

About practicing and learning by contributing to Pharo

stephane.ducasse@inria.fr

<https://github.com/pharo-project/pharo>





z. zidane

**Practiced
football?**



z. zidane

Practiced
football? Yes



z. zidane



r. federer

**Practiced
tennis?**



r. federer

Practiced
tennis? Yes



r. federer



k. buniatishvili

Practiced Piano?



k. buniatishvili

Practiced
Piano? Yes



k. buniatishvili

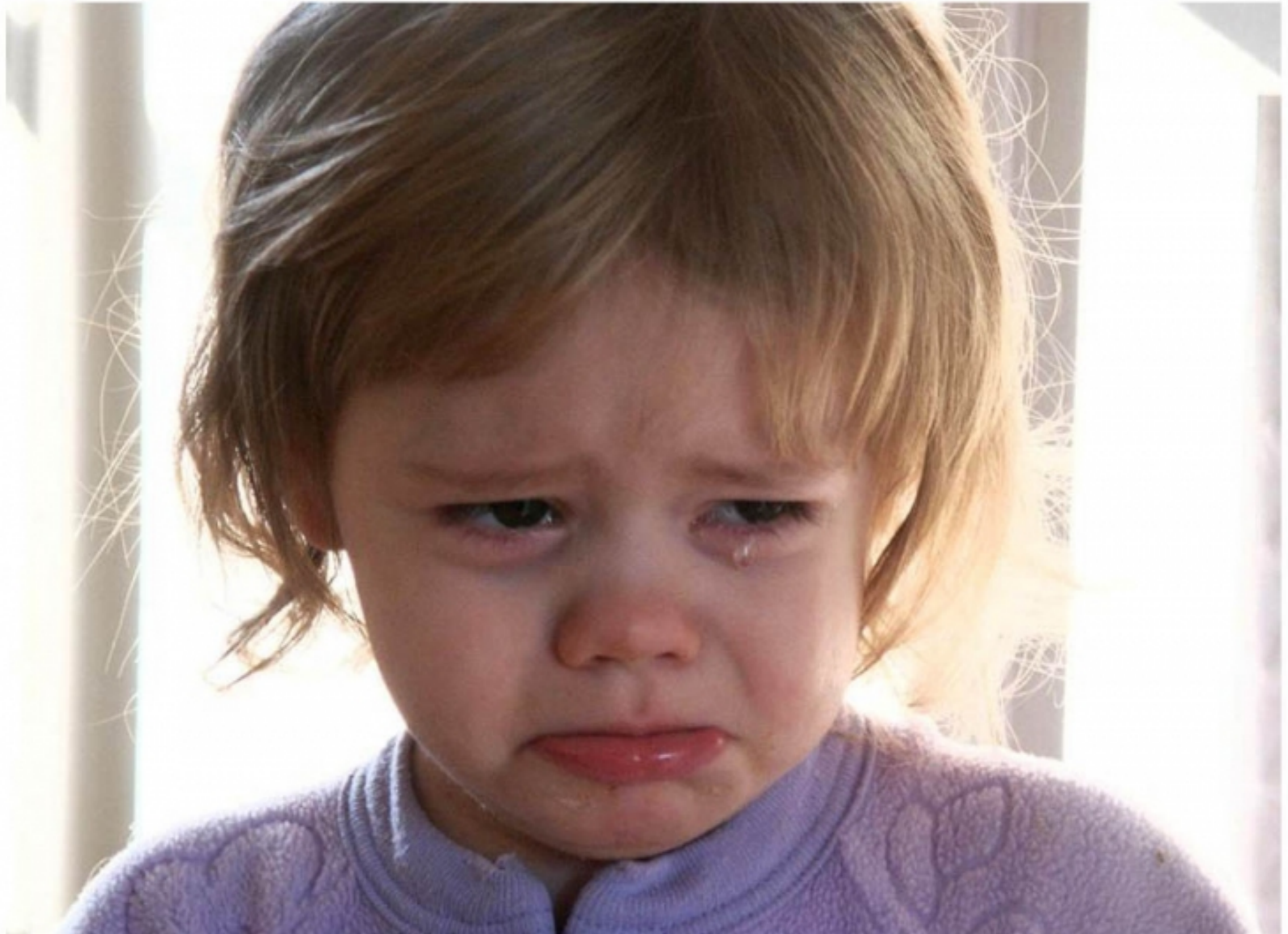
Did you see the
pattern?

How to **practice**
software skills?

How to practice software skills?

- Program (but how to get feedback) code reviews are precious but difficult to get
- Code katas (always fun)
- Participate to sprints - Pair program
- **Join the community...**

But I'm a noob





**And the world is
complex**

Wait wait wait...

We are all noobs!



**The solution is just
some clicks away...**

Take **any** opportunity
to read code

Take **any** opportunity
to contribute

A PIECE OF CAKE



**Contributing is easy &
simple**

Possible contributions

- **Read/answer newbie questions**
- **Participate to the wiki**
- **Review bug entry description**
- **Review bug fixes**
- **Write tests for bug entries**
- **Improve test coverage**
- **Improve comments**

Newbie Questions

- I learned a lot answering them :)
- When I did not know I looked for the answers
- Try and learn

Read discord

Pharo

exercism building out a portfolio for ht...

