

## Compte rendu PPE Javascript

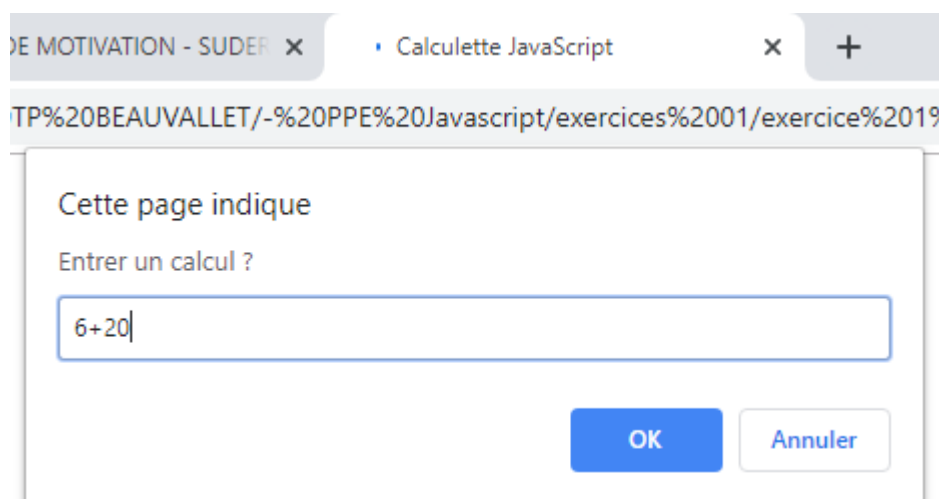
Le but de cet activité fut de réaliser plusieurs exercices afin de savoir bien maîtriser le Javascript.

### Exercice 1.1 :

Le but de ce premier exercice fut de savoir écrire une calculatrice tout en utilisant les fonctions « eval » et « prompt » :

```
<html>
<head><title>Calculatrice JavaScript</title></head>
<body>
<script language="JavaScript">
document.write("<BIG><B>M</BIG>a calculatrice <BIG>J</BIG>ava<BIG>S</BIG>cript</B>");
document.write("<BR>-----</BR>");
var calcul;
calcul=prompt("Entrer un calcul ?", "");
document.write("Votre calcul : ",calcul," est égal à ",eval(calcul),".");
</script>
</body>
</html>
```

Ce premier exercice introduit la manière dont nous allons constamment utiliser le Javascript, c'est à dire en introduisant dans notre fichier .html, un script Javascript dans lequel il va nous être possible d'utiliser plusieurs fonctions comme « eval » qui lui nous permet d'effectuer un calcul à partir de variables introduit dans le script, ou encore « document.write » qui nous permet d'introduire du texte.



## Ma calculette JavaScript

Votre calcul : 6+20 est égal à 26.

Comme on peut le voir ci-dessous, en effectuant le calcul 20+6, on obtient alors 26.

### Exercice 1.2 :

Dans ce second exercice, notre but fut d'écrire une page HTML pouvant afficher la date et l'heure courante :

```
<html>
<head><title>Date et Heure JavaScript</title></head>
<body>
<script language="JavaScript">
document.write("<BIG><B>D</BIG>ate et <BIG>H</BIG>eure <BIG>J</BIG>ava<BIG>S</BIG>cript</B>");
document.write("<BR>-----</BR>");
var maintenant=new Date();
var jour=maintenant.getDate();
var mois=maintenant.getMonth()+1;
var an=maintenant.getFullYear();
var heure=maintenant.getHours();
var minutes=maintenant.getMinutes();
var secondes=maintenant.getSeconds();

document.write("Nous sommes le ",jour,"/",mois,"/",an,".");
document.write("<BR>Il est ",heure,"h ",minutes,"mn et ",secondes,"s.</BR>");
</script>
</body>
</html>
```

Le principe reste le même pour cet exercice et les autres à venir, on introduit le script dans la page .html, on utilise « document.write » pour afficher du texte ainsi que les variables définies juste avant qui permettront d'afficher une date et heure actuelle (« getDate() », « getMonth() », « getFullYear() », ...).

## Date et Heure JavaScript

Nous sommes le 2/12/2019.  
Il est 20h 27mn et 47s.

