



Python Logging

It is highly recommended to store complete application flow and exceptions information to a file. This process is called logging.

The main advantages of logging are:

1. We can use log files while performing debugging
2. We can provide statistics like number of requests per day etc

To implement logging, Python provides inbuilt module logging.

Logging Levels:

Depending on type of information, logging data is divided according to the following 6 levels in python

1. **CRITICAL==>50**
Represents a very serious problem that needs high attention
2. **ERROR ==>40**
Represents a serious error
3. **WARNING ==>30**
Represents a warning message, some caution needed. It is alert to the programmer.
4. **INFO==>20**
Represents a message with some important information
5. **DEBUG ==>10**
Represents a message with debugging information
6. **NOTSET==>0**
Represents that level is not set

By default while executing Python program only WARNING and higher level messages will be displayed.