Search

FOR NEW USERS

- # welcome
- # mooc
- # tutorial-help
- # video-tutorials
- # exercism

PHARO

- # news
- # general
- # development
- # offtopic
- # iceberg
- # bootstrap
- # vm
- # research
- # spec

LIBRARIES

- # databases
- # seaside

@tmackinnon

roelof Today at 11:14 AM
Oke, I see the symmertics but do not see the way I can calculate the spaces

roelof Today at 1:21 PM
still not right:"

```
makeRow: aChar  
  | half outspaces innerspaces |  
  outspaces := aChar asInteger -  
  $A asInteger + 1.  
  innerspaces := aChar asInteger -  
  $A asInteger - outspaces + 1.  
  half := String  
    streamContents: [ :stream |  
      stream  
        nextPutAll: (self  
displaySpaces: outspaces);  
        nextPut: aChar;  
        nextPutAll: (self  
displaySpaces: innerspaces) ].  
  ^ half , half allButLast reversed
```

ADMIN—5

- Alexandre Bergel
- astares
- BenComan
- Esteban Lorenzано
- Serge Stinckwich

ONLINE—97

- [MT]fire
Playing Garry's Mod
- Aliaksei Syrel
- Allex Oliveira
- Atsa
- Bence.Fabian
- Benoit Verhaeghe
- benravin
- Bill Dargel
- hitmanner

NEW MESSAGES

BenComan Today at 1:58 PM
Can you show what you do get?

Answer questions

- On discord
- Mailing-list

The screenshot shows a Discord chat window for the 'Pharo' server. The channel is '# general'. The chat history shows a message from Kasper asking about ORM frameworks for Smalltalk, followed by Julien's response about a DSL prototype, and Kasper's reply. The interface includes a sidebar with channel categories like 'FOR NEW USERS', 'PHARO', and 'LIBRARIES'. A user list on the right shows 'ADMIN-5' and 'ONLINE-100' members.

FOR NEW USERS

- # welcome
- # mooc
- # tutorial-help
- # video-tutorials
- # exercism

PHARO

- # news
- # general
- # development
- # offtopic
- # iceberg
- # bootstrap
- # vm
- # research
- # spec

LIBRARIES

- # databases
- # seaside

ADMIN-5

- Alexandre Bergel
- astares
- BenComan
- Esteban Lorenzani
- Serge Stinckwich

ONLINE-100

- [MT]fire (Playing Garry's Mod)
- Aliaksei Syrel
- Allex Oliveira
- Atsa
- Bence.Fabian
- Benoit Verhaeghe
- benravin
- Bill Dargel
- bitmapper
- BlazingMagpie

Message #general

Kasper Last Monday at 11:50 AM
Do any of the object relational mapping frameworks for smalltalk do something similar to .Net linq? I used the AST to get a super preliminary thing running:

(PliqTestAuthor new select: [:auth | auth born ~= 1960]) sqlCode.
gives this result:
select *
from Author as auth
where auth.born != 1960

Most things are not yet possible, but just wanted to know if it had been done before. (edited)

Julien Last Monday at 1:27 PM
@Kasper I have a prototype of DSL to generate SQL code from Pharo but it is not public yet I can put it on a github repository
we should definitely talk @ ESUG :p

Kasper Last Monday at 2:39 PM
@Julien Ohh - sounds like fun. See you next week sometime.

DenisKudriashov Last Monday at 8:56 PM
@Kasper there is an ORM GLOPP which supports

Bug issue contribution

- Can you understand it?
 - else ask for precision
- Can you repeat the problem mentioned?
 - Report it! This is a valuable information

Bug issue contribution

- Write a test to show the problem?
- If the contributor provides a test, is it sound?

Join/read bug issue discussion



dalehenrich commented 19 days ago

Author + 😊 ...

Once you get past the missing tab message you aren't completely out of the woods. If you want the STON output to be correctly display on stdout (on Ubuntu 16.04) you have to arrange for STONWriter to use `String lf`, instead of the default `String cr` --- if you don't, the output is pretty much scrambled ...



dalehenrich referenced this issue 19 days ago

(Pharo 7.0.3) StdioStream>>peek (on stdin) returns a byte instead of a character #4173

Closed



dalehenrich commented 19 days ago

Author + 😊 ...

FWIW, this implementation satisfies my needs:

```
tab
  ^ self nextPutAll: String tab
```



akgrant43 commented 18 days ago

Collaborator + 😊 ...

As with #4173, take a look at wrapping stdout with a `ZnCharacterWriteStream` and `ZnNewLineWriterStream`.



dalehenrich commented 18 days ago

Author + 😊 ...

@akgrant43 thanks a lot

None yet

Milestone

No milestone

Notifications [Customize](#)

Unsubscribe

You're receiving notifications because you're watching this repository.

2 participants



Lock conversation

Pin issue

Transfer issue

A person with short dark hair, wearing a white button-down shirt, is shown from the chest up. They are looking slightly to the right with a thoughtful expression, their right hand is raised towards their face. The background is a plain, light-colored wall.

What if you do not know?

**What if you do not
understand?**

Easy!

**Drop it and pick the
next one**

Read tests to learn

- Tests are cool to learn
- Tests can be improved, so you can improve them

Improve test coverage

- It is often easier to write tests than the code :)
- Grab an object and inspect it
- Turn this in a test

Add executable comments

```
allRangesOfSubstring: aSubstring
  "'Ab cd ef Ab cd' allRangesOfSubstring: 'cd' >>> {(4 to: 5). (13 to: 14)}"
  "'Ab cd ef Ab cd' allRangesOfSubstring: 'zz' >>> #()"

^ Array streamContents: [:s | | start subSize |
  start := 1.
  subSize := aSubstring size.
  [start isZero]
  whileFalse: [ start := self findString: aSubstring startingAt: start.
    start > 0
    ifTrue: [s nextPut: (start to: start + subSize - 1).
      start := start + subSize]]]
```

[Free side effect]



When you contribute to open-source you improve **your** visibility

**Do not miss a great
opportunity to grow your
knowledge**

Join and invent the future

