

Open the output database (.odb)

```

Import the post- >>> from odbAccess
processing       import *
library

open the output >>>myOdb =
database        openOdb(path='path\m
                yfile.odb')
```

Once you've created your odb object, you can access all the variables through the different attributes of the class. In the following cheatsheet, we only point out the way to access the results that mostly interest us such as **Field Outputs** and **History Outputs**

Accessing to Field Outputs

In the **odb object** we will access the **steps** attribute, then the **frames** attribute to get our field output.

```

>>>steps = myOdb.steps
>>>frames =
steps['my_step_name'].frames
```

List of the main attributes in a step

description

domain

frames

historyRegion

inertiaAboutCenter

inertiaAboutOrigin

loadCases

mass

massCenter

name

number

previousStepName

procedure

timePeriod

totalTime

List of the main attributes in a frame

frames.description Return description if given
ion

frames.domain TIME or FREQUENCY

frames.fieldOutputs **return all the different outputs you specified in the simulation for each node and element**

frames.frameId

frames.frameValue return the simulation time as a float

frames.incrementNumber return the increment number as an integer

Which field output do we usually encounter

NODAL OUTPUTS

U displacement

UR rotational displacement

COORD coordinates

RF Reaction force

ELEMENT OUTPUT

S

SMISES

Exemple to access the displacement along the x axis (U1), for a given node:

```

>>>myOdb =
openOdb(path='path\myodb.odb')
>>>frame1 =
myOdb.steps['Step-1'].frames[-1]
>>>t = frame1.frameValue
>>>u1 =
frame1.fieldOutput['U'].values[27].data[0]
```

For all possible outputs, see the official Abaqus documentation

Main attributes in a fieldOutput object

componantLabels

description

name

type

Main attributes in a fieldOutput object (cont)

values all the main values are stored in
values[myNode].data

What about the instance

the **Root Assembly** The odb object has an attribute rootAssembly, which gives informations about assembly such as:

elementSets

elements

instances

name

nodeSets

nodes

rigidBodies

sectionAssignment

surfaces

Access History Output results

In your simulation, you may have specify **History Outputs** for several sets. Your odbStep object has an attribute named **historyRegion** in which you will find all the sets that have history output, as historyOutput Objects. In such objets are stored all the history outputs specified in your simulation.

```

region =
myOdb.steps['stepName'].historyRegions['Node PART-1-1.15']
```

Here is a quick exemple to access to time and volume in a fluid cavity:

```

>>>myStep.historyRegions['Node
PART-1-
1.1565'].historyOutputs[CVOL].data
returns a tuple organized as:
((t1, CVOL1), (t2, CVOL2) ...)
```

