Stack Overflow

1. About
2. Products
3. For Teams

1. Stack Internal Implement a knowledge platform layer to power your enterprise and AI tools.
2. Stack Data Licensing Get access to top-class technical expertise with trusted & attributed content.
3. Stack Ads Connect your brand to the world's most trusted technologist communities.
4. Releases Keep up-to-date on features we add to Stack Overflow and Stack Internal.
5. About the company Visit the blog

Loading…

current community

Stack Overflow

help chat

- Meta Stack Overflow

your communities

Sign up or log in to customize your list.

more stack exchange communities

company blog

3. Log in
4. Sign up

AI Assist is now on Stack Overflow. Start a chat to get instant answers from across the network. Sign up to save and share your chats.

Home
Questions
AI Assist
Tags
Challenges
7. Chat
Articles
Users
Companies
Collectives

Communities for your favorite technologies. Explore all Collectives

Stack Internal

Stack Overflow for Teams is now called **Stack Internal**. Bring the best of human thought and AI automation together at your work.

Try for free Learn more
Stack Internal

Bring the best of human thought and AI automation together at your work. Learn more

Collectives™ on Stack Overflow

Find centralized, trusted content and collaborate around the technologies you use most.

Learn more about Collectives

Stack Internal

Knowledge at work

Bring the best of human thought and AI automation together at your work.

Explore Stack Internal

Unable to add two f64 numbers together

Asked 10 years, 6 months ago

Modified [10 years, 6 months ago](#)

Viewed 790 times

0

I have two `f64` numbers I wish to add together, however the following code produces an error. How can I go about adding these two floats together?

Code:

```
total_duration = 0.925338 + 0.741495;
```

Error:

```
src/main.rs:56:26: 56:57 error: the trait `core::ops::Add<_>` is not implemented for the type `f64` [E0277]
src/main.rs:56          total_duration = r.elapsed_time + total_duration;
```

- [rust](#)
- [addition](#)

Share

Improve this question

Follow

[edited Jun 1, 2015 at 17:49](#)

[Shepmaster](#)

439k116116 gold badges1.3k1.3k silver badges1.5k1.5k bronze badges

asked Jun 1, 2015 at 17:24

[Jacob Clark](#)

3,46755 gold badges3535 silver badges6262 bronze badges

5

> 2
>
> Your code works [in the playpen](#).
>
> Shepmaster – [Shepmaster](#)
>
> 2015-06-01 17:49:59 +00:00
>
> Commented Jun 1, 2015 at 17:49

> 4
>
> Your error message indicates a completely different line than the one you've indicated is causing the problem. [stackoverflow.com/help/mcve](#)
>
> Benjamin Lindley – [Benjamin Lindley](#)
>
> 2015-06-01 19:28:24 +00:00
>
> Commented Jun 1, 2015 at 19:28

> @BenjaminLindley, there is one line of code, are you OK? It could ONLY be that line.
>
> Jacob Clark – [Jacob Clark](#)
>
> 2015-06-02 08:00:02 +00:00
>
> Commented Jun 2, 2015 at 8:00

> 3
>
> @JacobClark it seems there are at least 56 lines of code, and the one in the error message doesn't **at all** look like the one you claim to cause the problem. [stackoverflow.com/help/mcve](#)
>
> Micha■ Politowski – [Micha■ Politowski](#)
>
> 2015-06-02 09:06:33 +00:00
>
> Commented Jun 2, 2015 at 9:06

> 1
>
> If that's your only line, then the rust compiler has a pretty serious bug. You should file a bug report. It should be complaining about expecting an item, since an assignment statement cannot appear outside of a function. The error you should be getting is something like this: `src/main.rs:1:1: 1:15 error: expected item, found 'total_duration' src/main.rs:1 total_duration = 0.925338 + 0.741495;`
>
> Benjamin Lindley – [Benjamin Lindley](#)
>
> 2015-06-02 18:35:55 +00:00
>
> Commented Jun 2, 2015 at 18:35

Add a comment |

1 Answer

Sorted by: [Reset to default](#)

Highest score (default) Trending (recent votes count more) Date modified (newest first) Date created (oldest first)

It's probably because you mix `f64` with `f32`. You may have to specify the type when you do this kind of additions:

```
let total_duration = 0.925338f64 + 0.741495f64;
```

You don't have to do this when you add two variables that already are type f64:

```
let x: f64 = 0.925338;
let y: f64 = 0.741495;
let total_duration = x + y;
```

I'm not getting an error though when using your code, is your compiler up to date?