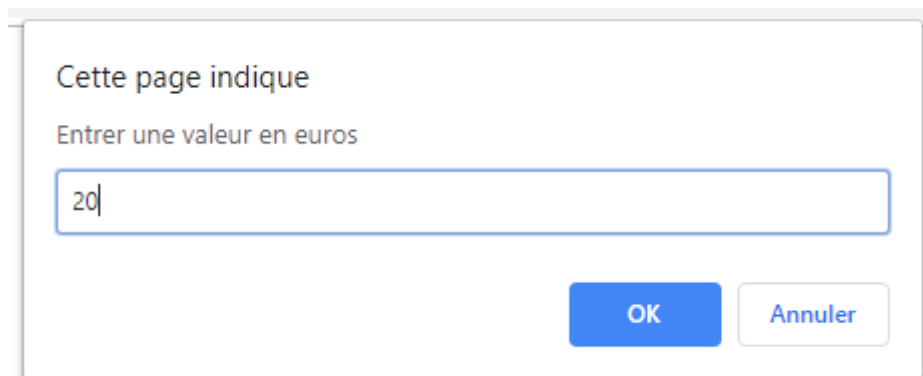
20:27  
02/12/2019

## Exercice 1.3 :

Pour ce troisième exercice, il a fallu cette fois-ci écrire une page HTML permettant de convertir en francs des valeurs en euros :

```
<html>
<head><title>Euros-Francs JavaScript</title></head>
<body>
<script language="JavaScript">
document.write("<BIG><B>E</BIG>uros-<BIG>F</BIG>rancs <BIG>J</BIG>ava<BIG>S</BIG>cript</B>");
document.write("<BR>-----</BR>");
var euros;
euros=prompt("Entrer une valeur en euros","");
document.write("Valeur en euros : ",euros,".");
document.write("<BR>Valeur en francs : ",euros*6.56,".</BR>");
</script>
</body>
</html>
```

Ici, j'ai utilisé la variable « euros », dans laquelle on rentre n'importe quel valeur que l'on souhaite convertir en francs, celle-ci est alors multiplié par 6,56 afin d'obtenir une somme en francs.



Cette page indique

Entrer une valeur en euros

OK Annuler

### **Euros-Francs JavaScript**

-----

Valeur en euros : 20.

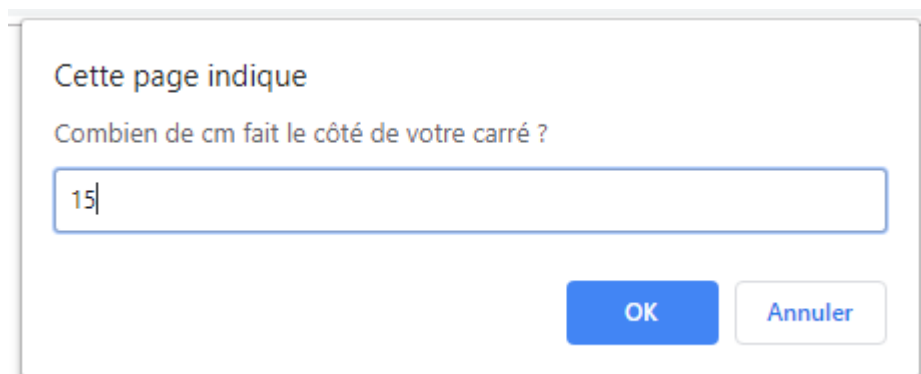
Valeur en francs : 131.2.

## Exercice 1.4 :

Sur ce quatrième exercice, l'objectif fut de calculer le périmètre ainsi que l'aire d'un carré depuis une page HTML :

```
<html>
<head><title>Carrés JavaScript</title></head>
<body>
<script language="JavaScript">
document.write("<BIG><B>C</BIG>arrés <BIG>J</BIG>ava<BIG>S</BIG>cript</B>");
document.write("<BR>-----</BR>");
var carre;
carre=prompt ("Combien de cm fait le côté de votre carré ?");
document.write("Côté du carré : ",carre,".");
document.write("<BR>Périmètre du carré : ",carre*4,".</BR>");
document.write("Aire du carré : ",carre*carre,".");
</script>
</body>
</html>
```

On applique ici dans le script Javascript, les formules basiques de mathématiques permettant d'obtenir le périmètre et l'aire d'un carré.



### **Carrés JavaScript**

-----  
Côté du carré : 15.  
Périmètre du carré : 60.  
Aire du carré : 225.

