

'25 백베콘 Winter

ㅋㅋㅋㅋㅋ
EZ



옆집 뽀뽀도 시작하는
오픈소스 입문

 Merged

Pengoose

어째서 발표?

오픈소스 기여 주기가 준내빠르시던데

어찌하는겁니까 그거

Opened 5 other pull requests in 5 repositories

vites
용?

- vitest-dev/vitest
- toss/suspensive
- RVC-Project/Retrieval-based-Voice-Conversion-WebUI
- fumiana/RVC-Models-Downloader
- vanilla-extract-css/vanilla-extract

toss/suspensive test(react, react-await): separate success & failure description

@suspensive/react @suspensive/react-await

#875 by pengooseDev was merged on May 14, 2024 · Approved 1 task done

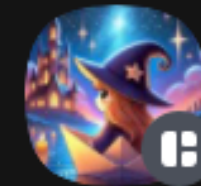
microsoft/playwright feat(testType): add support for test.fail.only method ✓

#33001 by pengooseDev was merged on Oct 16, 2024 · Approved

- 1 merged
- 1 merged
- 1 merged
- 1 merged
- 1 merged

오후 7:22

같이 오픈소스 기여



마녀

어케함?

백베콘발표부탁

오후 7:22

- 1
- 1

vitest-dev/vitest fix(runner): Ensure inner suite { sequential: true } correctly overrides outer suite { concurrent: true } ✓

#5737 by pengooseDev was merged on May 17, 2024 · Approved 6 tasks done

microsoft/playwright fix(expect): adjust normalization for regex values in toHaveText matcher ✗

vanilla-extract-css/vanilla-extract fix: change the order of function overloading(createTheme) ✓

#1410 by pengooseDev was merged on Jun 14, 2024 · Approved

jakobhoeg/shadcn-chat refactor: improve ChatInput handling for composition-languages ✓

#55 by pengooseDev was merged on Sep 28, 2024

line/line-bot-sdk-nodejs Refactoring Exception ✓

#739 by pengooseDev was merged on Mar 8, 2024 · Approved

microsoft/playwright feat: expect(locator).toHaveAccessibleErrorMessage ✓

#33904 by pengooseDev was merged 5 days ago · Approved

vitest-dev/vitest fix(runner): Ensure test.each print -0 and -NaN properly ✓

#5806 by pengooseDev was merged on Jun 3, 2024 · Approved 6 tasks done

toss/es-hangul test: 테스트 관심사에 따른 describ

#80 by pengooseDev was merged on Apr 27, 2024 · Approved

line/line-bot-sdk-nodejs Refactoring unit test(middleware.spec.ts) ✓

#744 by pengooseDev was merged on Mar 13, 2024 · Approved


RVC-Project/Retrieval-based-Voice-Conversion-WebUI

documentation

#2051 by pengooseDev was merged on May 12, 2024 6 tasks done

오픈소스에 관심을 가진 선생님들은 많다.
다만, 실제로 기여하시는 분들은 상대적으로 적다.


Why?

: 오픈소스는 어려워 보여요!

why?

“막연하기 때문”
(가이드 및 정보의 부재. 멘땅의 헤딩)

우리는 미지의 것에서 두려움을 느낌.
그렇기에 잘 시도 하지 않음.

: 오픈소스에 대한 막연함을 없애보
자!

펙구스의 실제 기여 사례들을 함께 살펴보며
오픈소스의 프로세스와 팁을 익혀 두려움을 없애자!

(해보면 생각보다 진짜 쉬움)

ㅋㅋㅋㅋㅋ
EZ



오픈소스의 프로세스

 Merged

전체 프로세스

70%



- * 기여할 프로젝트 찾기

- * 이슈 찾기 또는 생성
- 코드 분석
- 해결하기
- 회귀 테스트 작성
- 빌드 및 전체 테스트 통과 체크
- PR 작성
- 리뷰(반복)
- merged 🚀

ㅋㅋㅋㅋㅋ
EZ



 Merged

기여할 프로젝트 찾기

1. 기여할 프로젝트 찾기

기여하기 좋은 프로젝트 = 살아있는 프로젝트

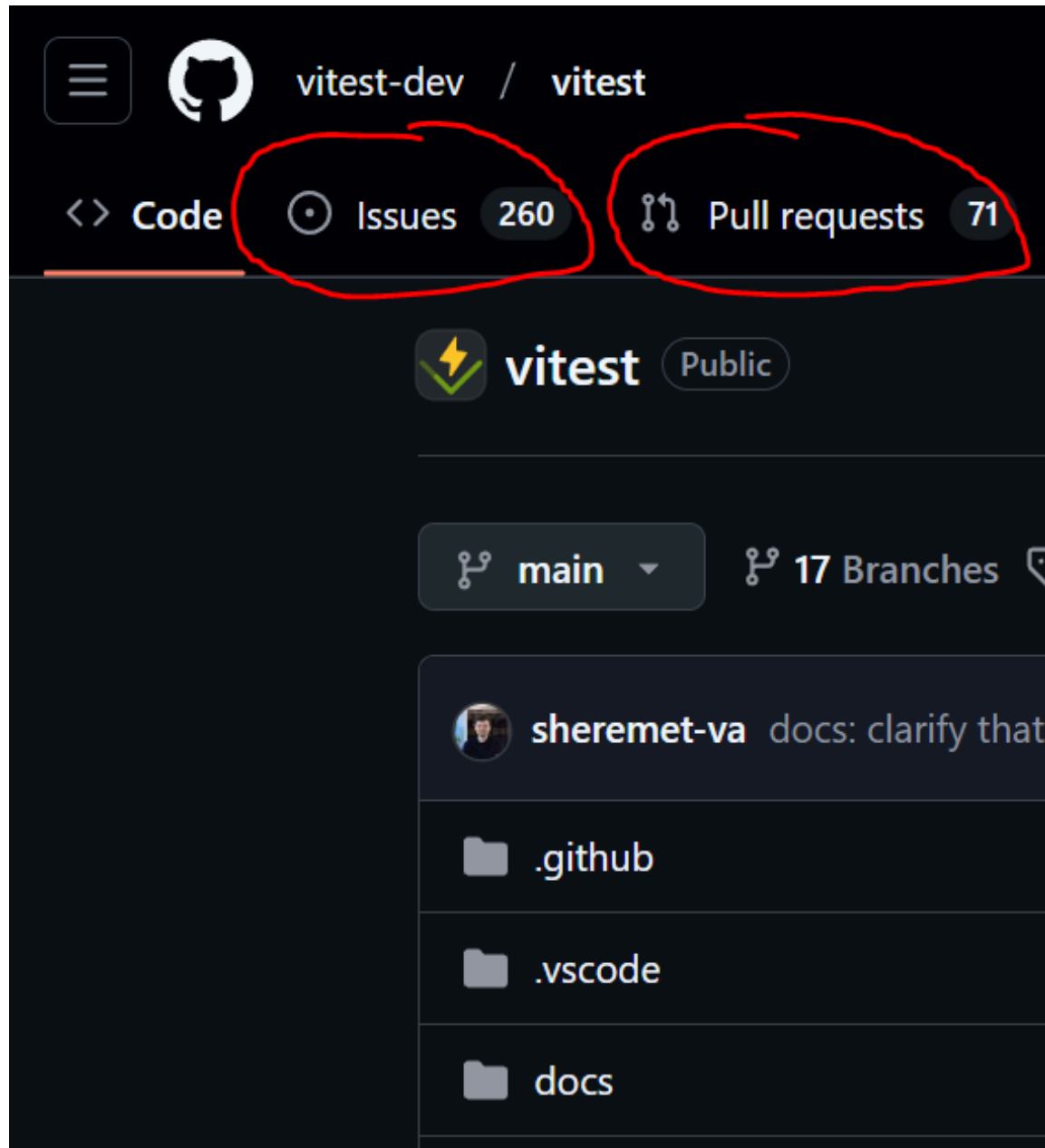
- Issue가 많은 프로젝트
- 이슈의 라벨링(태그)이 잘 붙어있는 프로젝트
- closed된 PR 중 외부인이 많은 프로젝트

+a 오래된 Issue가 잘 관리되는 프로젝트(stale하면 closed)

