

**Votre ami le
débugger**

C'est quoi ?

Un outil génial permettant de :

- stopper l'exécution
- inspecter le contexte

Pourquoi faire ?

- comprendre ce qui se passe à l'exécution

...plutôt que d'ajouter des logs

Je trouve ça où ?

- dans le navigateur
- dans l'IDE

Comment je m'en sers ?

- activer le debugger dans l'application
- placer un point d'arrêt dans le code (breakpoint)
- exécuter le code
 - via un test unitaire
 - manuellement en navigant dans l'application

Qu'est ce que je peux faire avec ?

Mettre le programme en pause

- à la volée
- à une ligne spécifique du programme
- lorsqu'une erreur est déclenchée
- en fonction d'une valeur d'une variable
- en fonction d'un autre breakpoint

Inspecter l'état du programme

- inspecter les valeurs
- inspecter la pile d'appels
- appeler une méthode pour voir son résultat

Comprendre le déroulement d'un algorithme

- dérouler pas à pas
- entrer/sortir d'une fonction
- revenir en arrière

Influer sur le programme

- modifier une valeur
- déclencher des erreurs
- changer le flot d'exécution

Avec Node.js

Lancer le serveur avec l'option `--inspect` :

```
{  
  "scripts": {  
    "start:debug": "node --require ts-node/register --inspect src/index.ts"  
  }  
}
```

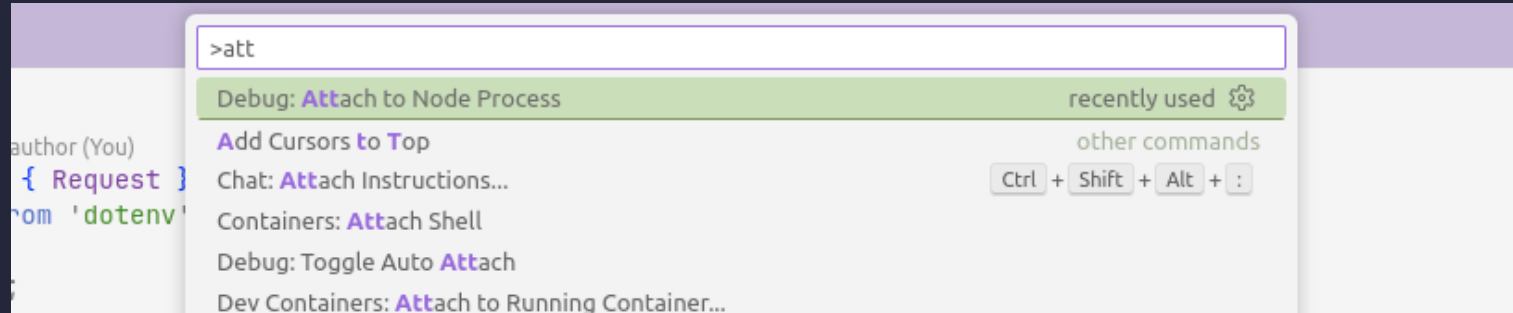
Définir un point d'arrêt

```
20
21 app.get('/fizzbuzz/:i', (req: Request<{i: number}>, res) => {
22   const i = req.params.i
23
24   let fb;
25
26   if(i % 15 == 0) {
27     fb = 'FizzBuzz'
28   }
29   else if(i % 3 == 0) {
30     fb = 'Fizz'
31   }
32   else if(i % 5 == 0) {
33     fb = 'Buzz'
34   }
35   else {
36     fb = `${i}`
37   }
38
39   res.send(fb)
40 }
```