## Exercice 2.1 :

Ensuite, pour ce cinquième exercice, l'objectif fut de convertir un mot en majuscule par un simple appui sur un bouton tout en utilisant la fonction « toUpperCase() » de l'objet String :

```
<html>
<head><title>Mise en Majuscules - JavaScript</title></head>
<body>
<script language="JavaScript">
document.write("<BIG><B>M</BIG>ise en <BIG><B>M</BIG>ajuscules - <BIG>J</BIG>ava<BIG>S</BIG>cript</B>");
document.write("<BR>-----</BR>");
document.write("Entrez un mot et appuyez sur le bouton =>.");

function transforme() {
    var exp=document.maj.expr.value;
    document.maj.reponse.value=exp.toUpperCase();
}
</script> <br> </br>
<FORM NAME="maj">
<INPUT NAME="expr" TYPE="text" SIZE="15">
<INPUT TYPE="button" VALUE="=>" ONCLICK="transforme()">
<INPUT NAME="reponse" TYPE="text" SIZE="15"
</body>
</html>
```

C'est à partir de cet exercice que l'on va commencer à utiliser des formes depuis Notepad++ qui vont nous permettre d'introduire des zones de texte, des boutons et autres.

### **Mise en Majuscules - JavaScript**

**Entrez un mot et appuyez sur le bouton =>.**

énucléation	=>	ÉNUCLÉATION
-------------	----	-------------

## Exercice 2.2 :

Ensuite, dans ce sixième exercice, il nous a fallu créer un document HTML nous permettant d'entrer un prix hors taxe et d'afficher le prix TTC de celui-ci pour un taux de TVA de 18,6% :

```
<html>
<head><title>Prix TTC - JavaScript</title></head>
<body>
<h1> Prix TTC - JavaScript </h1>
<h2> ----- <h2>
<script language="JavaScript">
function HTtoTTC() {
    var HT=eval(document.conversion.HT.value);
    var TTC=HT*(1+18.6/100);
    TTC=Math.round(TTC*100)/100;
    document.conversion.TTC.value=String(TTC);
}
</script>
<FORM NAME="conversion">
    Prix HT :
    <INPUT NAME="HT" TYPE="text" SIZE="10">
    <TD COLSPAN="2">
    <br> </br>
    <INPUT TYPE="button" VALUE="Calculer" ONCLICK="HTtoTTC()">
    <br> </br>
    Prix TTC :
    <INPUT NAME="TTC" TYPE="text" SIZE="10">
</FORM>
</body>
</html>
```

Même principe que sur l'exercice d'avant, on rentre les conversions d'un prix hors taxe à un prix toute taxe comprise depuis le script Javascript et sur le fichier .html on insère les zones de textes et le bouton « Calculer » dans lequel nous allons convertir les prix rentrés dans le script.

## Prix TTC - JavaScript

Prix HT :

Prix TTC :

## Exercice 2.3 :

Puis, dans ce septième exercice, notre but fut de créer une nouvelle fois un document HTML nous permettant d'entrer 3 notes de DS pouvant être calculés afin d'afficher la moyenne obtenue :

```
<html>
<head><title>Calcul de moyenne - JavaScript</title></head>
<body>
<h1> Calcul de moyenne - JavaScript <br> </h1>
<h2> ----- </br> <h2>
<script language="JavaScript">
function calculmoyenne() {
    var DS1=eval(document.notes.DS1.value);
    var DS2=eval(document.notes.DS2.value);
    var DS3=eval(document.notes.DS3.value);
    var Moyenne=(DS1+DS2+DS3)/3;
    Moyenne=Math.round(Moyenne*10)/10;
    document.notes.Moyenne.value=String(Moyenne);
}
</script>
<FORM NAME="notes">
Note du DS1 :
<INPUT NAME="DS1" TYPE="text" SIZE="10">
<br> </br>
Note du DS2 :
<INPUT NAME="DS2" TYPE="text" SIZE="10">
<br> </br>
Note du DS3 :
<INPUT NAME="DS3" TYPE="text" SIZE="10">
<br> </br>
<INPUT TYPE="button" VALUE="Calculer" ONCLICK="calculmoyenne()">
<br> </br>
Moyenne obtenue :
<INPUT NAME="Moyenne" TYPE="text" SIZE="10">
</FORM>
</body>
</html>
```

Toujours le même principe que sur les deux exercices précédents, on insère dans le script la formule nous permettant d'obtenir une moyenne des 3 notes et dans le html, on insère nos zones de textes afin d'y insérer les notes et la moyenne + un bouton calculant la moyenne.