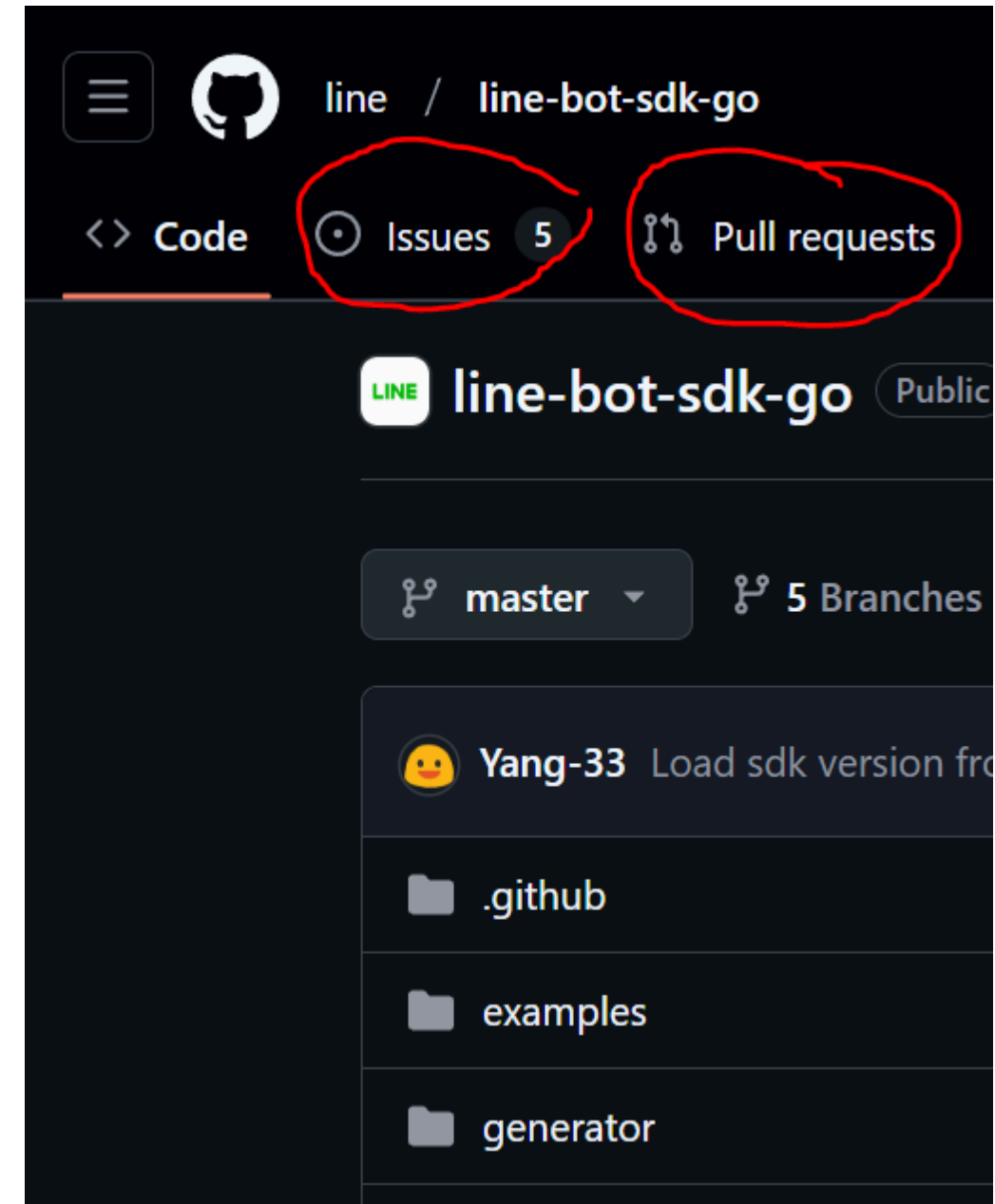
1. 기여할 프로젝트 찾기 > Issue가 많은 프로젝트

Issue가 많은 프로젝트

😊 Good!



😞 Bad



1. 기여할 프로젝트 찾기 > Issue가 많은 프로젝트

Issue가 많은 프로젝트 - Quiz

😊 Good?

😞 Bad?

The screenshot shows the GitHub repository page for 'jest' by 'jestjs'. The repository is public and has 287 issues, 71 pull requests, 560 watchers, 6.5k forks, and 44.4k stars. The main branch is selected, and there are 16 branches and 296 tags. The repository description is 'Delightful JavaScript Testing.' and the website is 'jestjs.io'. The repository includes tags for 'javascript', 'testing', 'facebook', 'snapshot', 'expectation', 'easy', 'painless-javascript-testing', 'immersive', and 'painless'. The repository also has a README and a MIT license.

jestjs / jest

Type / to search

<> Code Issues 287 Pull requests 71 Actions Projects Wiki Security Insights

jest Public Sponsor Watch 560 Fork 6.5k Starred 44.4k

main 16 Branches 296 Tags Go to file Add file Code

liuxingbaoyu Improve babel-plugin-jest-hoist logic (#15415) 611d1a4 · last month 7,275 Commits

.circleci	chore(deps): update node orb to v5.2.0 (#14833)	2 years ago
.eslintplugin	Update copyrights with Meta Platforms, restore original lice...	2 years ago
.github	chore: update yarn (#15394)	last month
.vscode	Add OpenJS Logo And Footer (#14904)	11 months ago
.yarn	chore: update yarn (#15394)	last month

About

Delightful JavaScript Testing.

jestjs.io

javascript testing facebook snapshot expectation easy painless-javascript-testing immersive painless

Readme MIT license

1. 기여할 프로젝트 찾기 > Issue가 많은 프로젝트

🙄 **Bad**

- 마지막 merge가 1달 전
- Star에 비해 PR 및 Issue수가 적음. (vitest 13.5k에 260 issue)

jestjs / jest

Type / to search

<> Code Issues 287 Pull requests 71 Actions Projects Wiki Security Insights

jest Public

Sponsor Watch 560 Fork 6.5k Starred 44.4k

main 16 Branches 296 Tags

Go to file Add file Code

liuxingbaoyu Improve babel-plugin-jest-hoist logic (#15415) 511d1a4 · last month 7,275 Commits

.circleci	chore(deps): update node orb to v5.2.0 (#14833)	2 years ago
.eslintplugin	Update copyrights with Meta Platforms, restore original lice...	2 years ago
.github	chore: update yarn (#15394)	last month
.vscode	Add OpenJS Logo And Footer (#14904)	11 months ago
.yarn	chore: update yarn (#15394)	last month

About

Delightful JavaScript Testing.

jestjs.io

javascript testing facebook snapshot expectation easy

painless-javascript-testing immersive

painless

Readme

MIT license

1. 기여할 프로젝트 찾기 > Issue 라벨링

Issue의 특성이 잘 라벨링 되어있는 프로젝트

😊 Good!

The screenshot shows a list of GitHub issues. The following labels are highlighted with red circles or lines:

- Issue #7083: `p3-minor-bug`
- Issue #7079: `feat: browser` and `p4-important`
- Issue #7073: `p3-minor-bug`

pending triage:

논의중.

아주 쉽다면 PR 날려봄직 함.
but, 버려질 수 있음.

p4-important:

리뷰가 백셀 가능성이 높음.

쉬워보인다면 도전.

but 첫 기여로 권장하진 않음.

p3-minor:

첫 시도로 부담이 적어서 좋음.