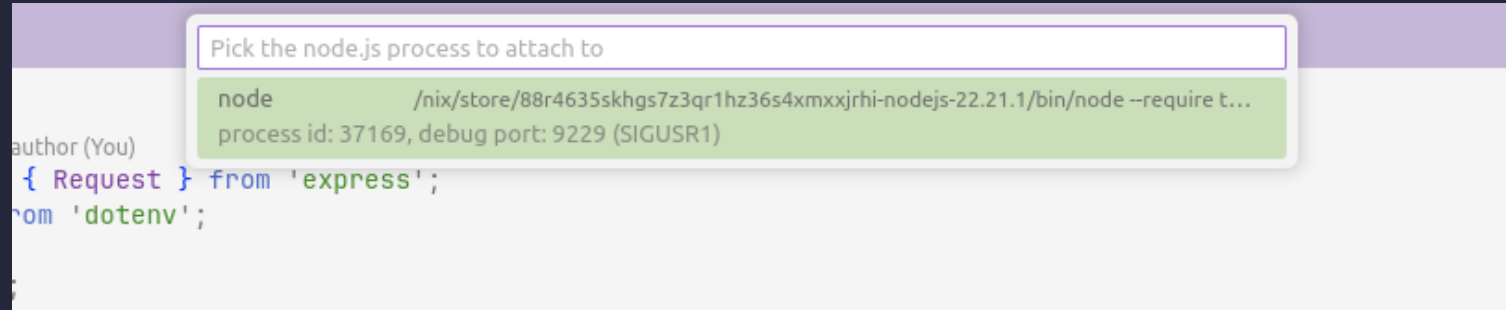
Click to add a breakpoint

You, last week • chore(2026): init

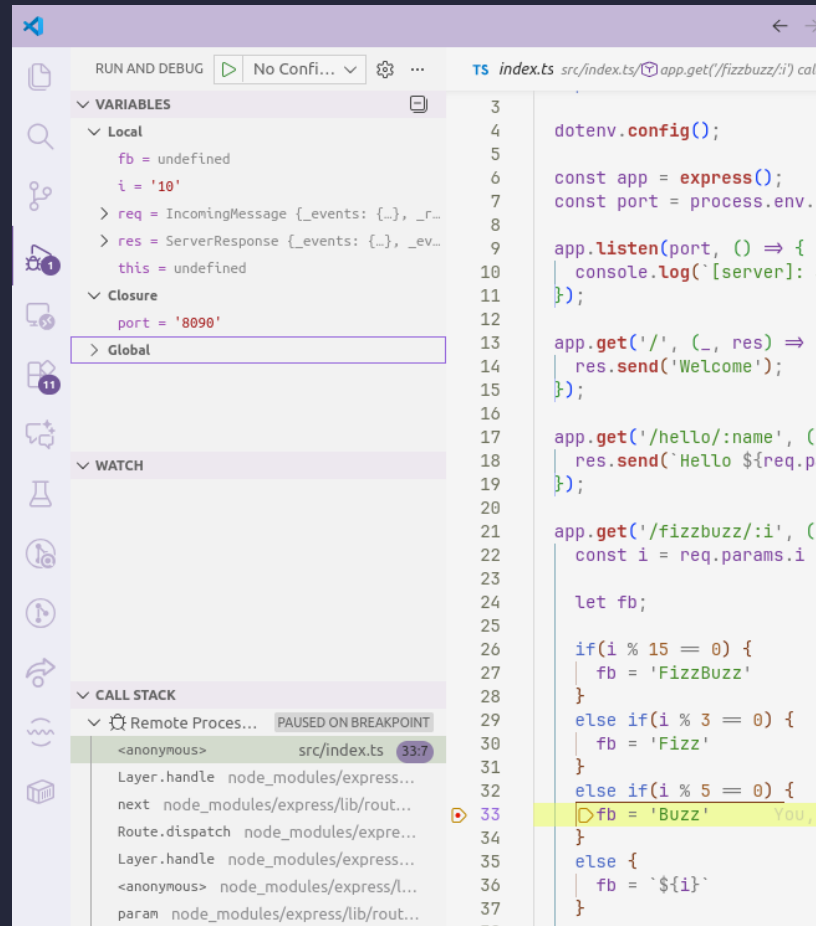
Démarrer le debugger



Sélectionner le serveur



Déclencher le point d'arrêt



Avec Java

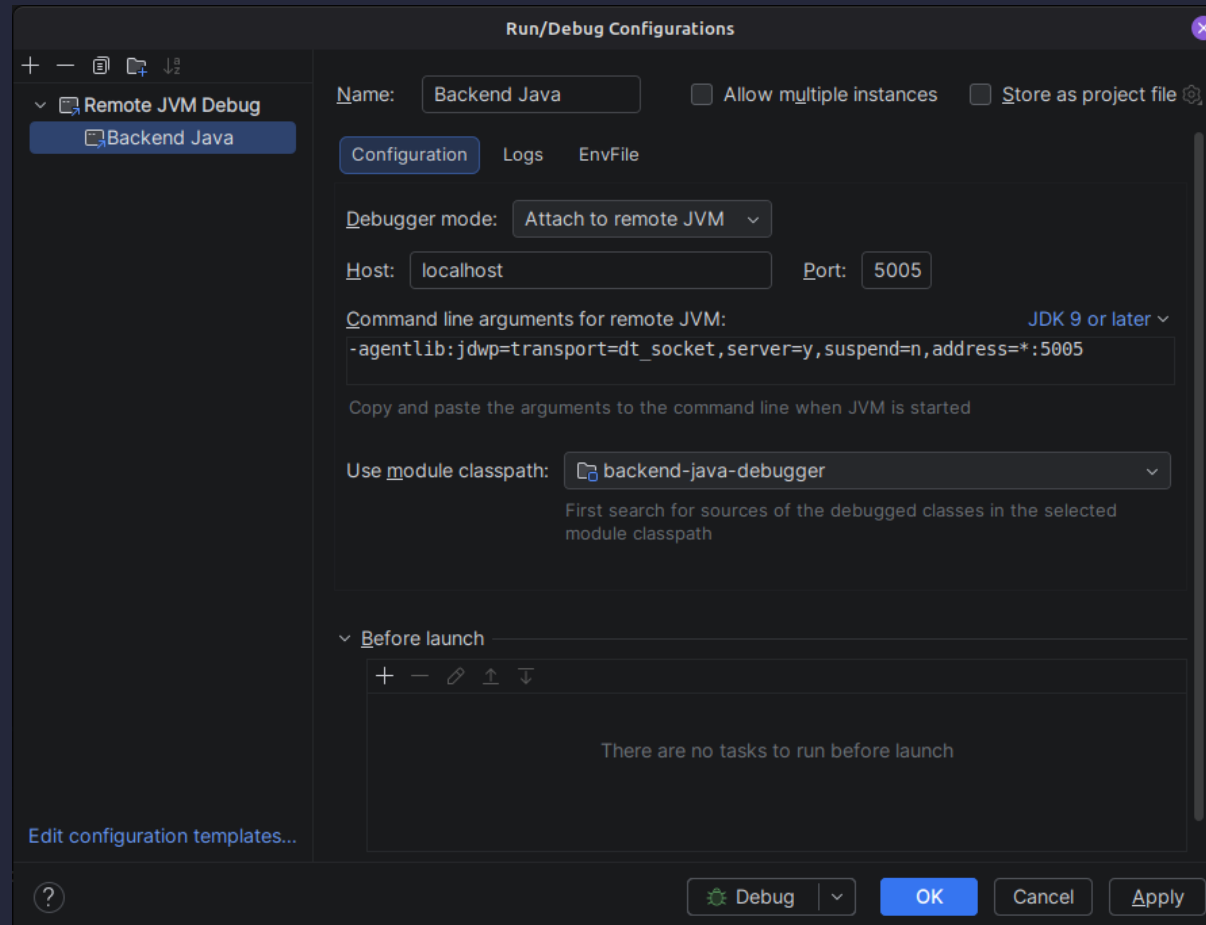
Ajouter l'agent au démarrage de l'application via l'option suivante :

```
-agentlib:jdwp=transport=dt_socket,server=y,suspend=n,address=*:5005
```

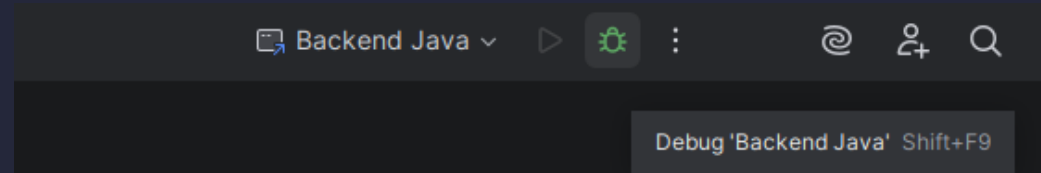
Définir un point d'arrêt

```
4  ▾  
3  
2  
1  ▾  
●  💡  
1  
2  
3  
4  
5  
6  
7  
8  
    .get( path: "/fizzbuzz/{i}", Context ctx → {  
        var value = Integer.parseInt(ctx.pathParam(s: "i"));  
  
        if (value % 15 == 0) {  
            ctx.result(resultString: "FizzBuzz");  
        } else if (value % 3 == 0) {  
            ctx.result(resultString: "Buzz");  
        } else if (value % 5 == 0) {  
            ctx.result(resultString: "Fizz");  
        } else {  
            ctx.result(Integer.toString(value));  
        }  
    })
```

