColor = getColor();
layer.setStyle({
    fillColor: fillColor,
    //fillColor: getColor,
    fillOpacity: 1,
    color: colorStyle,
    weight: 1,
    opacity: .5
});

if ("ontouchstart" in document.documentElement) {
    layer.on({
        //dblclick: clickLayer,
        click: touchLayer,
    });
} else {
    layer.on({
        click: clickLayer,
        mouseover: enterLayer,
        mouseout: leaveLayer
    });
}

function getColor() {
    var countryName = layer.feature.properties.name;
    var d = 0;
    for (var idx in countries) {
        if (countryName == countries[idx].name) {
            d += countries[idx].signatures.length;
        }
        for (var jdx in map2dataCountries) {
            var map2data = map2dataCountries[jdx];
            if ((countryName == map2data.map) && (countries[idx].name ==
map2data.data)) {
                d += countries[idx].signatures.length;
            }
        }
    }
    return thisColor(d);
}

ej function getLang() {
    var mySignatures = document.getElementById("signatures-country-
name");

    if( mySignatures !== 'undefined') return
mySignatures.getAttribute("data-lang");
    else return '';
}

```

```

    }

    function touchLayer() {
        if (divCountryName == null) return
        var countryName = this.feature.properties.name;
        var dataCountryView = divCountryName.getAttribute("data-country-
view");

        //if(dataCountryView == 'openPopup' || dataCountryView ==
countryName) return;
        if (dataCountryView == countryName) return;
        //var signatures = '';
        var signaturesFR = ['title'];
        var signaturesEN = ['title'];
        var signaturesDE = ['title'];
        var signaturesPT = ['title'];
        var signaturesES = ['title'];
        var signaturesIT = ['title'];
        var sigcount = 0;
        var lang = divCountryName.getAttribute("data-lang");
        allDiv.divFR.className = 'signatures-country-name-wrapper';
        allDiv.divEN.className = 'signatures-country-name-wrapper';
        allDiv.divDE.className = 'signatures-country-name-wrapper';
        allDiv.divES.className = 'signatures-country-name-wrapper';
        allDiv.divIT.className = 'signatures-country-name-wrapper';
        allDiv.divPT.className = 'signatures-country-name-wrapper';
        allDiv.divFR.style.display = 'none';
        allDiv.divEN.style.display = 'none';
        allDiv.divDE.style.display = 'none';
        allDiv.divES.style.display = 'none';
        allDiv.divIT.style.display = 'none';
        allDiv.divPT.style.display = 'none';
        if (lang == 'fr') allDiv.divFR.style.display = 'inline-block';
        else if (lang == 'en') allDiv.divEN.style.display = 'inline-
block';
        else if (lang == 'de') allDiv.divDE.style.display = 'inline-
block';
        else if (lang == 'es') allDiv.divES.style.display = 'inline-
block';
        else if (lang == 'it') allDiv.divIT.style.display = 'inline-
block';
        else if (lang == 'pt') allDiv.divPT.style.display = 'inline-
block';

        for (var idx in countries) {
            if (countryName == countries[idx].name) {
                for (var i in countries[idx].signatures) {
                    var data = countries[idx].signatures[i];
                    signaturesFR.push([data.nomprénom, data.fract]);
                    signaturesEN.push([data.nomprénom, data.enact]);
                }
            }
        }
    }

```

```

        signaturesDE.push([data.nomprénom, data.deact]);
        signaturesPT.push([data.nomprénom, data.ptact]);
        signaturesES.push([data.nomprénom, data.esact]);
        signaturesIT.push([data.nomprénom, data.itact]);
    }
    sigcount += countries[idx].signatures.length;
}
for (var jdx in map2dataCountries) {
    var map2data = map2dataCountries[jdx];
    if ((countryName == map2data.map) && (countries[idx].name ==
map2data.data)) {
        for (var i in countries[idx].signatures) {
            var data = countries[idx].signatures[i];
            signaturesFR.push([data.nomprénom, data.fract]);
            signaturesEN.push([data.nomprénom, data.enact]);
            signaturesDE.push([data.nomprénom, data.deact]);
            signaturesPT.push([data.nomprénom, data.ptact]);
            signaturesES.push([data.nomprénom, data.esact]);
            signaturesIT.push([data.nomprénom, data.itact]);
        }
        sigcount += countries[idx].signatures.length;
    }
}
}
//if (signatures != '') {
if (sigcount > 0) {
    divCountryName.setAttribute("data-country", "");
    //divCountryName.parentNode.children[0].style.cursor='grab';
    var countryNameSigs = '';
    if (sigcount > 1) {
        signaturesFR[0] = ["<center>" + countryName + " - " +
sigcount + " signataires</center>"];
        signaturesEN[0] = ["<center>" + countryName + " - " +
sigcount + " signatories</center>"];
        signaturesDE[0] = ["<center>" + countryName + " - " +
sigcount + " unterzeichner</center>"];
        signaturesPT[0] = ["<center>" + countryName + " - " +
sigcount + " signatários</center>"];
        signaturesES[0] = ["<center>" + countryName + " - " +
sigcount + " signatarios</center>"];
        signaturesIT[0] = ["<center>" + countryName + " - " +
sigcount + " firmatari</center>"];
    }
    else {
        signaturesFR[0] = ["<center>" + countryName + " - " +
sigcount + " signataire</center>"];
        signaturesEN[0] = ["<center>" + countryName + " - " +
sigcount + " signatory</center>"];
    }
}

```