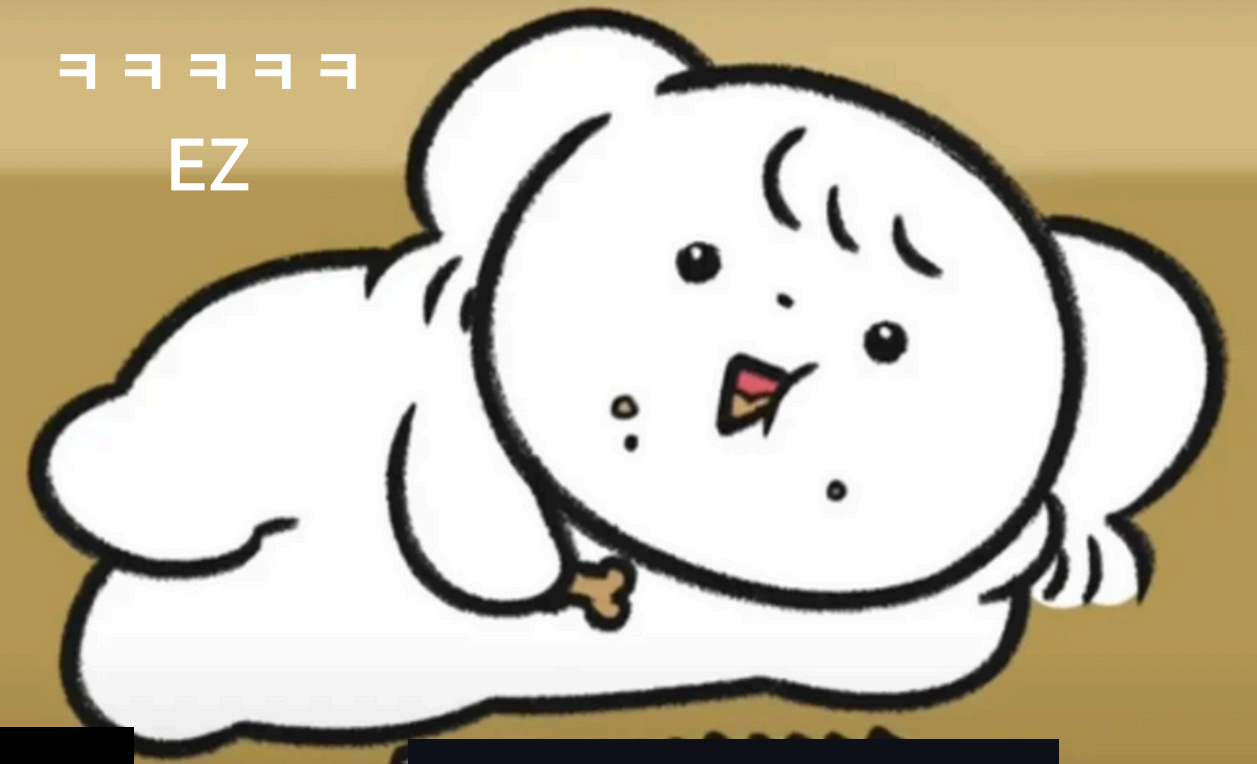
1. 기여할 프로젝트 찾기 > closed된 PR 중 외부인이 많은 프로젝트

- Version Packages** ✓
#1426 by vanilla-extract-ci was merged on Jun 14, 2024 • Approved
- css: Fix conditional rule merging bug** ✓
#1425 by askoufis was merged on Jun 14, 2024 • Approved
- Webpack vite compiler wip** ✓
#1423 by askoufis was closed on Jun 8, 2024 • Review required
- Convert all @vanilla-extract deps and dev** ✓
#1422 by askoufis was merged on Jun 6, 2024 • Approved
- Upgrade to pnpm v9** ✓
#1421 by askoufis was merged on Jun 6, 2024 • Review required
- Refactor css utils calc classes** ✓
#1419 by rowan-gud was closed on Jun 21, 2024 • Draft
- Version Packages** ✓
#1413 by vanilla-extract-ci was merged on May 22, 2024 • Approved

- fix: change the order of function overloading(createTheme)** ✓
#1410 by pengooseDev was merged on Jun 14, 2024 • Approved
- Version Packages** ✓
#1407 by vanilla-extract-ci was merged on May 4, 2024 • Approved
- addFileScope : Don't mangle non-template strings that contain newlines** ✓
#1406 by askoufis was merged on May 4, 2024 • Review required
- test, ci: Playwright config tweaks, update action versions** ✓
#1401 by askoufis was merged on Apr 30, 2024 • Review required
- Version Packages** ✓
#1400 by vanilla-extract-ci was merged on Apr 30, 2024 • Approved
- vite: Ensure compiler is re-used between builds** ✓
#1399 by askoufis was merged on Apr 30, 2024 • Review required
- Version Packages** ✓
#1398 by vanilla-extract-ci was merged on Apr 29, 2024 • Approved



ㅋㅋㅋㅋㅋ
EZ



사례와 함께 살펴보는
기여 프로세스

 Merged

사례와 함께 살펴보는 기여 프로세스

- * 이슈 찾기
- 코드 분석
- 해결하기
- 회귀 테스트 작성
- 빌드 및 전체 테스트 통과 체크
- PR 작성
- 리뷰(반복)
- merged 🚀

엥? ㅋㅋ
이따다키마스-



Issue에 답이 있어요!
[Vanilla-extract]편

 Merged

[Issue]
[PR]

[#1267]:

createTheme() 함수가 잘못된 오버로드 시그니처와 매핑 되어, TypeScript 자동 완성 기능이 작동하지 않습니다. #1267

Typescript can't resolve which overload to use in createTheme() so we don't have autocomplete. #1267

Closed

2 tasks done

nasheomirro opened this issue on Dec 30, 2023 · 1 comment · Fixed by #1410

nasheomirro commented on Dec 30, 2023

Describe the bug

Typescript doesn't seem to resolve to the correct overload signature when calling createTheme() :

```
// try this on the reproduction link
const vars = createThemeContract({
  a: null,
  b: null
});

const theme = createTheme(vars, {
  // typescript doesn't give suggestions
})
```

Labels

priority-low

type issue

Issue에 답이 있는 경우 > 1. Issue 찾기

Possible solutions

This one can be done on the user's side. Create another function that just returns `createTheme()` but will always expect two arguments, `contract` and `token`. This is a little tedious cause vanilla-extract doesn't export the internal types needed so I think the only way is to infer/copy those types somehow.

```
// imagine I infered the types correctly
function createThemeFrom<T extends Contract>(
  contract: T,
  tokens: MapLeafNodes<T, string>,
  id?: string
) {
  return createTheme(contract, tokens, id);
}
```

Another one just for the user is to steal the `MapLeafNodes<T, Leaf>` utility and make your own:

```
type Theme<T> = MapLeafNodes<T, string>

// then use it like:
const vars = createThemeContract({ a: null, b: null });
const lightTheme: Theme<typeof vars> = {
  // ...
}

const theme = createTheme(vars, lightTheme);
```

A weird but easy fix I found was to change the order of the function arguments. Maybe when errors are raised from multiple overload matches, the

```
// simplified example:
type JustMaps<T> = { [K in keyof T]: T[K] };

declare function example<T>(t: T, obj: JustMaps<T>, id?: string): void;
declare function example<T>(t: T, id?: string): void;
```

Reproduction

<https://www.typescriptlang.org/play?ts=4.5.5#code/JYWwDg9gTglgBAAbzgYygUwIYzQFQBZohoA0K6WuBRAwhAHYxQbLwC+cAZIBCHAEQABAG4Y6wADbiMAVRsxgB6ZAGc1fANwAoHcnpr4lqGrgBeMpmz5CaWgyYsAFah1wPcDAC44dAK6SxO6eAEa+AZI6rACUuvqG8DBUaBZWFLZEziEcMrKcMlo6HDAZnQQXv4wEAbg4mjY0XF6OkVwaqBg4sAcwGgAJnDyGE1o3jowAJ5gqQBS-kYAshhgaagA8QAB8alIqA2qDSEYPrwANZosxAccDgAur44lw9wrPHDaMhS5Pz+dBYwHoYzkEz6aB2u2cMGenGAWwA->



nasheomirro commented on Dec 30, 2023

Describe the bug

Typescript doesn't seem to resolve to the correct overload s

```
// try this on the reproduction link
const vars = createThemeContract({
  a: null,
  b: null
});

const theme = createTheme(vars, {
  // typescript doesn't give suggestions
})
```

I think its because `typeof vars` can extend both `Tokens` and `Contract`. Since `Contract` is the first argument passed, so it relies on the second argument. In that part, the user's mistake can either be that the user didn't pass an object with missing/incorrect properties.