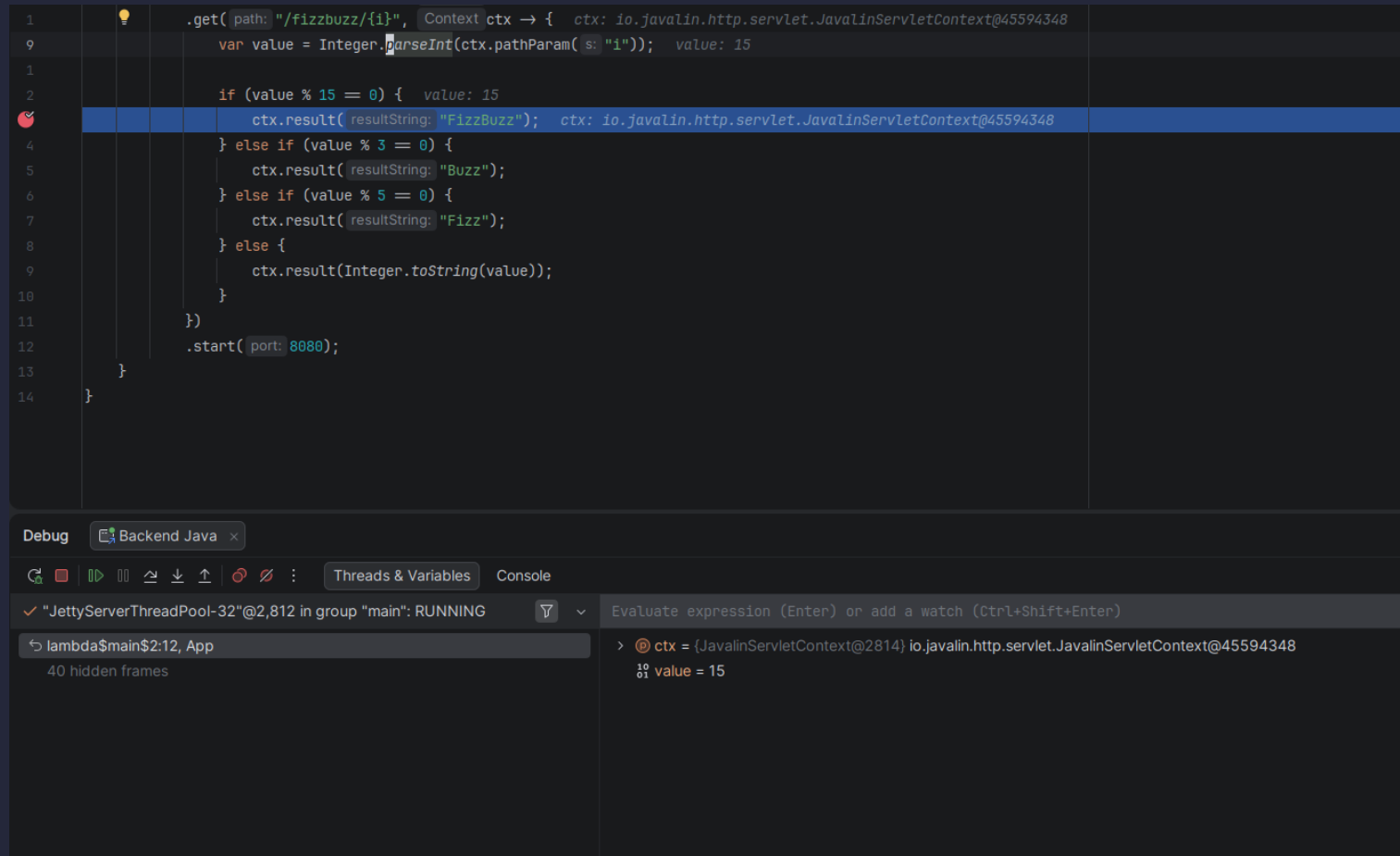
Configurer le debugger



Lancer le debugger



Déclencher le point d'arrêt



```
1  .get(path: "/fizzbuzz/{i}", Context ctx → { ctx: io.javalin.http.servlet.JavalinServletContext@45594348
9  var value = Integer.parseInt(ctx.pathParam(s: "i")); value: 15
1
2  if (value % 15 == 0) { value: 15
3  ctx.result(resultString: "FizzBuzz"); ctx: io.javalin.http.servlet.JavalinServletContext@45594348
4  } else if (value % 3 == 0) {
5  ctx.result(resultString: "Buzz");
6  } else if (value % 5 == 0) {
7  ctx.result(resultString: "Fizz");
8  } else {
9  ctx.result(Integer.toString(value));
10 }
11 })
12 .start(port: 8080);
13 }
14 }
```

