

# Debug JavaScript

Christophe Jollivet  
Apside TOP

# Objectifs

- Utilisation des outils
- Apprendre quelques astuces

# Démo

<https://github.com/jollivetc/debugJS/tree/bbl>

# La meilleur façon

- Ne pas avoir besoin de le faire
- Ne dispense pas de lire la documentation
- Faites des tests

# Procédure

- Lire les logs
- Mettre des breakpoints
- « essayer des choses »

# Générer des logs

- Console.log()
- **Attention à IE 8**

# String interpolation

- %s pour une String
- %i ou %d pour un integer
- %f pour un « floating point »
- %o pour un élément du DOM
- %O pour un objet JavaScript

# Formatage dans la console

- `console.log()`, `console.info()`, `console.warning()`, `console.error()`
- `console.group()` .... `console.groupEnd()`
- `console.time(label)` ... `console.timeEnd(label)`
- `console.count(label)`
- `console.table(arrayToShow, arrayOfProperties)`



# Breakpoint

- normal
- conditionnel
- programmatique
- sur le DOM
- sur XHR
- sur Events

# Asynchrone et latence

- Simuler la latence du réseau

# Contexte et console

- langage dynamique
- arguments
- code snippet
- <https://github.com/paulirish/devtools-addons/wiki/Snippets>
- <http://bgrins.github.io/devtools-snippets>

# Avec un framework

- Exemple de AngularJS

# Et côté serveur

- node-inspector

# Q & A

- <https://github.com/jollivetc/debugJS/tree/bbl>