netLog

Logging network events for a session.

Process: Main

```
const { netLog } = require('electron')

app.whenReady().then(async () => {
  await netLog.startLogging('/path/to/net-log')
  // After some network events
  const path = await netLog.stopLogging()
  console.log('Net-logs written to', path)
})
```

See `--log-net-log` to log network events throughout the app's lifecycle.

Note: All methods unless specified can only be used after the `ready` event of the `app` module gets emitted.

Methods

netLog.startLogging(path[, options])

- `path` string - File path to record network logs.
- `options` Object (optional)
  - `captureMode` string (optional) - What kinds of data should be captured. By default, only metadata about requests will be captured. Setting this to `includeSensitive` will include cookies and authentication data. Setting it to `everything` will include all bytes transferred on sockets. Can be `default`, `includeSensitive` or `everything`.
  - `maxFileSize` number (optional) - When the log grows beyond this size, logging will automatically stop. Defaults to unlimited.

Returns `Promise<void>` - resolves when the net log has begun recording.

Starts recording network events to `path`.

netLog.stopLogging()

Returns `Promise<void>` - resolves when the net log has been flushed to disk.

Stops recording network events. If not called, net logging will automatically end when app quits.

Properties

netLog.currentlyLogging *Readonly*

A `boolean` property that indicates whether network logs are currently being recorded.