Issue에 답이 있는 경우 > 1. Issue 찾기

v4.5.5 ▾ Run Export ▾ Share

```
1 import { createTheme, createThemeContract } from "@vanilla-extract/css";
2
3 const vars = createThemeContract({
4   a: null,
5   b: null
6 });
7
8 const theme = createTheme(vars, {
9   // there is no autocomplete
10  });
11
12
13 // simplified example:
14 type JustMaps<T> = { [K in keyof T]: T[K] };
15
16 declare function example<T>(t: T, id?: string): void
17 declare function example<T>(t: T, obj: JustMaps<T>, id?: string): string
18
19 const result = example({ a: '' }, {
20   // same problem
21  });
22
```

// weirdly though, this works:
declare function example2<T>(t: T, obj: JustMaps<T>,
declare function example2<T>(t: T, id?: string): void

```
const working = example2({ a: '' }, {
  // has autocomplete!
});
```

Write

Preview

버그있음

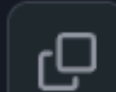
createTheme 매개변수가 다른 함수 시그니처(Tokens와 Contract)까지 만족함.
그래서 자동완성 안됨.

해결책

오버로드 순서를 바꾸면 될 듯?

```
declare function example<T>(t: T, obj: { [K in keyof T]: T[K] }, id?: string): string
declare function example<T>(t: T, id?: string): void

const result = example({ a: '' }, {
  // 돌아감!
});
```



구현 예시 링크

여기 버그, 해결책 재현한 [> TS 예시 링크임](#)

- ~~이슈 찾기~~
- ~~코드 분석~~
- ~~문제 해결하기~~
- ~~회귀 테스트 작성~~
- ~~빌드 및 전체 테스트 통과 체크~~
- PR 작성
- 리뷰(반복)
- merged 🚀

이슈 잘 찾았더니



- ~~이슈 찾기~~
- ~~답지보고 해결~~
- ~~답지보고 해결~~
- ~~답지보고 해결~~
- ~~답지보고 해결~~
- PR 작성
- 리뷰(반복)
- merged 🚀



pengooseDev commented on May 11, 2024 • edited by askoufis ▾

Contributor

PR Context

Fixes [#1267](#)

```
44 - export function createTheme<ThemeTokens extends Tokens>(
45 -   tokens: ThemeTokens,
46 -   debugId?: string,
47 - ): [className: string, vars: ThemeVars<ThemeTokens>];
```

```
48 export function createTheme<ThemeContract extends Contract>(
49   themeContract: ThemeContract,
50   tokens: MapLeafNodes<ThemeContract, string>,
51   debugId?: string,
52 ): string;
```

```
44 export function createTheme<ThemeContract extends Contract>(
45   themeContract: ThemeContract,
46   tokens: MapLeafNodes<ThemeContract, string>,
47   debugId?: string,
48 ): string;
```

```
49 + export function createTheme<ThemeTokens extends Tokens>(
50 +   tokens: ThemeTokens,
51 +   debugId?: string,
52 + ): [className: string, vars: ThemeVars<ThemeTokens>];
```


Fixed

```
const myVars = createThemeContract({
  color: {
    primary: null,
    secondary: null,
  },
});
```

```
const myTheme = createTheme(myVars, {
  color: {
    primary
  }
});
```

No overload match
No value exists in scope for the shorthand property

- primary (property) primary: string
- private method Private Method Definition
- prop Define a full property
- public method Public Method Definition



askoufis approved these changes on Jun 14, 2024

[View reviewed changes](#)

askoufis left a comment

[Contributor](#) ⋮

PR ㄱ. 함수 오버로드 콤수로 고친 거 같긴 한데, 이거 말곤 딱히 답 없을 듯.
완벽하진 않지만 뭐 어찌겠음.



askoufis enabled auto-merge (squash) 7 months ago



askoufis merged commit **99e974b** into `vanilla-extract-css:master` on Jun 14, 2024

5 checks passed

[View details](#)

[Revert](#)



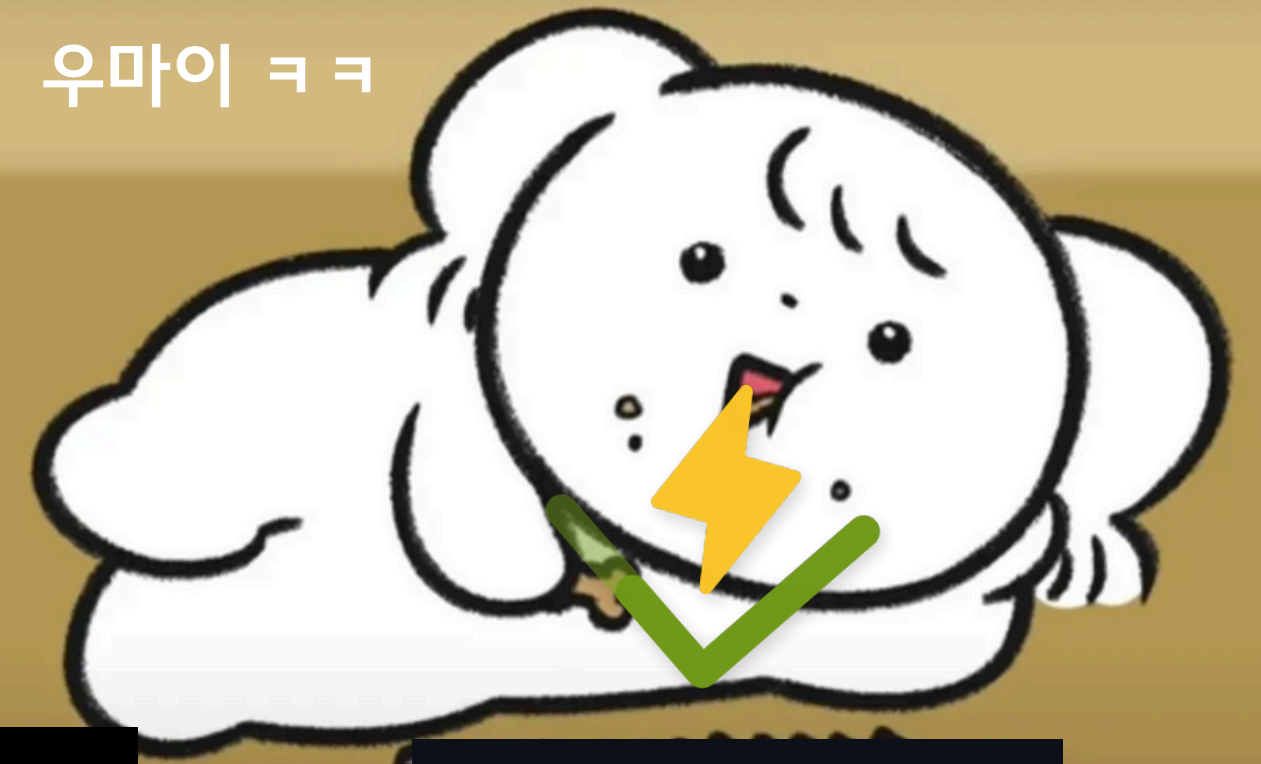
vanilla-extract-ci mentioned this pull request on Jun 14, 2024

Version Packages #1426



[Merged](#)

우마이 ㅋㅋ



Issue에 답이 있어요! 2
[Vitest]편

Merged

[Issue]

[PR](vitest PR은 잘 안열림)

[#1267]:

내부 { sequential: true }이 외부 { concurrent: true } 무시함 #5716



hi-ogawa commented on May 14, 2024

Contributor ...

Describe the bug

생각해보니까 describe 겹쳐서서 { 순서대로: true, 동시에: true } 같이 모순된 옵션 쓰면 어떻게 처리함..?
엣지케이스 테스트 터지는데..?

```
describe('repro', { concurrent: true }, () => {  
  // 이거 터짐.  
  describe('bad', { sequential: true }, () => { ... });  
});
```

Labels

p3-minor-bug

아마 아래 조건이랑 관련 있을 듯:

```
// 현재 suite에서 concurrent/sequential 상속  
options.concurrent = this.concurrent || (!this.sequential && options?.concurrent)  
options.sequential = this.sequential || (!this.concurrent && options?.sequential)
```

해결 못한 회귀테스트 일단 풀어주고

```
test/core/test/concurrent-suite.test.ts
```

119	checkSequentialTests() 120 } 121	119	checkSequentialTests() 120 } 121
122	- // TODO: not working? 123 - // describe('s-x-2', { sequential: true, }, () => { 124 - // checkSequentialTests() 125 - // })	122	+ describe('s-x-2', { sequential: true }, () => { 123 + checkSequentialTests() 124 + })
126	127 describe('s-y', () => { 128 checkParallelTests()	125	126 describe('s-y', () => { 127 checkParallelTests()