Debug Backend Java x

Threads & Variables Console

✓ "JettyServerThreadPool-32"@2,812 in group "main": RUNNING

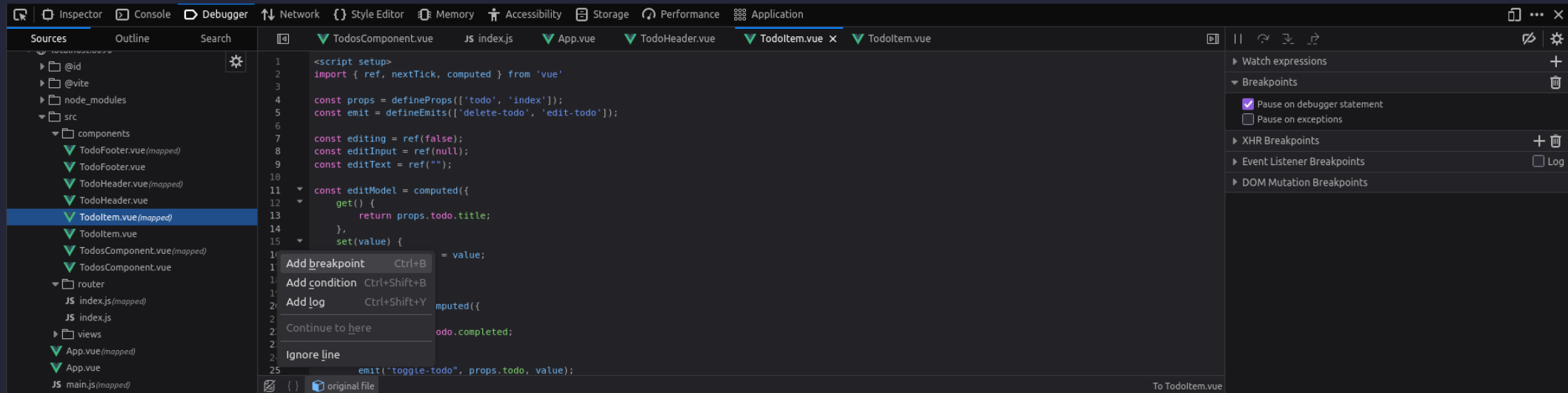
lambda\$main\$2:12, App
40 hidden frames

> @ ctx = {JavalinServletContext@2814} io.javalin.http.servlet.JavalinServletContext@45594348
value = 15

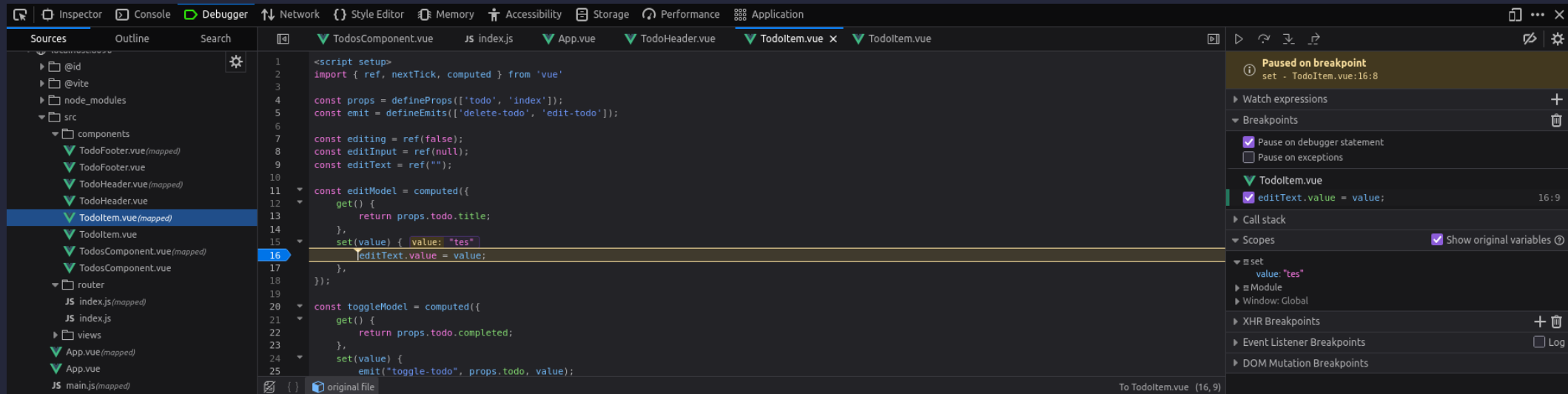
Avec Javascript

Ouvrir les outils développeur : F12

Définir un point d'arrêt



Déclencher le point d'arrêt



Merci de votre attention