# Calcul de moyenne - JavaScript

---

Note du DS1 :

Note du DS2 :

Note du DS3 :

Moyenne obtenue :

## Exercice 2.4 :

Pour ce huitième exercice, il nous a fallu, comme dans le premier exercice, créer une calculatrice mais cette fois-ci affichant une zone d'entrée des calculs avec un bouton « = » et une zone de réponse :

```
<html>
<head><title>Calculatrice en ligne - JavaScript</title></head>
<body>
<h1> Calculatrice en ligne - JavaScript </h1>
<h2> ----- <h2>
<script language="JavaScript">
function calculer() {
    var exp=document.calculatrice.calcul.value;
    var rep=eval(exp);
    document.calculatrice.reponse.value=rep;
}
</script>
Entrez votre calcul et cliquez sur le bouton "=".
<FORM NAME="calculatrice">
<INPUT NAME="calcul" TYPE="text" SIZE="15">
<INPUT TYPE="button" VALUE=" = " ONCLICK="calculer()">
<INPUT NAME="reponse" TYPE="text" SIZE="6">
</FORM>
</body>
</html>
```

Même principe que pour les trois derniers exercices.



# Calculatrice en ligne - JavaScript

---

Entrez votre calcul et cliquez sur le bouton "=".

20*5+2	=	102
--------	---	-----

## Exercice 3.1 :

Pour ce neuvième exercice, l'objectif fut une nouvelle fois de créer la moyenne de 3 notes de DS mais cette fois-ci en rajoutant un commentaire en fonction de la moyenne obtenue :

```
function commentaires(Moyenne) {  
  if (Moyenne==0) return "C'est très mauvais, à revoir !";  
  else if (Moyenne<7) return "C'est très insuffisant !";  
  else if (Moyenne<10) return "La moyenne, peut mieux faire.";  
  else if (Moyenne<12) return "C'est assez bien.";  
  else if (Moyenne<14) return "C'est bien.";  
  else if (Moyenne<16) return "C'est très bien.";  
  else if (Moyenne<18) return "C'est presque parfait.";  
  else return "C'est excellent !";  
}
```

Le fichier .html ne diffère en rien de ce que j'ai pu rentrer dans le septième exercice mis à part cette fonction « commentaire(Moyenne) » qui, en fonction de la valeur de la variable « Moyenne », enverra le texte écrit sur le programme.

Note du DS1 :

Note du DS2 :

Note du DS3 :

Moyenne obtenue :

## Exercice 3.2 :

Pour ce dixième exercice, j'ai dû cette fois-ci créer un document HTML similaire au second exercice dans lequel il m'a fallu entrer la date en indiquant le nom du jour en français :

```
<html>
<head><title>Date - JavaScript</title></head>
<body>
<script language="JavaScript">

    var tabjours=["Lundi", "Mardi", "Mercredi", "Jeudi", "Vendredi", "Samedi", "Dimanche"];

    var maintenant=new Date();
    var jour=maintenant.getDate();
    var mois=maintenant.getMonth()+1;
    var an=maintenant.getFullYear();
    var numJour=maintenant.getDay();

    document.write("<BIG><B>D</BIG>ate - <BIG>J</BIG>ava<BIG>S</BIG>cript</B>");
    document.write("<BR>-----</BR>");
    document.write("Nous sommes le "+tabjours[numJour-1]+" "+jour+"/"+mois+"/"+an+".");

</script>
</body>
</html>
```

Le procédé reste le même que dans le second exercice sauf que cette fois-ci nous rajoutons une table contenant les 7 jours de la semaine. Sachant que l'utilisateur rentrera une valeur entre 1 et 7 et que la table commence de 0 jusqu'à 6, on entrera alors le calcul « tabjours[numJour-1] » afin que le bon jour soit affiché.

### **Date - JavaScript**

-----  
Nous sommes le Lundi 2/12/2019.