Nested 테스트의 상속구조 파악

```
286 if (currentSuite?.options)
287   options = { ...currentSuite.options, ...options }
288
289 - // inherit concurrent / sequential from current suite
290 - options.concurrent = this.concurrent || (!this.sequential &&
options?.concurrent)
291 - options.sequential = this.sequential || (!this.concurrent &&
options?.sequential)
292
293   return createSuiteCollector(formatName(name), factory, mode,
this.shuffle, this.each, options)
294 }
```

해결!

```
286 if (currentSuite?.options)
287   options = { ...currentSuite.options, ...options }
288
289 + // inherit concurrent / sequential from suite
290 + const isConcurrent = options.concurrent || (this.concurrent &&
!this.sequential)
291 + const isSequential = options.sequential || (this.sequential &&
!this.concurrent)
292 + options.concurrent = isConcurrent && !isSequential
293 + options.sequential = isSequential && !isConcurrent
294
295   return createSuiteCollector(formatName(name), factory, mode,
this.shuffle, this.each, options)
296 }
```

this (현재 테스트) currentSuite?.options (상위에서 명시적으로 주입된 옵션) options (현재 명시적으로 주입된 옵션)

각각의 concurrent, sequential가 undefined, null 등등 복잡도가 높아서 관리가 안된 사례

fix(runner): Ensure inner suite { sequential: true } correctly overrides outer suite { concurrent: true } #5737

Edit

<> Code

Merged sheremet-va merged 2 commits into vitest-dev:main from pengooseDev:fix-override-sequential-config on May 17, 2024

Conversation 0

Commits 2

Checks 16

Files changed 2

+8 -7



pengooseDev commented on May 17, 2024

Contributor

Description

PR은 이슈 링크, 변경사항 작성

Fixes [issue #5716](#)

This PR addresses an issue where an inner suite with { sequential: true } does not correctly override an outer suite with { concurrent: true }. Previously, the inheritance logic for options.concurrent and options.sequential led to unexpected behavior where conflicting options were not handled properly. This fix ensures that inner suites with { sequential: true } override outer suites with { concurrent: true }, providing the expected behavior for nested suite configurations.

- Adjust the logic in createSuiteCollector to properly handle conflicting concurrent and sequential options.
- Ensure that options.concurrent and options.sequential are inherited with clear default values, preventing undefined values.
- Update the task context to accurately reflect the intended concurrency behavior.

Please don't delete this checklist! Before submitting the PR, please make sure you do the following:

- It's really useful if your PR references an issue where it is discussed ahead of time. If the feature is substantial or introduces breaking changes without a discussion, PR might be closed.
- Ideally, include a test that fails without this PR but passes with it.
- Please, don't make changes to `npm-lock.yaml` unless you introduce a new test example.

Reviewers

sheremet-va ✓

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these

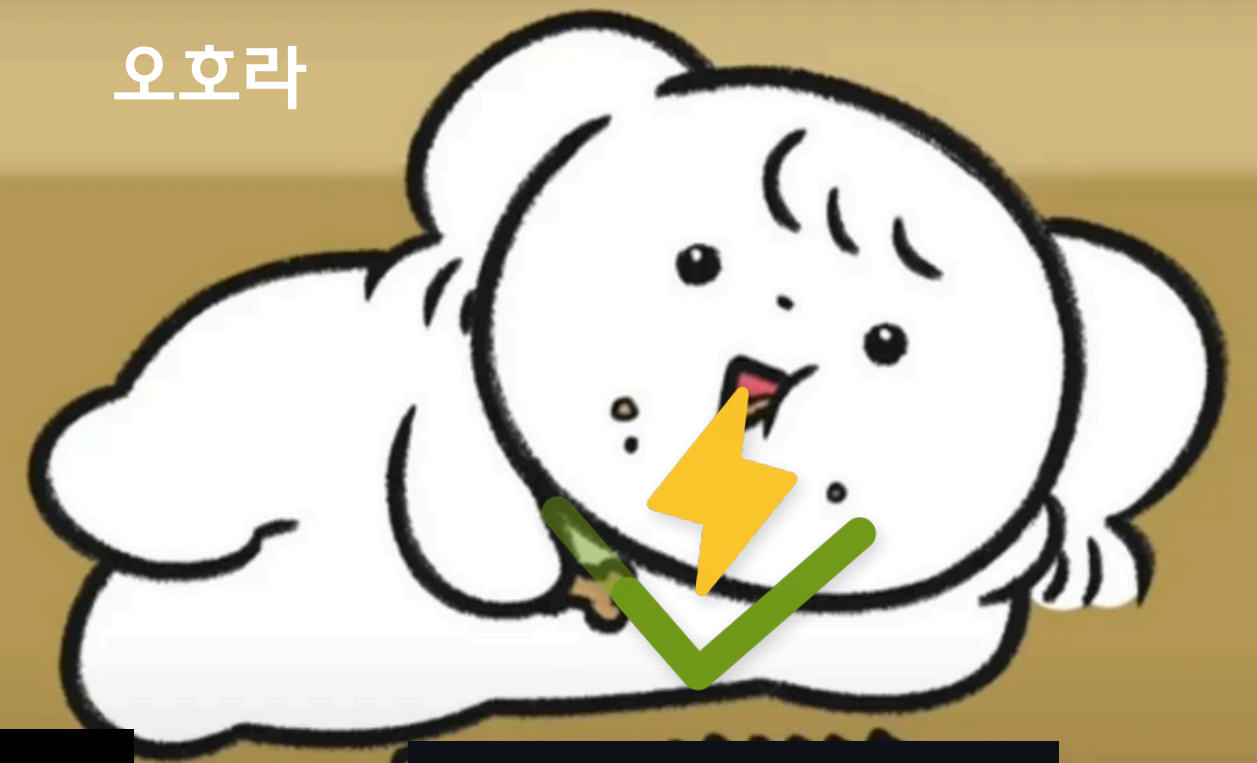
The screenshot shows a GitHub pull request merge notification. At the top, a green checkmark icon indicates approval by 'sheremet-va' on May 17, 2024. Below this, a purple merge icon shows 'sheremet-va' merging commit 'a20e75b' into 'vittest-dev:main', with '16 checks passed'. To the right, there are buttons for 'View reviewed changes' and 'View details'. A smiley face icon and a thumbs-up icon with the number '2' are also visible. At the bottom, a purple box contains the message: 'Pull request successfully merged and closed. You're all set — the pengooseDev:fix-override-seque... branch can be safely deleted. If you wish, you can also delete this fork of vittest-dev/vittest in the settings.' To the right of this box is a 'Delete branch' button.

sheremet-va approved these changes on May 17, 2024 [View reviewed changes](#)

sheremet-va merged commit `a20e75b` into `vittest-dev:main` [View details](#)
16 checks passed

Pull request successfully merged and closed
You're all set — the `pengooseDev:fix-override-seque...` branch can be safely deleted.
If you wish, you can also delete this fork of `vittest-dev/vittest` in the [settings](#). [Delete branch](#)

오호라



영향을 끼치는 범위는
최소화 해요

Merged

[Issue]

[PR1](vittest PR은 잘 안열림)

[PR2](vittest PR은 잘 안열림)

[#5744]: test.each에서 -NaN이 NaN으로 프린트 됨 #5716



arv commented on May 17, 2024

Describe the bug

When using `test.each` the descriptive text for `-0` and `-NaN` is printed as `0` and `NaN`.

