Ministry of Education of Republic of Moldova
Technical University of Moldova
Faculty of Computers, Informatics and Microelectronics
Software Engineering Department

Web Programming

Laboratory Work №2

Author:

Basso Nicolae

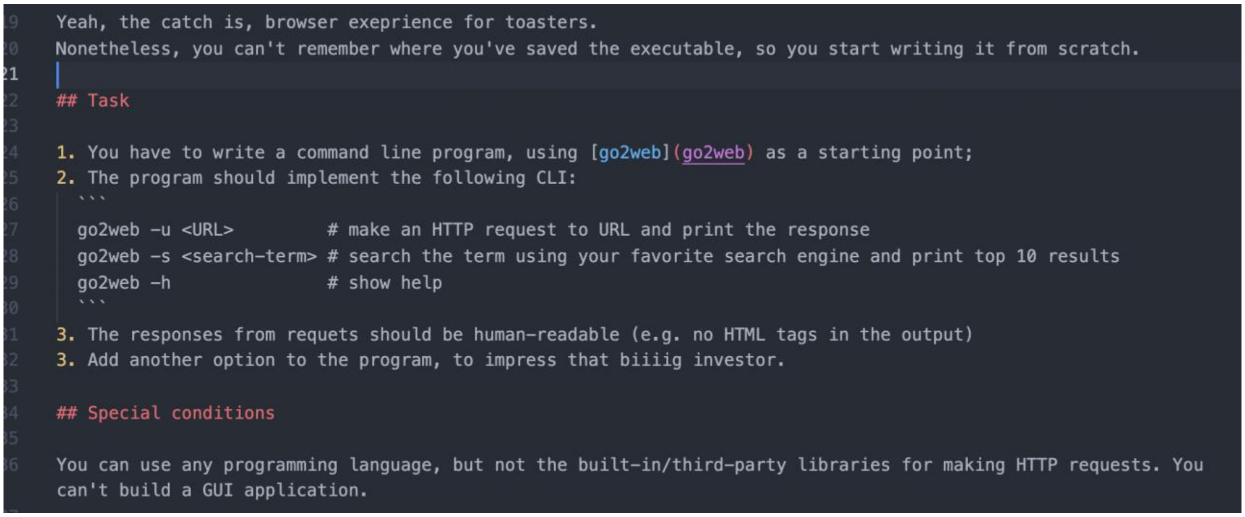
FAF-191

Supervisor

Alexei Șerșun

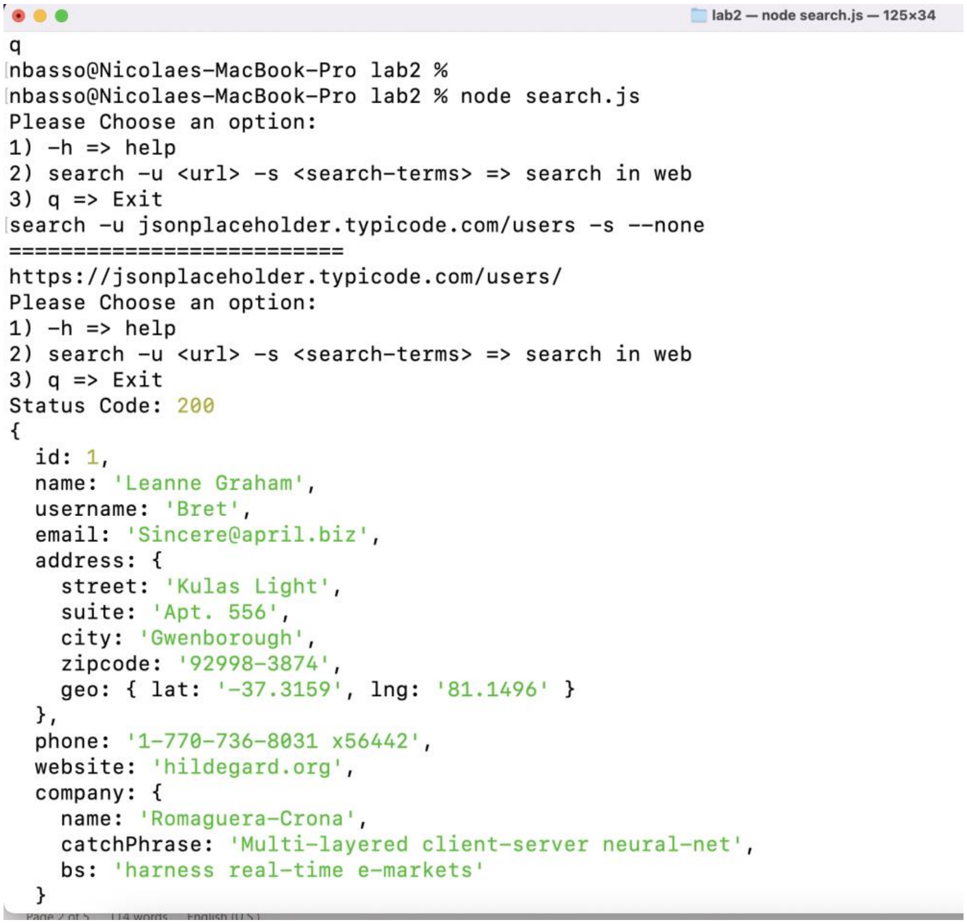
Chișinău 2022

1. Task

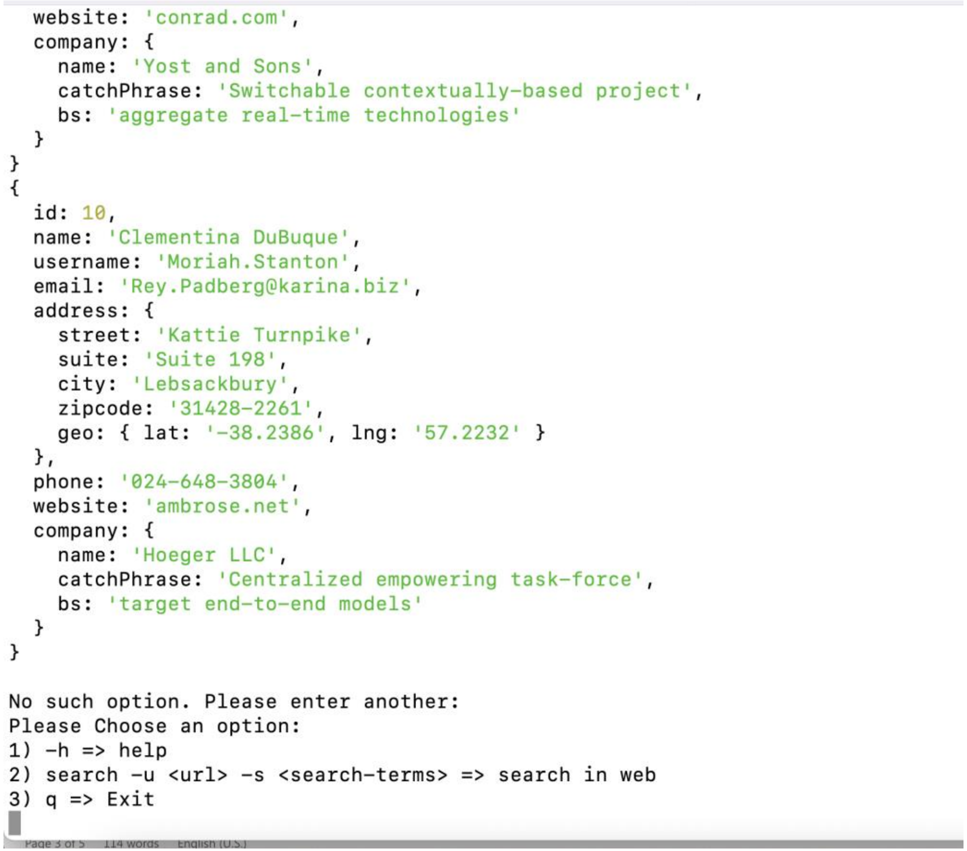


~~~
Yeah, the catch is, browser exeprience for toasters.
Nonetheless, you can't remember where you've saved the executable, so you start writing it from scratch.

## Task

1. You have to write a command line program, using [go2web](go2web) as a starting point;
2. The program should implement the following CLI:
  ```
  go2web -u <URL>         # make an HTTP request to URL and print the response
  go2web -s <search-term> # search the term using your favorite search engine and print top 10 results
  go2web -h               # show help
  ```
3. The responses from requets should be human-readable (e.g. no HTML tags in the output)
3. Add another option to the program, to impress that biiiig investor.