```

        signaturesDE[0] = ["<center>" + countryName + " - " +
sigcount + " unterzeichner</center>"];
        signaturesPT[0] = ["<center>" + countryName + " - " +
sigcount + " signatário</center>"];
        signaturesES[0] = ["<center>" + countryName + " - " +
sigcount + " signatario</center>"];
        signaturesIT[0] = ["<center>" + countryName + " - " +
sigcount + " firmatario</center>"];
    }
    writeTable(allDiv.divFR, signaturesFR, 'signatures-country-
name');
    writeTable(allDiv.divEN, signaturesEN, 'signatures-country-
name');
    writeTable(allDiv.divDE, signaturesDE, 'signatures-country-
name');
    writeTable(allDiv.divES, signaturesES, 'signatures-country-
name');
    writeTable(allDiv.divIT, signaturesIT, 'signatures-country-
name');
    writeTable(allDiv.divPT, signaturesPT, 'signatures-country-
name');

    this.bindPopup(divCountryName.innerHTML.replace(/signatures-
country-name/gi, 'localTableSignatures'), {
        sticky: true,
        maxHeight: 160,
        maxWidth: 400,
        minWidth: 250
    })
    var countryName = this.feature.properties.name;
    var dataCountry = divCountryName.getAttribute("data-country");
    var dataCountryView = divCountryName.getAttribute("data-
country-view");
    //if( dataCountryView == 'openPopup') return;
    var mapId = document.getElementById("mapSignatures");
    var mapWidth = mapId.offsetWidth /
mapId.offsetParent.offsetWidth;
    if (mapWidth < 0.95) {
        this.unbindPopup();
        if (dataCountry != 'none' && dataCountryView != countryName) {
            divCountryName.setAttribute("data-country-view",
this.feature.properties.name);
            if (divCountryName.parentNode.children.length == 3)
divCountryName.parentNode.removeChild(divCountryName.parentNode.children[2]);
            var cln = divCountryName.cloneNode(true);
            cln.style.display = 'block'
            divCountryName.parentNode.appendChild(cln);
        }
    }

```

```

    }
    else {
        if (divCountryName.parentNode.children.length == 3)
divCountryName.parentNode.removeChild(divCountryName.parentNode.children[2]);
        if (dataCountry != 'none' && dataCountryView != countryName) {
            //divCountryName.setAttribute("data-country-view",
"openPopup");
            divCountryName.setAttribute("data-country-view",
this.feature.properties.name);
            this.openPopup();
        }
    };
    this.getPopup().on('remove', function() {
        divCountryName.setAttribute("data-country-view", "");
        /*var mapId = document.getElementById("mapSignatures");
        var mapWidth = mapId.offsetWidth /
mapId.offsetParent.offsetWidth;
        if (mapWidth < 0.95) {
            setCallBackLarge();
        }
        else {
            setCallBackSmall();
        }*/
    });
}
else {
    divCountryName.setAttribute("data-country", "none");
    //divCountryName.parentNode.children[0].style.cursor='default';
}
}

function enterLayer() {
    if (divCountryName == null) return
    var countryName = this.feature.properties.name;
    var dataCountryView = divCountryName.getAttribute("data-country-
view");
    //if(dataCountryView == 'openPopup' || dataCountryView ==
countryName) return;
    if (dataCountryView == countryName) return;
    //var signatures = '';
    var signaturesFR = ['title'];
    var signaturesEN = ['title'];
    var signaturesDE = ['title'];
    var signaturesPT = ['title'];
    var signaturesES = ['title'];
    var signaturesIT = ['title'];
    var sigcount = 0;
    var lang = divCountryName.getAttribute("data-lang");

```

