

| | | |
|--|--|-----------------------------------|
| Institut Galilée Paris 13 | LANGAGES WEB - JAVASCRIPT Exercice Chronomètre Version Objet | Réf. : WEBC [0.9] Page : 1 - 1 |
|--|--|-----------------------------------|

Exercice Chronometre (Version Objet)

Vous reprenez votre code de l'exercice précédent et vous en faites une copie.

Etape 1 (Version Objet 1):

On veut gérer plusieurs chronomètres dans la même page. On veut donc que le bouton '+' soit toujours actif. Pour cela, il faut isoler et stocker les chronomètres chacun dans un Objet. A cet effet, écrire une classe Chronometre qui sera instanciée à chaque fois que l'on clique sur le bouton '+'.

La seule chose délicate dans cet exercice est de penser à "lier" l'objet "this" lorsqu'une fonction est lancée en mode "callback" (par exemple pour `addEventListener()` et d'autres cas...). Pour la fonction `demarrer()` il faudra ajouter ceci:

```
this.demarrer = this.demarrer.bind(this)
```

Attention, normalement dans votre code, il y a d'autres fonctions exécutées en mode callback. A vous de les repérer et d'établir pour elles aussi, le lien avec l'objet "this".

Désormais les chronomètres seront repérés automatiquement par un identificateur différent: "chrono1" puis "chrono2", "chrono3", etc...

Normalement le cours et surtout vos compétences en Objet devrait suffire à faire cet exercice. Vous pouvez même si ca vous dit, encapsuler votre classe (private vs public).

Etape 2 (Version Objet 2):

Vous reprenez votre code de l'étape précédente et vous en faites une copie.

Petit supplément. A chaque fois que l'on clique sur le '+' on crée un nouvel objet et on le perd un peu dans la nature. On aimerait stocker ces objets (donc nos chronomètres) dans une HashMap (un dictionnaire).

Ecrire une toute petite classe Horlogerie permettant de stocker nos chronomètres. Prévoir l'ajout d'un chronomètre et la suppression d'un chronomètre. Ça devrait suffire !

Réadaptez votre code en conséquence.