```python
import logging
from config import log_config


class Logging:
    def __init__(self, name):
        """
        Function is used to instantiate the custom logger
        :param name: custom name for the logger
        """
        try:
            # Creating Custom Logger
            self.logger = logging.getLogger(name)

        except Exception as e:
            raise Exception(e)

    def initialize_logger(self):
        """
        This function adds the custom formatters and handlers to the logger object
        """
        try:
            if len(self.logger.handlers) == 0:

                # Read the mode
                log_level = log_config.log_mode

                if log_level == 'ERROR':
                    self.logger.setLevel(logging.ERROR)
                elif log_level == 'DEBUG':
                    self.logger.setLevel(logging.DEBUG)

                # Creating the formatters
                formatter = logging.Formatter('%(asctime)s:%(levelname)s:%(name)s:%(message)s'
)

                # Creating Handlers
                file_handler = logging.FileHandler('Advance Image Downloader.log')

                # Adding Formatters to the Handlers
                file_handler.setFormatter(formatter)

                # Adding Handler to loggers
                self.logger.addHandler(file_handler)

            return self.logger
        except Exception as e:
            raise Exception(e)

    def print_log(self, log_statement, log_level):
        """
        This function is use for printing and logging the statements
```

```
52          :param log_statement: Statement for logging
53          :param log_level : Level of log that needs to be maintained
54          """
55          try:
56              if log_level == 'info':
57                  self.logger.info(log_statement)
58              elif log_level == 'error':
59                  self.logger.error(log_statement)
60              elif log_level == 'exception':
61                  self.logger.exception(log_statement)
62          except Exception as e:
63              raise Exception(e)
64
```