```

allDiv.divFR.className = 'signatures-country-name-wrapper';
allDiv.divEN.className = 'signatures-country-name-wrapper';
allDiv.divDE.className = 'signatures-country-name-wrapper';
allDiv.divES.className = 'signatures-country-name-wrapper';
allDiv.divIT.className = 'signatures-country-name-wrapper';
allDiv.divPT.className = 'signatures-country-name-wrapper';
allDiv.divFR.style.display = 'none';
allDiv.divEN.style.display = 'none';
allDiv.divDE.style.display = 'none';
allDiv.divES.style.display = 'none';
allDiv.divIT.style.display = 'none';
allDiv.divPT.style.display = 'none';
if (lang == 'fr') allDiv.divFR.style.display = 'inline-block';
else if (lang == 'en') allDiv.divEN.style.display = 'inline-
block';
else if (lang == 'de') allDiv.divDE.style.display = 'inline-
block';
else if (lang == 'es') allDiv.divES.style.display = 'inline-
block';
else if (lang == 'it') allDiv.divIT.style.display = 'inline-
block';
else if (lang == 'pt') allDiv.divPT.style.display = 'inline-
block';

for (var idx in countries) {
  if (countryName == countries[idx].name) {
    for (var i in countries[idx].signatures) {
      var data = countries[idx].signatures[i];
      signaturesFR.push([data.nomprénom, data.fract]);
      signaturesEN.push([data.nomprénom, data.enact]);
      signaturesDE.push([data.nomprénom, data.deact]);
      signaturesPT.push([data.nomprénom, data.ptact]);
      signaturesES.push([data.nomprénom, data.esact]);
      signaturesIT.push([data.nomprénom, data.itact]);
    }
    sigcount += countries[idx].signatures.length;
  }
  for (var jdx in map2dataCountries) {
    var map2data = map2dataCountries[jdx];
    if ((countryName == map2data.map) && (countries[idx].name ==
map2data.data)) {
      for (var i in countries[idx].signatures) {
        var data = countries[idx].signatures[i];
        signaturesFR.push([data.nomprénom, data.fract]);
        signaturesEN.push([data.nomprénom, data.enact]);
        signaturesDE.push([data.nomprénom, data.deact]);
        signaturesPT.push([data.nomprénom, data.ptact]);
        signaturesES.push([data.nomprénom, data.esact]);
        signaturesIT.push([data.nomprénom, data.itact]);
      }
    }
  }
}

```

```

        }
        sigcount += countries[idx].signatures.length;
    }
}
//if (signatures != '') {
if (sigcount > 0) {
    divCountryName.setAttribute("data-country", "");
    //divCountryName.parentNode.children[0].style.cursor='grab';
    var countryNameSigs = '';
    if (sigcount > 1) {
        signaturesFR[0] = ["<center>" + countryName + " - " +
sigcount + " signataires</center>"];
        signaturesEN[0] = ["<center>" + countryName + " - " +
sigcount + " signatories</center>"];
        signaturesDE[0] = ["<center>" + countryName + " - " +
sigcount + " unterzeichner</center>"];
        signaturesPT[0] = ["<center>" + countryName + " - " +
sigcount + " signatários</center>"];
        signaturesES[0] = ["<center>" + countryName + " - " +
sigcount + " signatarios</center>"];
        signaturesIT[0] = ["<center>" + countryName + " - " +
sigcount + " firmatari</center>"];
    }
    else {
        signaturesFR[0] = ["<center>" + countryName + " - " +
sigcount + " signataire</center>"];
        signaturesEN[0] = ["<center>" + countryName + " - " +
sigcount + " signatory</center>"];
        signaturesDE[0] = ["<center>" + countryName + " - " +
sigcount + " unterzeichner</center>"];
        signaturesPT[0] = ["<center>" + countryName + " - " +
sigcount + " signatário</center>"];
        signaturesES[0] = ["<center>" + countryName + " - " +
sigcount + " signatario</center>"];
        signaturesIT[0] = ["<center>" + countryName + " - " +
sigcount + " firmatario</center>"];
    }
    writeTable(allDiv.divFR, signaturesFR, 'signatures-country-
name');
    writeTable(allDiv.divEN, signaturesEN, 'signatures-country-
name');
    writeTable(allDiv.divDE, signaturesDE, 'signatures-country-
name');
    writeTable(allDiv.divES, signaturesES, 'signatures-country-
name');
    writeTable(allDiv.divIT, signaturesIT, 'signatures-country-
name');

```

