

L'inférence de types en TypeScript

...

Xavier NOPRE
Human Talks Grenoble
10/12/2024

Qui suis-je ?



Xavier NOPRE

Artisan-développeur

CTO @ vrtice



AlpesCraft

- Coding-dojo : <https://www.meetup.com/fr-FR/alpescraft-coding-dojos/>
- Meetup : <https://www.meetup.com/fr-FR/alpescraft-grenoble-meetup>
- Conférence :





*Une partie du contenu de cette présentation a été
générée avec l'aide d'une intelligence artificielle*

C'est quoi ?



“L'inférence de types est un **mécanisme** qui permet à un **compilateur** ou un interpréteur de rechercher **automatiquement** les types associés à des expressions, sans qu'ils soient indiqués **explicitement** dans le code source.”

Quels langages

TypeScript	<code>let x = 42; // number</code>
Kotlin	<code>val x = 42; // Int</code>
Rust	<code>let x = 42; // i32</code>
Haskell	<code>let x = 42; – Num</code>
Inférence de types partielle :	
C# > 3.0	<code>var x = 42; // int</code>
C++ > 11	<code>auto x = 42; // int</code>
Swift	<code>let x = 42; // Int</code>
Java > 10	<code>var x = 42; // int</code>

Et aussi, parfois avec outillage : Python, Ruby, OCaml, F#, Erlang, ...

Typage explicite ou implicite (inférence)

```
let x: number = 42;  
let name: string = "Alice";  
let isActive: boolean = true;  
  
function add(a: number, b: number): number {  
  return a + b;  
}  
  
const result: number = add(10, 20);
```



```
let x = 42;  
let name = "Alice";  
let isActive = true;  
  
function add(a: number, b: number) {  
  return a + b;  
}  
  
const result = add(10, 20);
```

Du code ! 😄

Conclusion

- A utiliser sans modération
 - \Rightarrow Limite le volume de code (à relire)
- Expliciter le typage aux endroits stratégiques :
 - \Rightarrow Casser les “chaînes de transmission”
 - \Rightarrow Expliciter le typage

Merci ! Des questions ?



Xavier NOPRE

Artisan-développeur

CTO @ vrtice