Reproduction

<https://stackblitz.com/edit/vitest-dev-vitest-gkmdyn?file=test%2Fbasic.test.ts>

```
import { test, expect } from 'vitest';

test.each([[0], [-0], [NaN], [-NaN], [Infinity], [-Infinity]] as const)(
  '%f',
  (n) => {
    expect(n).toBe(n);
  }
);
```

The terminal output from this is:

```
✓ test/basic.test.ts (6)
  ✓ 0
  ✓ 0
```

Assignees

No one assigned

Labels

p2-edge-case

Projects

None yet

Milestone

No milestone

Development

Open in Workspace

Successfully merging a pull request may close this issue.

fix(runner): Ensure test.each print -0 and -NaN p...
pengooseDev/vitest

fix(utils): Ensure test.each print -0 and -NaN pro...
pengooseDev/vitest

테스트 그대로 좁기(회귀 테스트 추가)

```
298 +
299 + describe('isNegativeNaN', () => {
300 +   test.each`
301 +     value | expected
302 +     ${Number.NaN} | ${false}
303 +     ${-Number.NaN} | ${true}
304 +     ${0} | ${false}
305 +     ${-0} | ${false}
306 +     ${1} | ${false}
307 +     ${-1} | ${false}
308 +     ${Number.POSITIVE_INFINITY} | ${false}
309 +     ${Number.NEGATIVE_INFINITY} | ${false}
310 +     `('isNegativeNaN($value) -> $expected', ({ value, expected }) => {
311 +       expect(isNegativeNaN(value)).toBe(expected)
312 +     })
313 + })
```

test.each 테스트 파일 찾아서
기존 컨벤션 따라가기

```
describe('objDisplay', () => {
  test.each`
    value | expected
    ${'a'.repeat(100)} | ${`'${'a'.repeat(37)}...'`}
    ${'🐱'.repeat(100)} | ${`'${'🐱'.repeat(18)}...'`}
    ${`a${'🐱'.repeat(100)}...`} | ${`a${'🐱'.repeat(18)}`}
  `('Do not truncate strings anywhere but produce valid
  for $value', ({ value, expected }) => {
    // encodeURIComponent can be used to detect invalid strings
    // code-points
    // note: our code should not split surrogate pairs
    // graphemes
    expect(() => encodeURIComponent(objDisplay(value))).not.toThrow()
    expect(objDisplay(value)).toEqual(expected)
  })
})
```

로직 파악

```
219 +  
220 + export function isNegativeNaN(val: number) {  
221 +   if (!Number.isNaN(val))  
222 +     return false  
223 +   const f64 = new Float64Array(1)  
224 +   f64[0] = val  
225 +   const u32 = new Uint32Array(f64.buffer)  
226 +   const isNegative = u32[1] >>> 31 === 1  
227 +  
228 +   return isNegative  
229 + }
```

해결?

모든 파일에서 쓰는 코어 모듈(format)
해당 모듈의 %f 포매팅에서 해당 로직 처리!

커스텀 유틸 isNegativeNaN 추가하여 검증 로직 추가

영향을 끼치는 범위는 최소화 해요



sheremet-va commented on May 28, 2024

Member ...

Issue [#5744](#) requests that `-0` and `-NaN` should be rendered as `-0` and `-NaN` respectively when formatted with `%f`, instead of `0` and `NaN`. Considering the `pending triage` label has been removed and the `p2-edge-case` label has been added, I understood this as an indication of support for this feature.

`.each` 변경은 괜찮지만, Node.js 포맷은 깨질 수 있는 Breaking Change라서 이 방법은 안 됨. 우회 방법 있으면 좋겠음.

Why ArrayBuffer?

I am fine with any implementation of `-NaN` detection.



영향을 끼치는 범위는 최소화 해요

```
packages/runner/src/suite.ts
430     template = template
431         .replace(/%/g, '__vitest_escaped_%__')
432         .replace(/%/g, `${idx}`)
433         .replace(/__vitest_escaped_%__/g, '%')
434     }
435     const count = template.split('%').length - 1
436 +
437 +     if (template.includes('%f')) {
438 +         const placeholders = template.match(/%f/g) || []
439 +         placeholders.forEach((_, i) => {
440 +             if (isNegativeNaN(items[i]) || Object.is(items[i], -0)) {
441 +                 // Replace the i-th occurrence of '%f' with '-%f'
442 +                 let occurrence = 0
443 +                 template = template.replace(/%f/g, (match) => {
444 +                     occurrence++
445 +                     return occurrence === i + 1 ? '-%f' : match
446 +                 })
447 +             }
448 +         })
449 +     }
450 +
451     let formatted = format(template, ...items.slice(0, count))
452     if (isObject(items[0])) {
453         formatted = formatted.replace(
```

의존성 및 사이드 이펙트가 가장 적은 위치

Suite 내부의 Title 렌더링 함수

View로 포매팅 로직 이동

영향을 미치는 범위는 최소화 해요

fix(runner): Ensure test.each print -0 and -NaN properly #5806

Edit

<> Code

Merged sheremet-va merged 6 commits into vitest-dev:main from pengooseDev:fix-format on Jun 3, 2024

Conversation 1

Commits 6

Checks 16

Files changed 4

+45 -3



pengooseDev commented on Jun 1, 2024

Contributor

Description

Fixes [issue #5744](#)

[Prev PR\(closed\)](#)

This bug fix ensures that when using test.each, the descriptive text for `-0` and `-NaN` is correctly printed as `-0` and `-NaN` instead of `0` and `NaN`.

Reviewers

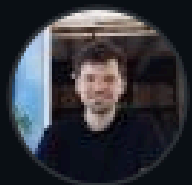
sheremet-va ✓

Assignees

No one assigned

Labels

None yet



sheremet-va approved these changes on Jun 3, 2024

[View reviewed changes](#)

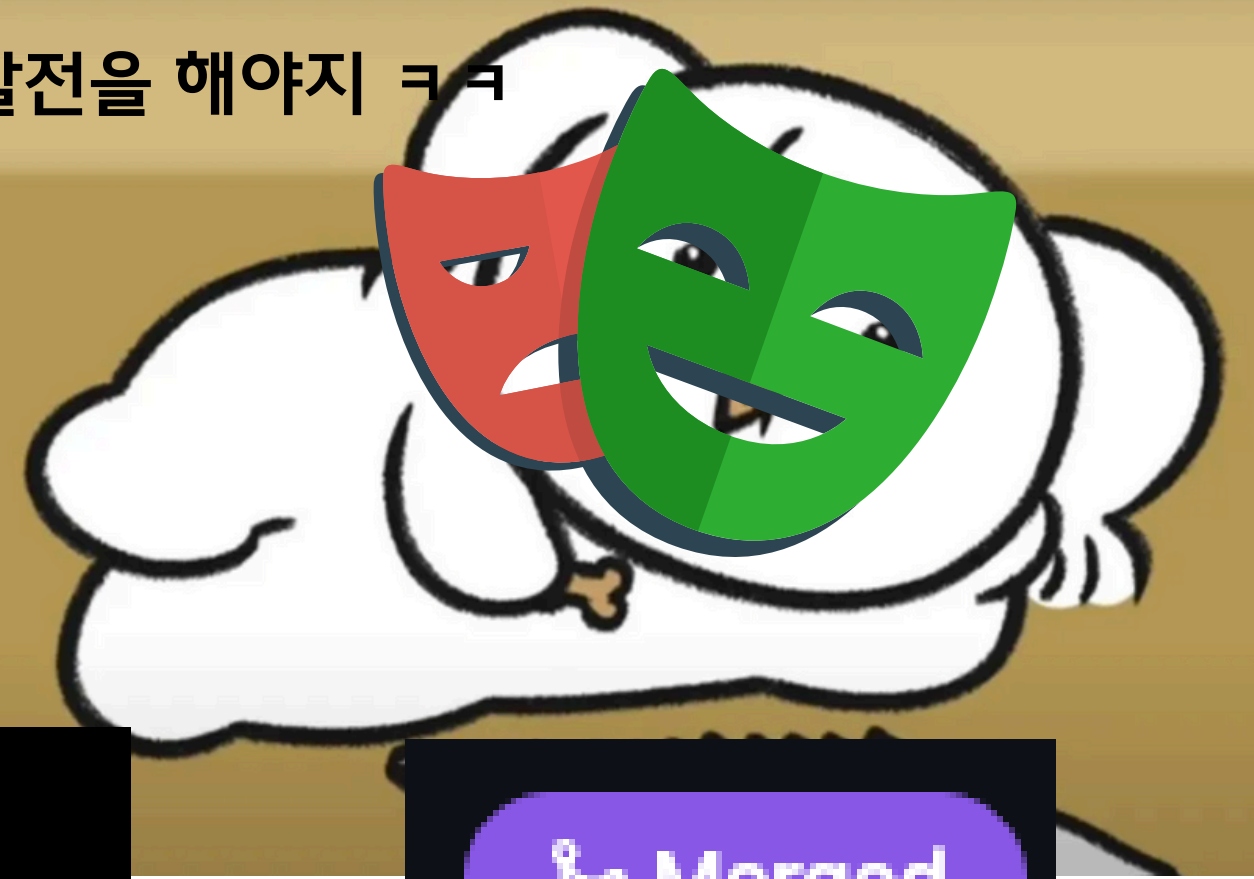


sheremet-va merged commit `9ac8ff9` into `vitest-dev:main` on Jun 3, 2024
16 checks passed

[View details](#)

2. Manually verifying without adding a test suite.

인간은 발전을 해야지 ㅋㅋ



배웠던 건 써먹어요
(부재: 사실 까보면 안어려워요)

[Issue]
[PR]

배웠던 건 써먹어요(부재: 어려워 보여도 사실 쉬워요)

[#29382]: expect().toHaveText() 정규표현식 오류 #29382

[Bug]: Diff message of `expect().toHaveText()` is hard to read when there are matching and non-matching regex in expected texts #29382

New issue

Closed

#33533



tui95 opened on Feb 6, 2024 · edited by tui95

Edits ...