```

writeTable(allDiv.divPT, signaturesPT, 'signatures-country-
name');

    if (lang == 'fr') this.bindTooltip("<span
class='tooltiptext'>cliquez pour les détails</span>", {
        sticky: true,
        className: 'myCSSClassTooltip'
    }).openTooltip();
    else if (lang == 'en') this.bindTooltip("<span
class='tooltiptext'>click for details</span>", {
        sticky: true,
        className: 'myCSSClassTooltip'
    }).openTooltip();
    else if (lang == 'de') this.bindTooltip("<span
class='tooltiptext'>Klicken Sie hier für Details</span>", {
        sticky: true,
        className: 'myCSSClassTooltip'
    }).openTooltip();
    else if (lang == 'es') this.bindTooltip("<span
class='tooltiptext'>click para más detalles</span>", {
        sticky: true,
        className: 'myCSSClassTooltip'
    }).openTooltip();
    else if (lang == 'it') this.bindTooltip("<span
class='tooltiptext'>clicare per i dettagli</span>", {
        sticky: true,
        className: 'myCSSClassTooltip'
    }).openTooltip();
    else if (lang == 'pt') this.bindTooltip("<span
class='tooltiptext'>clique para detalhes</span>", {
        sticky: true,
        className: 'myCSSClassTooltip'
    }).openTooltip();
    this.bindPopup(divCountryName.innerHTML.replace(/signatures-
country-name/gi, 'localTableSignatures'), {
        sticky: true,
        maxHeight: 160,
        maxWidth: 400,
        minWidth: 250
    })
    this.bringToFront();
    this.setStyle({
        weight: 2,
        opacity: 1
    });
    this.getPopup().on('remove', function() {
        divCountryName.setAttribute("data-country-view", "");
        var mapId = document.getElementById("mapSignatures");

```

```

        var mapWidth = mapId.offsetWidth /
mapId.offsetParent.offsetWidth;
        if (mapWidth < 0.95) {
            setCallBackLarge();
        }
        else {
            //setCallBackSmall();
        }
    });
}
else {
    divCountryName.setAttribute("data-country", "none");
    //divCountryName.parentNode.children[0].style.cursor='default';
}
}

function clickLayer() {
    var countryName = this.feature.properties.name;
    var dataCountry = divCountryName.getAttribute("data-country");
    var dataCountryView = divCountryName.getAttribute("data-country-
view");

    //if( dataCountryView == 'openPopup') return;
    this.unbindTooltip();
    var mapId = document.getElementById("mapSignatures");
    var mapWidth = mapId.offsetWidth /
mapId.offsetParent.offsetWidth;
    if (mapWidth < 0.95) {
        this.unbindPopup();
        if (dataCountry != 'none' && dataCountryView != countryName) {
            divCountryName.setAttribute("data-country-view",
this.feature.properties.name);
            if (divCountryName.parentNode.children.length == 3)
divCountryName.parentNode.removeChild(divCountryName.parentNode.children[2]);
            var cln = divCountryName.cloneNode(true);
            cln.style.display = 'block'
            divCountryName.parentNode.appendChild(cln);
        }
    }
    else {
        if (divCountryName.parentNode.children.length == 3)
divCountryName.parentNode.removeChild(divCountryName.parentNode.children[2]);
        if (dataCountry != 'none' && dataCountryView != countryName) {
            //divCountryName.setAttribute("data-country-view",
"openPopup");
            divCountryName.setAttribute("data-country-view",
this.feature.properties.name);
            this.openPopup();
        }
    }
}

```

```
    };
    //divCountryName.setAttribute("data-country",
this.feature.properties.name);
    }

    function leaveLayer() {
        //var dataCountryView = divCountryName.getAttribute("data-
country-view");
        //if( dataCountryView == 'openPopup') return;
        //$CountryName.hide();

        //divCountryName.style.display = 'none'
        //divCountryName.parentNode.children[0].style.cursor='default';
        this.bringToBack();
        this.setStyle({
            weight: 1,
            opacity: .5
        });
    }

    }
    addLegend();
    },
    simpleSheet: false
    })
}).catch(function() {
    console.log('Oh no, epic failure!');
});
};
</script>
```