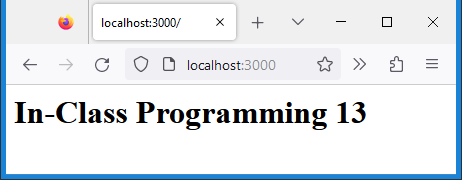
# In-Class Programming 13

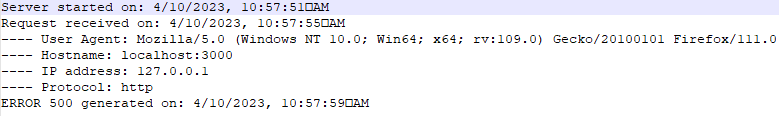
Create the following JavaScript code to create a Node Express web server with a logging facility

**Step 1:** Create basic web server

* Create Node Express server as demonstrated in class including a 404 and 500 handler
* Create the following constants:
  + HOST, PORT, DIR (current path plus ‘/logs’), FILE\_LOG (DIR plus ‘/log.txt’), HTML (h1 header showing ‘In-Class Programming 13’)
* For the home route (/) send constant HTML (no need for complete html page markup)
* Assign the return value of app.listen into a constant named server
* Save everything and test out the server

**Step 2:** Prepare directory and file

* Load the filesystem module
* Create route /error and throw error with message ‘Something went wrong’
* Send constant HTML
* In app.listen callback, test for directory logs, if it does not exist, create it using method fs.existsSync and fs.mkdirSync (this happens only once when server is started)



**Step 3:** Create function fLog

* Create function fLog using parameter strMsg to write to logging information to file /logs/log.txt using fs.appendFileSync
* Save log information as shown above
* Notice that some entries contain date/time stamp, others do not
* Add date/time stamp in fLog using parameter passed whether to use date/time or not (proposed logic)
* For root request (“/”) only, include user-agent, hostname, ip and protocol (Use get method or convenience properties of req object)
* Call function fLog in app.listen, in both routes, and error handler passing message strings as shown

**Step 4:** (Completely optional)

* We also want to log when we shut down the server
* Use on event registration method on process object using ‘SIGINT’ as the event name and anonymous event handler
* In the event handler:
  + Call fLog twice using the messages ‘SIGINT signal received on: ’ and 'Closing http server on ‘
  + Close server by using close method on constant server, within callback call fLog again with message 'Http server closed on: ' then exit the process by using process.exit(0)

Upload the following files to Canvas:

* Screenshots:
  + Browser window showing the home route of server (either pasted into a Word document or as an image)
  + Node.js console showing executed icp13.js file
* The log file log.txt showing logging starting server and home and error routes
* The JavaScript file icp13.txt (txt extension)

**Notes:**

* Upload the files individually (no zipping, compressing)
* Your code should contain some meaningful comments (if applicable)
* Your code should be well organized and formatted