Assignees

No one assigned

Steps to reproduce

`toHaveText()` 에서 정규표현식 무조건 틀렸다고 나옴.

st-runner

Example

```
import { expect, test } from "@playwright/test";

test("toHaveText()", async ({ page }) => {
  page.setContent(`
    <ul>
      <li>Coffee</li>
      <li>Tea</li>
      <li>Milk</li>
```



배웠던 건 써먹어요(부재: 어려워 보여도 사실 쉬워요)

[#29382]: expect().toHaveText() 정규표현식 오류 #29382

Expected behavior

정규식이 매칭되면, 당연히 맞았다고 판단해야 함.

Actual behavior

지금은 매칭된 정규식도 틀렸다고 Diff 메시지 띄움.

```
Error: Timed out 5000ms waiting for expect(locator).toHaveText(expected)
```

```
Locator: getByRole('listitem')
```

```
- Expected - 2
```

```
+ Received + 2
```

```
Array [
```



```
  "Coffee",
```


```
- /\d+/,
```

```
- /Milk/,
```

```
+ "Tea",
```


배웠던 건 써먹어요(부재: 어려워 보여도 사실 쉬워요)

  dgozman added `feature-test-runner` `feature-expect` on Feb 7, 2024

 Nickheythatsme on Feb 16, 2024

@dgozman ㅎㅎ, 이거 제가 해도 됨? 저 첫기여인데 이거 쉬워보이는듯.




 dgozman on Feb 16, 2024

Contributor ...

@Nickheythatsme

이 이슈 꽤 어려울 듯. diff 메시지는 외부 라이브러리 쓰는데, 이게 커스텀 Diff 지원 안함.
메시지 모듈 직접 구현하기엔 쉽지 않음.



 Nickheythatsme on Feb 17, 2024

@dgozman 아하, 다른 거 할게요!



배웠던 건 써먹어요(부재: 어려워 보여도 사실 쉬워요)

Maintainer 픽셀 어려움을 “수” 있음 > Issue 9개월 방치



Nickheythatsme on Feb 17, 2024

@dgozman 아하, 다른 거 할게요!



9개월 방치되었지만, 코드 뜯어보니 **꿀이슈**였음.
생각보다 실제로 코드 뜯어보는 사람들 X

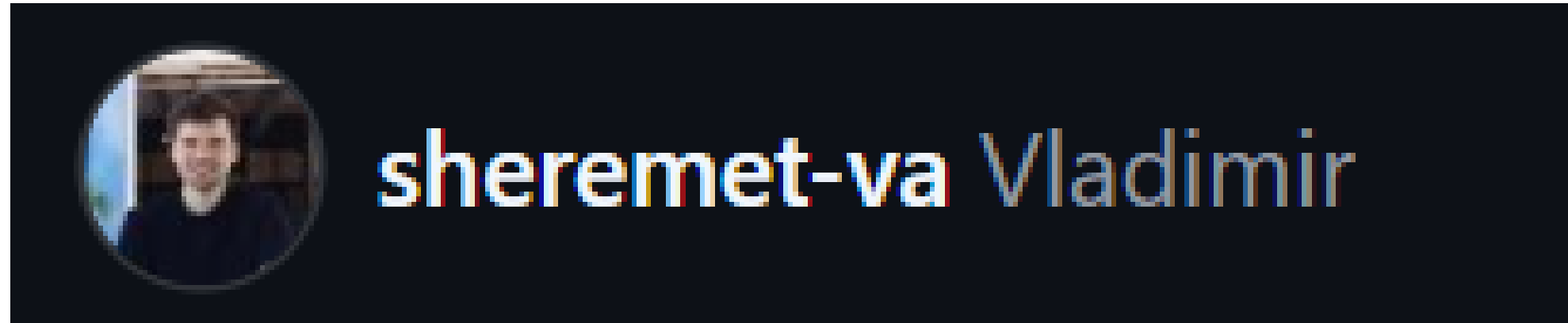


pengooseDev mentioned this on Nov 11, 2024

[fix\(expect\): adjust normalization for regex values in toHaveText matcher #33533](#)



yury-s closed this as completed in [#33533](#) on Nov 12, 2024



Vitest Maintainer 형님에게 배웠던 로직과 완전 동일.

View Layer 함수(여기선 외부 모듈)
invoke 직전에 값을 조정하여 주입

> 변경점 및 사이드 이펙트 최소화 & 이슈 해결

테스트 그대로 줍줍

```
619 + test('should only highlight unmatched regex in diff message for
    + toHaveText with array', async ({ runInlineTest }) => {
620 +   const result = await runInlineTest({
621 +     'a.spec.ts': `
622 +       import { test, expect } from '@playwright/test';
623 +
624 +       test('toHaveText with mixed strings and regexes (array)', async ({
    + page }) => {
625 +         await page.setContent(`
626 +           <ul>
627 +             <li>Coffee</li>
628 +             <li>Tea</li>
629 +             <li>Milk</li>
630 +           </ul>
631 +         `);
632 +
633 +         const items = page.locator('li');
634 +         await expect(items).toHaveText(['Coffee', /\d+/, /Milk/]);
635 +       });
636 +     },
637 +   });
638 +   expect(result.exitCode).toBe(1);
639 +   const output = result.output;
640 +   expect(output).toContain('- \d+');
641 +   expect(output).toContain('+ "Tea",');
642 +   expect(output).not.toContain('- /Milk/');
643 +   expect(output).not.toContain('- "Coffee",');
644 + });
```

배웠던 건 써먹어요(부재: 어려워 보여도 사실 쉬워요)

문제 파악

```
export function toHaveText(  
  // ... codes  
) {  
  // 🚩 1. 배열인 경우.  
  if (Array.isArray(expected)) {  
    // 🚩 2. toEqual에서 Diff 모듈 직전에 값 조절해주면 됨.  
    return toEqual.call(this, 'toHaveText', locator, 'Locator', async (isNot, timeout) => {  
      // ... codes  
    }) else {  
    // 🚩 3. 배열이 아닌 경우  
    // 이걸 회귀 테스트로 문제 없음을 확인했음  
    return toMatchText.call(this, 'toHaveText', locator, 'Locator', async (isNot, timeout) => {  
      // ... codes  
    })  
  }  
}
```

단일 정규식은 문제 X (회귀 테스트로 확인).

배열 내부의 정규식. toEqual에 문제가 있음을 확인

배웠던 건 써먹어요(부재: 어려워 보여도 사실 쉬워요)

문제 해결

printDiff(외부 모듈)

넘겨주기 전, 값 조정 진행.

match되면 그냥 동일한 값
으로 갈아끼워줌.