## Exercice 3.3 :

Dans ce onzième exercice, il faut cette fois-ci entrer une date (jour, mois et année) permettant de nous dire à quel jour appartient cette date :

```

<html>
<head><title>Jour Anniversaire - JavaScript</title></head>
<h1> Jour Anniversaire - JavaScript </h1>
<HR>
<body>
<script language="JavaScript">
function calculjour() {
var tabjours=["Lundi", "Mardi", "Mercredi", "Jeudi", "Vendredi", "Samedi", "Dimanche"];
var maintenant=new Date();
var jour=eval(document.JourAnniversaire.Jour.value);
var mois=eval(document.JourAnniversaire.Mois.value);
var an=eval(document.JourAnniversaire.Annee.value);
if (an<100) an=an+1900;
var ladate=new Date(an,mois-1,jour);
var numJour=ladate.getDay();
document.JourAnniversaire.résultat.value="Le "+jour+"/"+mois+"/"+an+" est un "+tabjours[numJour-1]+" !";
}
</script>
<FORM NAME="JourAnniversaire">
Quel jour êtes-vous né ?
<br> <br>
Lundi ?, Mardi ?, Mercredi ?, Jeudi ?, Vendredi ?, Samedi ?, Dimanche ?
<br> <br>
Il vous suffit de remplir le formulaire suivant pour le savoir.
<br> <br>
Jour :
<INPUT NAME="Jour" TYPE="text" SIZE="10">
<br> <br>
Mois :
<INPUT NAME="Mois" TYPE="text" SIZE="10">
<br> <br>
Année :
<INPUT NAME="Annee" TYPE="text" SIZE="10">
<br> <br>
<INPUT TYPE="button" VALUE="Calculer" ONCLICK="calculjour()">
<br> <br>
<INPUT NAME="résultat" TYPE="text" SIZE="40">
</FORM>
</body>
</html>

```

Le procédé est identique aux autres exercices effectués précédemment.

## Jour Anniversaire - JavaScript

Quel jour êtes-vous né ?

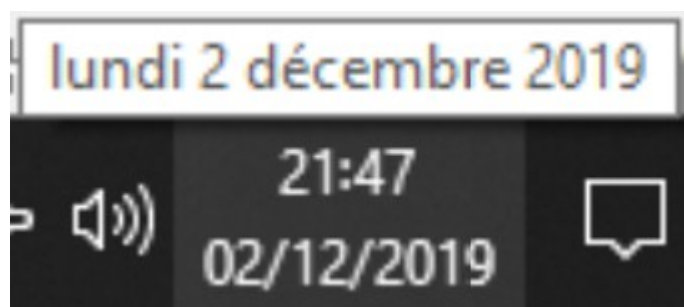
Lundi ?, Mardi ?, Mercredi ?, Jeudi ?, Vendredi ?, Samedi ?, Dimanche ?

Il vous suffit de remplir le formulaire suivant pour le savoir.

Jour :

Mois :

Année :



## Exercice 3.4 :

Dans ce douzième exercice, il a fallu créer un répéteur de tables de multiplications dans laquelle une multiplication entre la table de 2 et de 9 est proposée (afin que l'opération ne soit pas trop facile et qu'elle ait un but) et où la réponse est fournie dans une zone d'édition avec notamment un bouton permettant de vérifier la réponse :

```
<html>
<head><title>Tables de multiplications - JavaScript</title></head>
<h1> Tables de multiplications - JavaScript </h1>
<HR>
<body>
<script language="JavaScript">
var n1=Math.floor(Math.random()*8)+2;
var n2=Math.floor(Math.random()*8)+2;

function verifcalcul() {
  rep=document.calcul.reponse.value;
  if (eval(rep)==n1*n2) {
    alert("Bonne réponse !");
  }
  else {
    alert("Faux !");
  }
}
document.write("<B><FONT SIZE='1'>",n1," x ",n2," = </FONT></B>");
</script>
<FORM NAME="calcul">
<INPUT NAME="reponse" TYPE="text" SIZE="4">
<INPUT TYPE="button" VALUE="Vérifier" ONCLICK="verifcalcul();">
</FORM>
</body>
</html>
```

J'ai utilisé ici la fonction « Math.floor(Math.random()) » afin d'obtenir un nombre entre 2 et 9 pour les calculs mentaux qui seront proposées. En cas de bonne réponse, l'utilisateur verra un message le félicitant de sa bonne réponse et dans le cas contraire lui annoncera qu'il a donné une réponse incorrecte.

### Tables de multiplications - JavaScript

Cette page indique  
Faux !

7 x 7 =

### Tables de multiplications - JavaScript

Cette page indique  
Bonne réponse !

