CasADi tutorial 1

```
#
#
#
#
#
#
#
```

This tutorial file explains the use of CasADi's Matrix<T> in a python context. Matrix<T> is a general class for sparse matrices. We inspect it with the help of Matrix<double> Let's start with the import statements to load CasADi.

```
from casadi import *
from numpy import *
```

Contructors & printing

The python name for Matrix<double> is DM

```
a = DM.zeros(3,4)
print a
```

```
[[0, 0, 0, 0],
 [0, 0, 0, 0],
 [0, 0, 0, 0]]
```

The string representation shows only the structural non-zero entries. In this case there are none. Let's make a DM with some structural non-zero entries.

```
w = DM(Sparsity(4,3,[0,2,2,3],[1,2,1]),[3,2.3,8])
print w
```

```
[[00, 00, 00],
 [3, 00, 8],
 [2.3, 00, 00],
 [00, 00, 00]]
```

Internally, the Matrix<> class uses a Compressed Column Format which containts the offset to the first nonzero on each column ...

```
print "column offsets: ", w.colind()
```

```
column offsets: [0, 2, 2, 3]
```

... the row for each nonzero ...

```
print "row: ", w.row()
```

```
row: [1, 2, 1]
```

... and the nonzero data entries:

```
print "nonzeros: ", w.nonzeros()
```

```
nonzeros: [3.0, 2.3, 8.0]
```

Conversion

DM can easily be converted into other data formats

```
print w.nonzeros()
```

```
[3.0, 2.3, 8.0]
```

```
print w.full()
```

```
[[ 0.   0.   0. ]
 [ 3.   0.   8. ]
 [ 2.3  0.   0. ]
 [ 0.   0.   0. ]]
```

```
print array(w)
```

```
[[ 0.   0.   0. ]
 [ 3.   0.   8. ]
 [ 2.3  0.   0. ]
 [ 0.   0.   0. ]]
```

```
print matrix(w)
```

```
[[ 0.   0.   0. ]
 [ 3.   0.   8. ]
 [ 2.3  0.   0. ]
 [ 0.   0.   0. ]]
```

```
print w.sparse()
```

```
  (1, 0)    3.0
  (2, 0)    2.3
  (1, 2)    8.0
```