Demo Program for File Handler:

```
1)  import logging
2)  class LoggerDemoConsole:
3)
4)      def testLog(self):
5)          logger = logging.getLogger('demologger')
6)          logger.setLevel(logging.INFO)
7)
8)          fileHandler = logging.FileHandler('abc.log',mode='a')
9)          fileHandler.setLevel(logging.INFO)
10)
11)         formatter = logging.Formatter('%(asctime)s - %(name)s - %(levelname)s: %(message)s',
12)                 datefmt='%m/%d/%Y %I:%M:%S %p')
13)
14)         fileHandler.setFormatter(formatter)
15)         logger.addHandler(fileHandler)
16)
17)         logger.debug('debug message')
18)         logger.info('info message')
19)         logger.warn('warn message')
20)         logger.error('error message')
21)         logger.critical('critical message')
22)
23) demo = LoggerDemoConsole()
24) demo.testLog()
```

abc.log:
07/05/2018 08:58:04 AM - demologger - INFO: info message
07/05/2018 08:58:04 AM - demologger - WARNING: warn message
07/05/2018 08:58:04 AM - demologger - ERROR: error message
07/05/2018 08:58:04 AM - demologger - CRITICAL: critical message

Logger with Configuration File:

In the above program, everything we hard coded in the python script. It is not a good programming practice. We will configure all the required things inside a configuration file and we can use this file directly in our program.

```
logging.config.fileConfig('logging.conf')
logger = logging.getLogger(LoggerDemoConf.__name__)
```

Note: The extension of the file need not be conf. We can use any extension like txt or durga etc