9 x 9 =

## Exercice 3.5 :

Pour ce treizième exercice, l'objectif fut de faire un jeu dans lequel l'ordinateur choisissait un nombre entre 0 et 100 et que l'utilisateur tente de le deviner en faisant des propositions dans lesquelles l'ordinateur indiquait si le nombre proposé est trop grand ou trop petit :

```
<html>
<head><title>Nombre secret - JavaScript</title></head>
<h1> Nombre secret - JavaScript </h1>
<HR>
L'ordinateur a choisi un nombre entier inférieur à 100. <br>
A vous de le retrouver le plus rapidement possible en proposant des nombres et en tenant compte des réponses fournies.
<body>
<script language="JavaScript">
var nombreSecret=Math.round(Math.random()*100);
var nombreEssais=0;

function nombre() {
nombreEssais++;
var nb=document.nombresecret.proposition.value;
var nombre=eval(nb);
var commentaire="Essai "+nombreEssais+" : ";
if (nombre==nombreSecret) commentaire="Bonne réponse !";
else if (nombre<nombreSecret) commentaire="Plus grand !";
else commentaire="Plus petit !";
document.nombresecret.reponse.value=commentaire;
}
</script>
<FORM NAME="nombresecret">
<br>
Nombre proposé :
<INPUT NAME="proposition" TYPE="text" SIZE="5">
<INPUT TYPE="button" VALUE="Valider" ONCLICK="nombre()">
<br>
<BR>
<INPUT NAME="reponse" TYPE="text" SIZE="30">
</FORM>
<HR>
<A HREF="file:///F:/BTS%20SIO/SI/Compte-rendu%20TF%20BEAUVALLLET/-%20PFE%20Javascript/exercices%2003/exercice%203%20-%205).html">Autre partie</A>
</body>
</html>
```

Le procédé est assez identique à celui de l'exercice précédent, on utilise une fois de plus la fonction « `Math.floor(Math.random())` » afin que l'ordinateur sélectionne un chiffre entre 1 et 100 et quand l'utilisateur rentre un nombre dans la variable « nombre » celui-ci sera avertie si le chiffre secret est plus grand, plus petit ou si il vient de trouver le chiffre secret. On continue d'ailleurs d'utiliser les zones de textes et les boutons avec cette fois-ci un lien nommé « Autre partie » menant sur la même page (la page s'actualise et permet de refaire une partie).

## Nombre secret - JavaScript

---

L'ordinateur a choisi un nombre entier inférieur à 100.

A vous de le retrouver le plus rapidement possible en proposant des nombres et en tenant compte des réponses fournies.

Nombre proposé :

---

[Autre partie](#)

# Nombre secret - JavaScript

L'ordinateur a choisi un nombre entier inférieur à 100.

A vous de le retrouver le plus rapidement possible en proposant des nombres et en tenant compte des réponses fournies.

Nombre proposé :

[Autre partie](#)

# Nombre secret - JavaScript

L'ordinateur a choisi un nombre entier inférieur à 100.

A vous de le retrouver le plus rapidement possible en proposant des nombres et en tenant compte des réponses fournies.

Nombre proposé :

[Autre partie](#)

## Exercice 4.1 :

Dans ce quatorzième exercice, notre objectif fut de créer un document HTML nous demandant un nombre afin de pouvoir nous afficher la table de multiplication assigné à celui-ci :

```
<html>
<head><title>Tables de multiplications - JavaScript</title></head>
<h1> Tables de multiplications - JavaScript </h1>
<HR>
<body>
<script language="JavaScript">
  var nombre=prompt("Entrer un nombre entier.");
  var rep=0;
  for (i=1; i<=10; i++) {
    rep=nombre*i;
    document.writeln("<BR>",nombre,"x",i," = ",rep);
  }
</script>
<br> </br>
<HR>
<A HREF="file:///F:/BTS%20SIO/SI/Compte-rendu%20TP%20BEAUVALLET/-%";
</body>
</html>
```

Dans cet exercice, j'ai utilisé une boucle FOR afin de faire en sorte que tant que la valeur i est inférieure ou égale à 10, on incrémente celle-ci jusqu'à obtenir toute la table du nombre avec un lien permettant d'effectuer une autre table.

Cette page indique

Entrer un nombre entier.

