1 [1] In simplegrep_async.js:14, what is the type of all of the symbols referenced in the function? Specifically, what is the type of lines, rawContents, and split?

-RawContents is the raw data being passed in that contains your page data. It is a big string.
-Lines is a 'split' version of rawContents it is an array of strings.
-Split is a function which separates unparsed data. By default splits by each 'new line' character.

2 [1] If I create a line 24 in simplegrep_async.js that says 'console.log("finished!");', when will finished! be printed relative to the other output of the program?

The 'finished' line will print immediately after line 23 is read but not necessarily after it is done reading the file. If the file being read is a large file 'finished' will print before it is fully loaded because of the async nature of node.

3 [1] What is a one-line change to tinywebserver.js that would make every HTTP response have a 404 response code?

If line 48 was changed to say 'response.writeHead(404, {' then all pages would return 404.

4 [1] How you could return a 403 error (access denied) to all requests from iPhones in tinywebserver.js? (Remember the User-Agent HTTP header.)

In request handler you would modify line 92 to look for:
        'request.headers['user-agent'].indexOf('iPhone')>-1)'
aswell as special characters. 'User Agent: iPhone' is the header that specifies the user is browsing on iPhone, when this is true it will send a 403

5 [6] How would the behavior of form-demo changed when you make the following changes? Specifically, what changes would you see on the server and in the browser? Be sure to consider messages that are logged on the server and console output in the browser.

a   Comment out form-demo/bin/www:7
Commenting out line 7 will break the program and cause it to not compile due to 'app' being called later in the function on line 16

b   Change form-demo/bin/www:15 to be "var port=3000;".
The server will launch listening on port 3000 and regardless of the parameter passed it will always be port 3000. This would mean pages will not load if you aren't sending requests on port 3000

c   Change form-demo/app.js:13 to be "app.set('views', '.');"
This would set the app setting 'views' to be the '.' directory (root) rather then the one directory specified in "path.join(__dirname, 'views')". This means that the app wont be looking in the right folder and your webpage wont display.

d   Comment out form-demo/routes/index.js:24

'module.exports' will be replaced by 'router' which is equal to the 'express.Router()'. This will clear the libraries being used and replace them with 'router'. If this is commented out you wont be able to use any of the functions within the express.Router() which will break things further in the program

e   Comment out form-demo/views/layout.jade:8

When commenting out 'block content' the compiler will throw an error as it expects there to be an input to the body but there is nothing there. Additionally, when the index and add are read they extend layout which should specify where 'block content' is to be displayed, when the line is commented out this will cause an error or warning depending on your version of jade.

f   Change the indentation of form-demo/views/index.jade:24 such that it is at the same indentation as form in line 8.

If you change the indentation of the button is will no longer have a method or action associated with it. This would make the button still visible but have no function.