



Products

Can't use locally created vector because "borrowed" [duplicate]

[Log in](#) [Sign up](#)

[Ask Question](#)

Asked 2 years, 3 months ago

Active 2 years, 3 months ago

Viewed 93 times



0



This question already has an answer here:

[What does "cannot move out of index of" mean?](#) (1 answer)

Closed 2 years ago.

I can't get first element of the vector because of the error and can't change struct design either. I tried borrowing but struct expects a `ExtrudeGeometry`.

```
#[wasm_bindgen]
pub fn toCollection(arr: js_sys::Array, r_type: String) -> JsValue {
    let n_arr: Vec<ExtrudeGeometry> = arr.into_serde().unwrap();
    if r_type == "GeometryCollection" {
        return JsValue::from_serde(&OutputGeometryCollection {
            collection: n_arr,
            r_type: r_type,
        })
        .unwrap();
    } else {
        let ex: ExtrudeGeometry = n_arr[0];
        return JsValue::from_serde(&OutputObject {
            data: ex,
            r_type: r_type,
        })
        .unwrap();
    }
}
```

```
error[E0507]: cannot move out of borrowed content
  --> src/lib.rs:308:39
   |
308 |         let ex: ExtrudeGeometry = n_arr[0];
   |                                   ^^^^^^^^^
   |                                   |
   |                                   cannot move out of borrowed content
   |                                   help: consider borrowing here: `&n_arr[0]`
```

[json](#) [vector](#) [rust](#) [serde](#) [borrowing](#)

[Share](#)

[Improve this](#)

[question](#)

[Follow](#)

edited Sep 3 '19 at 7:50



[Boiethios](#)

28.9k • 10 • 104 • 147

asked Sep 3 '19 at 7:28



[mertsclimb](#)

157 • 7

2

Hi there! Does [this](#) or [this](#) answer your question? Summary: you are trying to get an element from a `Vec` by value, but that is not possible without cloning or removing it entirely from the `Vec`.

– [Lukas Kalbertodt](#)

Sep 3 '19 at 7:34

I am new to rust. I don't understand the error. If cloning helps it will be my answer.

– [mertsclimb](#)

Sep 3 '19 at 7:37

1

If you throw the vector away, you don't need to clone the item. Just consume it: `n_arr.into_iter().next().unwrap()`

– [Boiethios](#)

Sep 3 '19 at 7:37

1

I just now also found [this Q&A](#) which probably fits your question more closely!

Your privacy [Lukas Kalbertodt](#)

By clicking "Accept all cookies", you agree Stack Exchange can store cookies on your device and disclose information in accordance with our [Cookie Policy](#).

1

Accept all cookies

[Customize settings](#)

Don't worry! Your question now serves as signpost and will be easier found by people who formulate the problem like you do.

– [Lukas Kalbertodt](#)

Sep 3 '19 at 7:57

Show 2 more comments

1 Answer

Active Oldest Votes



2



I assume in this answer that the ownership system of Rust is known. Your vector owns the items, so if you ask for the first one, you can only borrow it because the vector is composed of items contiguous in memory. You cannot randomly remove an item from it with the index notation.

If you want to take the first one, you have 3 choices:

- You don't care about the remaining of the vector: you can transform it into an iterator and take the first item iterated:

```
vector
    .into_iter() // consume the vector to get an iterator
    .next() // get the first iterated item
    .unwrap()
```

- You care about the remaining, but you don't care about the ordering: use [swap_remove](#) :

```
vector.swap_remove(0)
```

- You care about the remaining and the ordering: don't use a vector. If you don't have that choice, you can use [remove](#) , but that's a $O(n)$ function.

By the way, the return in last position is not idiomatic:

```
#[wasm_bindgen]
pub fn toCollection(arr: js_sys::Array, r_type: String) -> JsValue {
    let n_arr: Vec<ExtrudeGeometry> = arr.into_serde().unwrap();

    if r_type == "GeometryCollection" {
        JsValue::from_serde(&OutputGeometryCollection {
            collection: n_arr,
            r#type: r_type,
        })
        .unwrap()
    } else {
        let ex = n_arr.into_iter().next().unwrap();

        JsValue::from_serde(&OutputObject {
            data: ex,
            r#type: r_type,
        })
        .unwrap()
    }
}
```

Share

Improve this answer

Follow

edited Sep 3 '19 at 11:57

answered Sep 3 '19 at 7:47



Boithios

28.9k • 10 • 104 • 147

Add a comment

Not the answer you're looking for? Browse other questions tagged [json](#) [vector](#) [rust](#) [serde](#) [borrowing](#) or ask your own question.

The Overflow Blog

- Sequencing your DNA with a USB dongle and open source code
- Don't push that button: Exploring the software that flies SpaceX rockets and...

Featured on Meta

- Providing a JavaScript API for userscripts
- Congratulations to the 59 sites that just left Beta

Linked

What does "cannot move out of index of" mean?

How can I take an item from a Vec in Rust?

[Getting value from a collection without using the Clone trait](#)

Related

[403](#)

[Why can't I make a vector of references?](#)

[Cannot move out of borrowed content / cannot move out of behind a shared reference](#)

[Cannot move out of borrowed content when borrowing a generic type](#)

[How to implement iterator for boxed values?](#)

[How to clone last element from vector?](#)

[Pass self reference to contained object's function](#)






[How to capture self consuming variable in a struct?](#)

[Reference to unwrapped property fails: use of partially moved value: `self`](#)

[1](#)

[Rust cannot find module even though the file is present](#)

Hot Network Questions

-  [What's the social meaning of "He was a student of..."?](#)
 -  [Naruto fighting game with Hulk and Homer Simpson?](#)
 -  [On what basis do countries repay international loans?](#)
 -  [Why couldn't Smith absorb Neo in The Matrix Reloaded?](#)
 -  [Circuit analysis homework - LTspice results are different from calculations](#)
- [more hot questions](#)

STACK OVERFLOW

[Questions](#)
[Jobs](#)
[Developer Jobs Directory](#)
[Salary Calculator](#)
[Help](#)
[Mobile](#)

PRODUCTS

[Teams](#)
[Talent](#)
[Advertising](#)
[Enterprise](#)

COMPANY

[About](#)
[Press](#)
[Work Here](#)
[Legal](#)
[Privacy Policy](#)
[Terms of Service](#)
[Contact Us](#)
[Cookie Settings](#)
[Cookie Policy](#)

STACK EXCHANGE NETWORK

[Technology](#)
[Culture & recreation](#)
[Life & arts](#)
[Science](#)
[Professional](#)
[Business](#)
[API](#)
[Data](#)

[Blog](#)
[Facebook](#)
[Twitter](#)
[LinkedIn](#)
[Instagram](#)

site design / logo © 2021 Stack Exchange Inc; user contributions licensed under [cc by-sa](#). rev 2021.12.22.41046