Ask Question

Constructing a list with ID and a number of values [closed]

Asked today Modified today Viewed 28 times

-1

Closed. This question needs to be more focused. It is not currently accepting answers.

Want to improve this question? Update the question so it focuses on one problem only by editing this post.

Closed 3 hours ago.

Edit question

I am looking at constructing a list with ID and a number of values for use with `tabulated-list-print`.

But the following only has a single associated with each `ID`.

```
(setq id-values-list (list (cons ID1 va
                           (cons ID2 va
                           (cons ID3 va
```

Then continued with the following because I think it requires an associated list.

```
(setq id-values-list `(list (ID1 . ,(lis
                            (ID2 . ,(lis
                            (ID3 . ,(lis
```

But this prints as

```
List: ((ID1 "value1a" "value1b") (ID2 "
```

Rather than what I had expected

```
List: ((ID1 . ("value1a" "value1b")) (I
```

Is the last what is expected for `tabulated-list-print` ?

`(a . b)` represents a cons cell, which is a pair of two elements, while `(a b c)` represents a list, which is a sequence of elements enclosed within parentheses. In the case of associated lists with an ID and values I am getting very confused.

What does `tabulated-list-print` exactly require as input where there are multiple values? I would like to see an example.

If a list, each element has the form `(ID [DESC1 ... DESCN])` which indicates one uses a vector as well.

list conses

Share Edit Follow Flag

edited 6 hours ago

asked 10 hours ago
Pampaula
17 | 4

3

You're writing Emacs Lisp code, but lack the most fundamental understanding of Lisp. Asking the community for help when you encounter problems is not an issue. What I take issue with is that you **clearly know (or can easily guess) solving this problem only requires basic knowledge**, but you don't intend to look through the documentation and instead ask others to answer for you (**just because you never take a look to any documentation**). So please go learn some basic Lisp knowledge (from *An Introduction to Programming in Emacs Lisp*) first. This can save other people's time. Thank you. 😊 – shynur 8 hours ago Delete

I am getting confused about the `.` part used by alists. There is not much information for having multiple values. Asking to read a book and figure it out is not the way to mark a question as solved. – Pampaula 6 hours ago

Please consider reading the Elisp manual topics about alists (before reading about alists). In particular, learn about conses. That last link answers the implicit question in your last paragraph. – Drew 6 hours ago

Looks like that for single association a `dot` is used but for multiple ones it is just a sequence. `((a . 1) ("b" 2 3))`. – Pampaula 6 hours ago

At any rate a need a format suitable for `tabulated-list-print` and it is hard to made progress. – Pampaula 6 hours ago

Type `'(ID1 . ("value1a" "value1b"))` into your `*scratchs*` buffer and then evaluate it with `C-j`. Does that answer your question? Learning the fundamentals (including some experimentation with box-and-arrow diagrams) would help cut down your questions by 90%. – NickD 3 hours ago

I have (at least temporarily) closed this question, as it includes at least two questions, and it is unclear what would be most valuable way for our users to spend their time on. – Trevoke 3 hours ago

Add a comment

Browse other questions tagged list conses or ask your own question.

The Overflow Blog

Hype or not? AI's benefits for developers explored in the 2023 Developer Survey

Pair programming? We peek under the hood of Duet, Google's coding assistant....

Featured on Meta

Statement from SO: June 5, 2023 Moderator Action

Stack Exchange Network Outage – June 15, 2023

Related

1 How to continue giving list number after insert a block of code

0 Mapping a function over `frame-list` to get a list of the `top` frame-parameter values

1 How to create and traverse a multidimensional list

1 How to check if a list ends with another list?

1 Next greater or lesser number than ARG in a list of integers

0 Using `let` with `list`

1 Clear list when number of elements exceeds N

0 Working with a list of alists

0 Updating values in a list

Hot Network Questions

Differential Forms and Gauge Invariance?

Can Cutting Words be used while in an area of Silence?

Find all animals, to get the 13th animal

Did the grandson of John Rabe received medicine from China during the start of COVID?

What happens to matter at extreme temperatures?

Is my variable a discrete random variable, ordinal, or categorical?

Last odd digit of power of 2

uniq filter adjecent lines in pipeline

What is the difference between "the number of" and "the numbers of"?

While working at my last company, I wrote some code under the AGPL license. Am I allowed to reuse the code I wrote into my personal project?

If a moderator of an online forum edits my post to say something illegal, how could I prove I didn't write the incriminating message?

New tubeless-ready tire has unpluggable micro leaks on the tread

What is the mechanism through which mass is converted to thermal energy in the accretion disc of a black hole?

How is circuit "a" simplified to circuit "b"?

Has the "Lambek embedding" into the category of (co)product-preserving presheaves been studied very much?

Usage of "you" in scientific papers

What is the meaning of 富士山麓オウム返しC?

Difference between measure-theoretic integral and differential-geometric integral

Distance between a point and an Ellipsoid

Help me identify 90s or early 2000s japanese 2D turn based PC game with RPG elements that has mech fighting

Is allowing a player to choose the result on a d20 once per long rest more powerful than features like luck points or portents?

Why did Descartes claim that animals have no souls if they have the pituitary?

Determine the equations of the circles

What are the pros and cons of the practice that includes some questions in a math exam which are directly discussed in the lectures?

Site design / logo © 2023 Stack Exchange Inc; user contributions licensed under CC BY-SA; rev 2023.6.16.43501