

# LOGGING

## Design:

- Log files are created on hourly basis with 24 bucket(file) approach numbering from 0-23. Each bucket number contains log of that hour.
- Log files are stored at Log folder..
- It is designed to keep at most 24 bucket/ files(1 day log) in folder.
- When file count is increased beyond 24, application deletes oldest file from folder.
- To manage hourly creation of log files we have used cron (AP)scheduler.
- On creation of next log files, we are releasing lock from previous log files.
- Default log level is 'INFO'. It is configurable via options.

## Format:

logging.Formatter() used to specify layout for log records (TimeStamp - ModuleName - LogLevel-Message).

## Handler:

logging.FileHandler() used to create handler which sends the log records (created by loggers) to the appropriate destination.

## Setlevel:

logger.setLevel() used as filter to determine which log records to output.  
User can provide loglevel through command line options. As for ex-  
python server.py -l DEBUG

Currently, in debug mode we are logging complete request response and in Info mode we are logging informational messages.

## METADATA HIERARCHICAL REPRESENTATION

- Python is converting database data to d3.Json format and sending it to client.
- Client is converting d3.Json data to hierarchical representation.
- There is back button in “showtree” page which redirects it to home page.

