

- TP 1
  - Exercice 1
  - Aides du poulet
  - Aides du poussin
  - Exercice 2
  - Aides du poulet
  - Aides du poussin
  - Exercice 3
  - Aides du poulet
  - Aides du poussin
  - Exercice 4
    - Aides du poulet
  - Aides du poussin
  - Exercice 5
    - Aides du poulet
  - Aides du poussin
  - Exercice 6
    - Aides du poulet
    - Aides du poussin
  - Exercice 7
  - Aides du poulet
    - Aides du poussin
  - Exercice 8
    - Aides du poulet
  - Aides du poussin
  - Exercice 9
    - Aides du poulet
  - Aides du pousson
  - Exercice 10

# TP 1

---

Pour tous les exercices :

- Il est obligatoire de préciser le type de chaque variable et de chaque paramètre de fonction.
- N'oubliez pas de préciser le type de retour, même si la fonction ne retourne rien.

- Chaque exercice doit être dans une fonction séparée.
- Vous devez utiliser des fonctions natives de JavaScript (pas de librairie externe).
- La documentation des fonctions est obligatoire !

## Exercice 1

---

Créez une fonction qui prendra un prénom en paramètre. Votre fonction affichera "Bonjour " suivi du prénom dans un `console.log`.

```
Bonjour Michel !
```

## Aides du poulet

---

- N'oubliez pas de déclarer le type de votre paramètre !
- N'oublier pas de déclarer le type de retour de votre fonction !

## Aides du poussin

---

- Le type de retour de la fonction est `void` car elle ne retourne rien.
- Le type de votre paramètre est `string`.

## Exercice 2

---

Créez une fonction qui prendra deux nombres en paramètre. Votre fonction affichera la somme de ces deux nombres dans un `console.log`

```
5
```

## Aides du poulet

---

- N'oubliez pas de déclarer le type de vos paramètres !
- N'oublier pas de déclarer le type de retour de votre fonction !

## Aides du poussin

---

- Le type de retour de la fonction est `void` car elle ne retourne rien.
- Le type de vos paramètres est `number`.

## Exercice 3

---

Créez une fonction qui prendra en paramètre deux nombres : le cout de faabrication d'un objet, et son prix de vente. Votre fonction **retournera** le bénéfice de la vente de cet objet.

```
Perte de 15€  
Gain de 5€
```

## Aides du poulet

---

- N'oubliez pas de déclarer le type de vos paramètres !
- Vous devrez utiliser `return` pour retourner le résultat de votre fonction.
- Vous pouvez utilisez votre fonction directement dans un `console.log` pour afficher le résultat, ou le stocker dans une variable que vous afficherez ensuite.
- Vous aurez besoin d'un bloc if pour vérifier si le prix de vente est supérieur ou égal au cout de fabrication.

## Aides du poussin

---

- Le type de retour de la fonction est `string` car elle retourne du texte.
- N'oubliez pas que vous pouvez concaténez des chaînes de caractères avec le signe `+`, ou avec les magic strings : ``Bonjour ${prenom}``.
- Le type de vos paramètres est `number`.

# Exercice 4

---

Créer une fonction qui prendra 3 nombres en paramètre. Votre fonction retournera le nom de la plus grande valeur dans une phrase.

Le plus grand nombre est 5

## Aides du poulet

- Vous aurez besoin d'un bloc if pour vérifier si le premier nombre est plus grand que le deuxième.

## Aides du poussin

---

- Puis vous vérifiez si le premier nombre est plus grand que le troisième.

# Exercice 5

---

Ecrivez une fonction qui prend un paramètre un nombre indéfini de nombres. Votre fonction déterminera le nombre le plus grand.

## Aides du poulet

- Notez que le rest param prend un nombre indéfini de nombres, stocké dans un tableau. : `...nombres`
- N'oubliez pas de préciser son type, qui ici est un tableau de nombres.

