

# ssym

April 4, 2023

This file is part of CasADi.

CasADi -- A symbolic framework for dynamic optimization.  
Copyright (C) 2010-2023 Joel Andersson, Joris Gillis, Moritz Diehl,  
KU Leuven. All rights reserved.  
Copyright (C) 2011-2014 Greg Horn

CasADi is free software; you can redistribute it and/or  
modify it under the terms of the GNU Lesser General Public  
License as published by the Free Software Foundation; either  
version 3 of the License, or (at your option) any later version.

CasADi is distributed in the hope that it will be useful,  
but WITHOUT ANY WARRANTY; without even the implied warranty of  
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU  
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public  
License along with CasADi; if not, write to the Free Software  
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

## 1 SX.sym

```
[1]: from casadi import *
```

## 2 Construct using a single name

The names of the entries of the SX will be derived from the name provided as argument to SX.sym.  
Without shape arguments, a 1-by-1 matrix is constructed:

```
[2]: x = SX.sym("x")  
print(type(x), x)
```

```
<class 'casadi.casadi.SX'> x
```

Create a column matrix

```
[3]: print(SX.sym("x",2,1))
```

[x\_0, x\_1]

Create a row matrix

```
[4]: print(SX.sym("x",1,2))
```

[[x\_0, x\_1]]

Create a matrix

```
[5]: print(SX.sym("x",2,3))
```

[[x\_0, x\_2, x\_4],  
 [x\_1, x\_3, x\_5]]