이후 외부 모듈 전달

```
60     if (pass) {
61         printedExpected = `Expected: not
        ${this.utils.printExpected(expected)}`;
62         printedReceived = `Received: ${this.utils.printReceived(received)}`;
63 +     } else if (Array.isArray(expected) && Array.isArray(received)) {
64 +         const normalizedExpected = expected.map((exp, index) => {
65 +             const rec = received[index];
66 +             if (isRegExp(exp))
67 +                 return exp.test(rec) ? rec : exp;
68 +             return exp;
69 +         });
70 +         printedDiff = this.utils.printDiffOrStringify(
71 +             normalizedExpected,
72 +             received,
73 +             EXPECTED_LABEL,
74 +             RECEIVED_LABEL,
75 +             false,
76 +         );
77 +     } else {
78         printedDiff = this.utils.printDiffOrStringify(
79             expected,
```

문제 해결

✓ yury-s approved these changes on Nov 12, 2024View reviewed changes

packages/playwright/src/matchers/toEqual.ts Outdated Hide resolved

```
62 + } else if (Array.isArray(expected) && Array.isArray(received)) {
63 +   const normalizedExpected = expected.map((exp, index) => {
64 +     const rec = received[index];
65 +     if (exp instanceof RegExp)
```

yury-s on Nov 12, 2024Member

nit: 정규 표현식 판단엔 instanceof 말고 playwright-core/lib/utils 에 isRegExp 쓰면 좋습니다.



pengooseDev on Nov 12, 2024

적용 했습니다

1

yury-s merged commit a501232 into microsoft:main on Nov 12, 202428 of 29 checks passedView detailsRevert



Pull request successfully merged and closed

You're all set — the pengooseDev:fix-toHaveText-reg... branch can be safely deleted. If you wish, you can also delete this fork of microsoft/playwright in the [settings](#).

Delete branch

알면 좋긴 함 ㅋㅋ



내부 로직을
전부 알 필요는 없어요
(기존 테스트 믿기)

+ 커밋 히스토리로 이전 구현에서 힌트 얻기

[Issue]

[PR](vitest PR은 잘 안열림)

내부 로직을 전부 알 필요는 없어요 (테스트를 믿기)

[#30662]: test.fail.only 메서드 지원 #30662

[Feature]: support test.fail.only #30662

New


Closed

#33001



vitalets opened on May 4, 2024



 Feature Request

test.fail.only() 만들어 줘.

Example

1번째 실패 테스트 짤 때, 저것만 돌리고 싶음.

```
test.fail.only('my test', async ({ page }) => {  
  throw new Error('foo');  
});
```

```
test('my test 2', async ({ page }) => {  
  // pass  
});
```



Assignees

No one assigned

Labels

feature-test-runner

Type

No type

Projects

No projects

Milestone

No milestone

기존 fail, only 테스트를 읽고 필요한 fail.only 추가

```
282 + test('should work with fail.only inside describe', () => {
runInlineTest }) => {
283 +   const result = await runInlineTest({
284 +     'a.test.ts': `
285 +       import { test, expect } from '@playwright/test';
286 +
287 +       test.describe.only("suite", () => {
288 +         test.skip('focused skip by suite', () => {});
289 +         test.fixme('focused fixme by suite', () => {});
290 +         test.fail.only('focused fail by suite', () => {
expect(1).toBe(2); });
291 +       });
292 +
293 +       test.describe.skip('not focused', () => {
294 +         test('no marker', () => {});
295 +       });
296 +     },
297 +   });
298 +   const expectTest = expectTestHelper(result);
299 +
300 +   expect(result.exitCode).toBe(0);
301 +   expect(result.passed).toBe(1);
302 +   expect(result.failed).toBe(0);
303 +   expect(result.skipped).toBe(0);
```

```
33 + test.skip('my test', async () => {});
34 test.fixme('my test', async () => {});
35 test.fail('my test', async () => {});
36 + test.fail.only('my test', async () => {});
37 });
38 153 + });
39 154 });
40 155
41 156 + test.fail.only('focused fail.only test', () => {
42 157 +   expect(1 + 1).toBe(3);
43 158 + });
44 159 +
45 160 test.describe('non-focused describe', () => {
46 161   test('describe test', () => {
47 162     expect(1 + 1).toBe(3);
48 163   });
49 164 });
50 165
51 176 test.only('test4', () => {
52 177   expect(1 + 1).toBe(2);
53 178 });
54 179 + test.fail.only('test5', () => {
55 180 +   expect(1 + 1).toBe(3);
56 181 + });
57 182 });
```

기존 컨벤션 따라가기

내부 로직을 전부 알 필요는 없어요 (테스트를 믿기)

로직 파악 및 신규 플래그 추가 / 타입 수정

```
85 + private _createTest(type: 'default' | 'only' | 'skip' |  
    'fail' | 'fail.only', location: Location, title: string, f  
Function | TestDetails, fn?: Function) {  
86     throwIfRunningInsideJest();  
87     const suite = this._currentSuite(location, 'test()');  
88     if (!suite)  
  
105     test._tags.push(...validatedDetails.tags);  
106     suite._addTest(test);  
107  
108 +     if (type === 'only' || type === 'fail.only')  
109         test._only = true;  
110     if (type === 'skip' || type === 'fixme' || type === 'f  
111         test._staticAnnotations.push({ type });  
112 +     else if (type === 'fail.only')  
113 +         test._staticAnnotations.push({ type: 'fail' });
```

필자는 test._staticAnnotations와 같은 필드의 특성과 동작 방식을 전부 이해한 뒤 구현. (꾸준히 기여할 생각이 있기 때문)

다만, 이것이 어렵게 느껴진다면, 네이밍으로 기능을 이해하고, 구현하는 한 뒤, 회귀 테스트와 기존의 테스트에서 확인하는 Flow도 가능.

Commit History에서 힌트 얻기

```
1145 +
1146 + To declare a focused "failing" test:
1147 + * `test.fail.only(title, body)`
1148 + * `test.fail.only(title, details, body)`
1149 +
1150 + **Usage**
1151 +
1152 + You can declare a focused failing test, so that Playwright runs only
1153 + this test and ensures it actually fails.
1154 + ```js
1155 + import { test, expect } from '@playwright/test';
1156 +
1157 + test.fail.only('focused failing test', async ({ page }) => {
1158 +   // This test is expected to fail
1159 + });
1160 + test('not in the focused group', async ({ page })
1161 +   // This test will not run
1162 + });
1163 + ```
1164 +
1165 + ### param: Test.fail.only.title
1166 + * since: v1.49
1167 +
1168 + - `title` ?<[string]>
1169 +
1170 + Test title.
1171 +
1172 + ### param: Test.fail.only.details
```

```
113 + fail: {
114 +   (title: string, body: (args: TestArgs & WorkerArgs, testInfo:
115 +     TestInfo) => Promise<void> | void): void;
116 +   (title: string, details: TestDetails, body: (args: TestArgs &
117 +     WorkerArgs, testInfo: TestInfo) => Promise<void> | void);
118 +   (condition: boolean, description?: string): void;
119 +   (callback: (args: TestArgs & WorkerArgs) => boolean, description?:
120 +     string): void;
121 +   (): void;
122 +   only: TestFunction<TestArgs & WorkerArgs>;
123 + }
```

타입 파일이 빌드 시점에 작성한 Spec 문서를 기반으로 동적으로 생성되는 방식이라 처음에 이해가 어려웠음.

이러한 경우, 해당 파일의 Commit History에서 **“현재 구현하고자 하는 Feature”와 비슷한 Commit**을 뜯어보면 쉽게 힌트를 얻을 수 있음.

내부 로직을 전부 알 필요는 없어요 (테스트를 믿기)

feat(testType): add support for test.fail.only method #33001

Edit <> Code

Merged dgozman merged 21 commits into microsoft:main from pengooseDev:feature-fail-only on Oct 16, 2024

Conversation 33 Commits 21 Checks 29 Files changed 7

+278 -17

pengooseDev commented on Oct 8, 2024 • edited by dgozman

Contributor

Reviewers

dgozman

pavelfeldman

Description

Fixes [issue #30662](#)

This PR adds support for `test.fail.only` in Playwright Test, fail.

- Implemented the `test.fail.only` method.
- Added regression tests for `test.fail.only` to ensure
- Updated documentation and TypeScript definitions (1)

As this is my first contribution, I would appreciate feedback & changes needed, I'm happy to make them. :)

👍 4 ❤️ 6 🚀 3

dgozman approved these changes on Oct 16, 2024

View reviewed changes

dgozman left a comment

Contributor

This looks good, I am going to merge it. I have an idea on how to improve generated types a bit, I will follow up with that. Thank you for the PR!

😊 😄 1

dgozman merged commit d10a5e5 into microsoft:main on Oct 16, 2024
29 checks passed

View details

Revert

dgozman mentioned this pull request on Oct 16, 2024

fix: correct types for things like `test.describe.only` #33142

Merged

**오픈소스는 사람이 관리한다.
코드 리뷰 또한 리소스!**



Github: [pengooseDev](#)
Solved: [pengoosedev](#)