Share

Improve this answer

Follow

edited Jun 1, 2015 at 17:34

answered Jun 1, 2015 at 17:29

Mathieu David

5,34833 gold badges2020 silver badges2828 bronze badges

Sign up to request clarification or add additional context in comments.

Comments

Add a comment

Your Answer

Thanks for contributing an answer to Stack Overflow!

- Please be sure to *answer the question*. Provide details and share your research!

But *avoid* …

- Asking for help, clarification, or responding to other answers.
- Making statements based on opinion; back them up with references or personal experience.

To learn more, see our tips on writing great answers.

Sign up or **log in**

Sign up using Google

Sign up using Email and Password

Submit

Post as a guest

Name

Email

Required, but never shown

Start asking to get answers

Find the answer to your question by asking.

Ask question

Explore related questions

- rust
- addition

See similar questions with these tags.

- The Overflow Blog
  Introducing Stack Overflow AI Assist—a tool for the modern developer
  Treating your agents like microservices
- Featured on Meta
  Chat room owners can now establish room guidelines
  AI Assist is now available on Stack Overflow
  Policy: Generative AI (e.g., ChatGPT) is banned

Related

7
Why does the binary + operator not work with two &mut int?
0
Adding two numbers without cloning both
33
Why do I get an error when adding an integer to a floating point?
11
How can I sum a range of numbers in Rust?
9
How do I add two Rust arrays element-wise?
16
How do I add a signed integer to an unsigned integer in Rust?
2
Addition with Rust and Webassembly
0
Rust how to add two referenced i32's together?
0
Rust add two numbers from stdin
4
How to sum elements of Vec<Vec<f64>> together into a Vec<f64>?

Hot Network Questions

- Would it be plausible to have a flat eye with a moving pupil?
- "Save as webpage, HTML Only" Fail while offline, Firefox
- Why Select Within Distance set to 1m selects every waterbody?
- Diagonal Arrows Under an Equation
- Vertical alignment of equations inside tabularx

- [How do we define time-ordering operations in QFT?](#)
- [How to check if two tables are identical?](#)
- [Where in the world is the chicken?](#)
- [Trying to understand UK tax brackets for salary above 150k](#)
- [What do the terms postponement, slip and scrub mean in the context of the space shuttle program?](#)
- [Calculating the Resistance in an Induction Heater](#)
- [Schottky OR-ing diodes passing reverse voltage; is it a faulty diode?](#)
- [Finding real variable where a 2×2 complex matrix is singular in Mathematica](#)
- [variable name change causes segfault](#)
- [When should I use tied notes vs. the whole note duration?](#)
- ["Imagine if..." vs. "Imagine..."](#)
- [Is there any way to restore hearts without consuming anything, without hot springs, and without shrines?](#)
- [Curve has weird bumps when triangulated](#)
- [Is it possible to use LaTeX to create RPG-related PDFs, such as character sheets or adventure modules?](#)
- [Looking for a proper formal substitute for "quick fix" for formal letter](#)
- [What does Felix mean by telling Bond to pick a contact point while "standing up" in Diamonds Are Forever?](#)
- [Explicit examples separating Wasserstein distances from Jensen–Shannon divergence?](#)
- [How to reduce GAM sensitivity to sparse data](#)
- [Is my homebrew Species "Attuned" balanced for D&D 2024?](#)

[more hot questions](#)

[Question feed](#)

Subscribe to RSS

Question feed

To subscribe to this RSS feed, copy and paste this URL into your RSS reader.

[Stack Overflow](#)

- [Questions](#)
- [Help](#)
- [Chat](#)

[Business](#)

- [Stack Internal](#)
- [Stack Data Licensing](#)
- [Stack Ads](#)

[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](#)

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