OK

Annuler

## Tables de multiplications - JavaScript

---

7x1 = 7  
7x2 = 14  
7x3 = 21  
7x4 = 28  
7x5 = 35  
7x6 = 42  
7x7 = 49  
7x8 = 56  
7x9 = 63  
7x10 = 70

---

[Autre table](#)

### Exercice 4.2 :

Dans ce quinzième exercice, le but fut de construire la table de Pythagore en imbriquant deux boucles for :

```

<html>
<head><title>Table de Pythagore - JavaScript</title></head>
<h1> Table de Pythagore - JavaScript </h1>
<HR>
<body>
<TABLE BORDER="1" CELLPADDING="10" CELLSPACING="2">
<script language="JavaScript">
var ligne;
var colonne;
for (ligne=1; ligne<10; ligne++) {
  document.writeln("<TR>");
  for (colonne=1; colonne<10; colonne++)
    document.writeln("<TD ALIGN=\"right\">",ligne*colonne,"</TD>");
  document.writeln("</TR>");
}
</script>
</TABLE>
</body>
</html>

```

Le procédé de la boucle FOR reste le même que dans l'exercice précédent, on multiplie la valeur se trouvant sur la ligne avec celle se trouvant sur la colonne et on crée le tableau à partir du html avec la taille de la bordure, les espacements dans les cases, ... (« BORDER », « CELLPADDING » et « CELLSPACING »).

## Table de Pythagore - JavaScript

1	2	3	4	5	6	7	8	9
2	4	6	8	10	12	14	16	18
3	6	9	12	15	18	21	24	27
4	8	12	16	20	24	28	32	36
5	10	15	20	25	30	35	40	45
6	12	18	24	30	36	42	48	54
7	14	21	28	35	42	49	56	63
8	16	24	32	40	48	56	64	72
9	18	27	36	45	54	63	72	81



## Exercice 4.3 :

Dans ce seizième exercice, il ma fallu créer un document HTML nous permettant d'entrer un entier n pouvant calculer la somme des entiers de 1 à n et qui affiche le résultat :

```
<html>
<head><title>Somme des premiers entiers - JavaScript</title></head>
<h1> Somme des premiers entiers - JavaScript </h1>
<HR>
<body>
<script language="JavaScript">
function sommen(n) {
    var total=0;
    for (var i=1; i<=n; i++) total+=i;
    return total;
}

function calcul() {
    var n=document.addition.valeurn.value;
    var rep=sommen(n);
    document.addition.reponse.value=rep;
}
</script>
<FORM NAME="addition">
Valeur de n :
<INPUT NAME="valeurn" TYPE="text" SIZE="4">
<br> <br>
<INPUT TYPE="button" VALUE="Calculer" ONCLICK="calcul();">
<br> <br>
1 + 2 + ... + n =
<INPUT NAME="reponse" TYPE="text" SIZE="8">
</FORM>
</body>
</html>
```

Même procédé que pour les exercices précédents.

On utilise une nouvelle fois la boucle FOR dans la fonction « somme(n) » afin d'incrémenter l'addition jusqu'au chiffre rentré et dans la seconde fonction on insère les résultats obtenues dans les zones de textes et on programme le bouton « Calculer ».

## Somme des premiers entiers - JavaScript

Valeur de n :

1 + 2 + ... + n =

## Exercice 4.4 :

Pour ce dix-huitième exercice, notre objectif fut de simuler es parties du jeu du 21 dans lequel deux joueurs lancent un dé l'un après l'autre et les points obtenus s'ajoutent, le premier qui atteint ou dépasse 21 points a gagné :

```
<html>
<head><title>Jeu du 21 - JavaScript</title></head>
<h1> Jeu du 21 - JavaScript </h1>
<HR>
Deux joueurs lancent un dé l'un après l'autre. <br>
Les points obtenus s'ajoutent, le premier qui atteint ou dépasse 21 a gagné. <br>
Voici une partie : <br> <br>
<body>
<TABLE BORDER="1" CELLPADDING="10" CELLSPACING="2" >
<TR><TD><B>Joueur 1</B>
</TD><TD><B>Joueur 2</B>
</TD><TD><B>Total</B></TD></TR>
<script language="JavaScript">
var total=0;
var joueur=1;
var dé=0;
while (total<21) {
  dé=1+Math.floor(Math.random()*6);
  total+=dé;
  document.writeln("<TR><TD>");
  if (joueur==1)
    document.writeln(dé,"</TD><TD><BR></TD>");
  else
    document.writeln("<BR></TD><TD>",dé,"</TD>");
  document.writeln("<TD>",total,"</TD></TR>");
  joueur=3-joueur;
}
</script>
</TABLE>
<br>
<SCRIPT LANGUAGE="JavaScript">
joueur=3-joueur;
document.writeln("Le joueur ",joueur," a gagné !");
</SCRIPT>
<br>
<br>
<A HREF="file:///F:/BTS%20SIO/SI/Compte-rendu%20TP%20BEAUVALLET/-%20PPE%20Javascri
</body>
</html>
```

J'ai utilisé ici une boucle TANT QUE afin de faire en sorte que la valeur du dé soit aléatoire (entre 1 et 6) et que les somme suivantes s'ajoutent à cette même valeur tant que le total de point n'a pas atteint 21 avec à la fin du programme, un message annonçant le vainqueur. Comme dans l'exercice précédant, le procédé est identique pour la construction du tableau.