## Aides du poussin

---

- Il vous faudra parcourir ce tableau :
  - Stockez le premier nombre dans une variable

- Comparez le premier nombre avec le second, puis avec le troisième, et ainsi de suite à l'aide d'une boucle **for** ou **while**.
- Vous pourrez ensuite consoler le résultat.

## Exercice 6

---

Ecrivez une fonction qui prendra en paramètre le nom de l'élève, suivi d'un nombre indéterminé de notes. Votre script calculera la moyenne de l'élève, et retournera le résultat suivi d'une mention.

- De 0 à 4 : Catastrophique
- De 5 à 10 : Insuffisant
- De 11 à 14 : Passable
- De 15 à 18 : Bien
- De 19 à 20 : Très bien

Si la note est supérieure à 20 ou inférieure à 0, affichera "Note non valide"

Vous avez l'obligation de découper votre code en plusieurs fonctions !

## Aides du poulet

- Vous aurez besoin d'un bloc if pour vérifier si la note est supérieure à 20 ou inférieure à 0.
- Pensez à utiliser l'opérateur ET (**&&**) pour éviter d'imbriquer les if

## Aides du poussin

- Le rest param sera stocké dans un tableau.
- Il vous faudra additionner tous les nombres entre eux (grâce à une boucle), et diviser le résultat par la longueur du tableau de nombres

# Exercice 7

---

Ecrivez une fonction qui prendra en paramètre deux nombres et un opérateur (+, -, \*, /). Votre fonction retournera le résultat de l'opération sous forme de nombre.

## Aides du poulet

---

- Il existe une méthode qui permet de faire un calcul à partir d'une chaîne de caractères. A vous de la trouver !

## Aides du poussin

- La méthode `eval` permet de calculer une expression mathématique.
- Elle prend en paramètre une expression mathématique, et renvoie le résultat.
- Il vous suffira donc de concaténer les trois variables et de l'envoyer à `eval` pour calculer le résultat.

# Exercice 8

---

Ecrivez une fonction qui prendra en paramètre un nombre. Votre fonction affichera une pyramide horizontale en fonction de ce nombre. Cette fois, vous utiliserez des `console.log` pour afficher la pyramide.

```
*  
**  
***  
****  
*****  
*****  
****  
***  
**  
*
```

## Aides du poulet

- Vous aurez besoin d'une boucle qui aura pour maximum le nombre saisi par l'utilisateur
- Vous aurez besoin d'une boucle pour ajouter les étoiles autant de fois que nécessaire.

## Aides du poussin

---

- Si vous utilisez un tableau :
  - La méthode `push` vous permettra d'ajouter une étoile au tableau
  - La méthode `join()` vous permettra d'afficher les étoiles sous forme de chaîne de caractère
- Notez que vous pouvez aussi utiliser une chaîne de caractères

## Exercice 9

---

Ecrivez une fonction qui prend en paramètre un nombre décimal. Votre fonction retournera le nombre de billets et de pièces nécessaires pour obtenir ce nombre.

```
ex : 2437,57
Billet de 500 : 4
Billet de 200 : 2
Billet de 20 : 1
Billet de 10 : 1
Billet de 5 : 1
Pièce de 2 : 1
Pièce de 0,50 : 1
Pièce de 0.02 : 1
```

Votre fonction vérifiera que l'utilisateur a bien entré un nombre, et affichera un message d'erreur si ce n'est pas le cas.

## Aides du poulet

- Le plus simple est de stocker toutes les monnaies existantes dans un tableau.
- Vous aurez besoin de la méthode `Math.floor()` pour effectuer calcul arrondi à l'inférieur

# Aides du pousson

---

- Si le résultat est inférieur à 5 alors c'est une pièce, sinon c'est un billet.
- Vous soustraierez les résultats au fur et à mesure de la boucle.

## Exercice 10

---

Rédigez une fonction qui prendra en paramètre un nombre indéfini de températures. Votre fonction retournera la valeur la plus proche de 0 La phrase à afficher sera : "La température la plus proche de 0 est de X degrés", et ne devra pas être gérée par la fonction mais par le `console.log`.