Code snippet:

```
import csm_dataface, csm_notation
…
sNotation = "CCCCn1cn(C)cc1{!re=+}.{NTf2(1-)}" # [bmim][NTf2] string
oDataFace = csm_dataface.DataFace()          # singleton object
oNotation = csm_notation.Notation(oDataFace)# initialze parser object
lstErrors = oNotation.parse(sNotation)       # evaluate [bmim][NTf2]
if len(lstErrors) == 0:                      # check consistency
  sWorkNotation = oNotation.work_notation() # work notation
  sMFTotal  = oNotation.mf_total()         # molecular formula, total
  sMFCation = oNotation.mf_compnt(1)       # molecular formula, cation
  sMFAnion  = oNotation.mf_compnt(2)       # molecular formula, anion
  lstAnionAtf0 = oNotation.atf0_compnt(2) # atom fragments, anion
  lstAnionDM   = oNotation.dmat_compnt(2) # distance matrix, anion
```

Content of assigned variables:

```
sWorkNotation =
  "CCCCn1cn(C)cc1{!re=+}.FC(F)(F)S(=O)(=O)[N-]S(=O)(=O)C(F)(F)F"
sMFTotal      = "C10H15F6N3O4S2"
sMFCation     = "C8H15N2(1+)"
sMFAnion      = "C2F6NO4S2(1-)"
lstAnionAtf0  = ['F{-}', 'C{-}{-}{-}{-}', F{-}', 'F{-}',
                 'S{=}{=}{-}{-}', 'O{=}', 'O{=}', '[N-]{-}{-}',
                 'S{=}{=}{-}{-}', 'O{=}', 'O{=}', 'C{-}{-}{-}{-}',
                 'F{-}', 'F{-}', F{-}']
lstAnionDM    = [ 1, 2, 2, 2, 3, 3, 3, 4, 5, 5, 5, 6, 6, 6,
                      1, 1, 1, 2, 2, 2, 3, 4, 4, 4, 5, 5, 5,
                         2, 2, 3, 3, 3, 4, 5, 5, 5, 6, 6, 6,
                            2, 3, 3, 3, 4, 5, 5, 5, 6, 6, 6,
                               1, 1, 1, 2, 3, 3, 3, 4, 4, 4,
                                  2, 2, 3, 4, 4, 4, 5, 5, 5,
                                     2, 3, 4, 4, 4, 5, 5, 5,
                                        1, 2, 2, 2, 3, 3, 3,
                                           1, 1, 1, 2, 2, 2,
                                              2, 2, 3, 3, 3,
                                                 2, 3, 3, 3,
                                                    1, 1, 1,
                                                       2, 2,
                                                          2 ]
```