# Jeu du 21 - JavaScript

---

Deux joueurs lancent un dé l'un après l'autre.

Les points obtenus s'ajoutent, le premier qui atteint ou dépasse 21 a gagné.

Voici une partie :

Joueur 1	Joueur 2	Total
5		5
	4	9
3		12
	6	18
2		20
	3	23

Le joueur 2 a gagné !

[Autre partie...](#)

## Exercice 4.5 :

Dans ce dix-neuvième exercice, il m'a fallu cette fois-ci répéter 1000 fois une partie du jeu du 21 pour savoir si le joueur démarrant la partie a plus de chances de gagner que le second joueur :

```

<html>
<head><title>Jeu du 21 (statistiques) - JavaScript</title></head>
<h1> Jeu du 21 (statistiques) - JavaScript </h1>
<HR>
Deux joueurs lancent un dé l'un après l'autre.
Les points obtenus s'ajoutent, le premier qui atteint ou dépasse 21 a gagné.
<h4> Qui a le plus de chances de gagner ? </h4>
Le joueur 1 qui commence à lancer ou le joueur 2 ? <br> <br>
Pour nous faire une opinion, simulons 1000 parties et observons les résultats.
<br>
<body>
<script language="JavaScript">
var score_joueur1=0;
var score_joueur2=0;

function partie() {
  var total=0;
  var joueur=1;
  var dé=0;
  while (total<21) {
    dé=1+Math.floor(Math.random()*6);
    total+=dé;
    joueur=3-joueur;
  }
  joueur=3-joueur;
  if (joueur==1) score_joueur1++;
  else score_joueur2++;
}

for (var i=0; i<1000; i++) partie();
document.writeln("<BR>Victoires du joueur 1 : ",score_joueur1);
document.writeln("<BR>Victoires du joueur 2 : ",score_joueur2);
</script>
</TABLE>
<br>
<HR>
<A HREF="file:///F:/BTS%20SIO/SI/Compte-rendu%20TP%20BEAUVALLET/-%20PPE%20Java
</body>
</html>

```

Même procédé que dans les précédents exercices mais en utilisant cette fois-ci une boucle FOR et une boucle TANT QUE dans le même programme.

## Jeu du 21 (statistiques) - JavaScript

Deux joueurs lancent un dé l'un après l'autre. Les points obtenus s'ajoutent, le premier qui atteint ou dépasse 21 a gagné.

Qui a le plus de chances de gagner ?

Le joueur 1 qui commence à lancer ou le joueur 2 ?

Pour nous faire une opinion, simulons 1000 parties et observons les résultats.

Victoires du joueur 1 : 497

Victoires du joueur 2 : 503

[Recommencer](#)

### Exercice 5.1 :

Enfin, pour ce vingtième et dernier exercice, mon objectif fut de créer un document HTML pouvant écrire la date avec le nom du jour et le nom du mois en français avec deux tableaux contenant ces noms :

```
<html>
<head><title> Ecrire la date (Tableaux) - JavaScript</title></head>
<h1> Ecrire la date (Tableaux) - JavaScript </h1>
<HR>
<body>
<script language="JavaScript">
var jours=["dimanche","lundi","mardi","mercredi","jeudi","vendredi","samedi"];
var mois=["janvier","février","mars","avril","mai","juin","juillet","août","septembre",
          "octobre","novembre","décembre"];

function afficheDate() {
    var maintenant=new Date();
    var jour=maintenant.getDate();
    var nummois=maintenant.getMonth();
    var an=maintenant.getFullYear();
    var numJour=maintenant.getDay();
    document.write("Nous sommes le ",jours[numJour]," ",jour," ",mois[nummois]," ",an,".");
}
afficheDate();
</script>
</body>
</html>
```

Même procédé que dans les exercices où l'objectif fut d'afficher la date actuelle mais cette fois-ci en utilisant des tableaux pour y inclure les jours ainsi que les mois, comme on peut le voir avec « var jours » et « var mois » avec les données entrées dans des crochets.

## Ecrire la date (Tableaux) - JavaScript

Nous sommes le mardi 3 décembre 2019.