## Special conditions

You can use any programming language, but not the built-in/third-party libraries for making HTTP requests. You can't build a GUI application.
~~~

2. Results

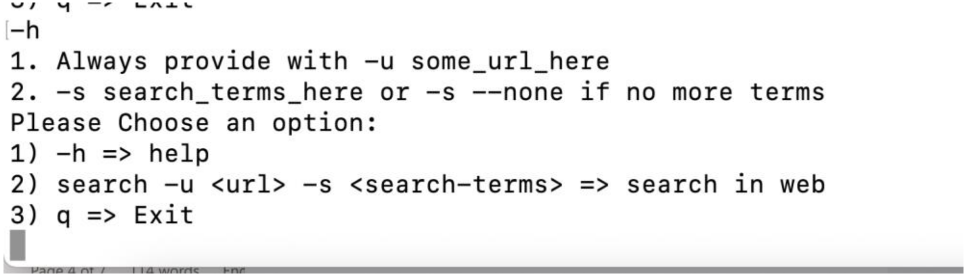


```
q
nbasso@Nicolaes-MacBook-Pro lab2 %
nbasso@Nicolaes-MacBook-Pro lab2 % node search.js
Please Choose an option:
1) -h => help
2) search -u <url> -s <search-terms> => search in web
3) q => Exit
search -u jsonplaceholder.typicode.com/users -s --none
==========================
https://jsonplaceholder.typicode.com/users/
Please Choose an option:
1) -h => help
2) search -u <url> -s <search-terms> => search in web
3) q => Exit
Status Code: 200
{
  id: 1,
  name: 'Leanne Graham',
  username: 'Bret',
  email: 'Sincere@april.biz',
  address: {
    street: 'Kulas Light',
    suite: 'Apt. 556',
    city: 'Gwenborough',
    zipcode: '92998-3874',
    geo: { lat: '-37.3159', lng: '81.1496' }
  },
  phone: '1-770-736-8031 x56442',
  website: 'hildegard.org',
  company: {
    name: 'Romaguera-Crona',
    catchPhrase: 'Multi-layered client-server neural-net',
    bs: 'harness real-time e-markets'
  }
```



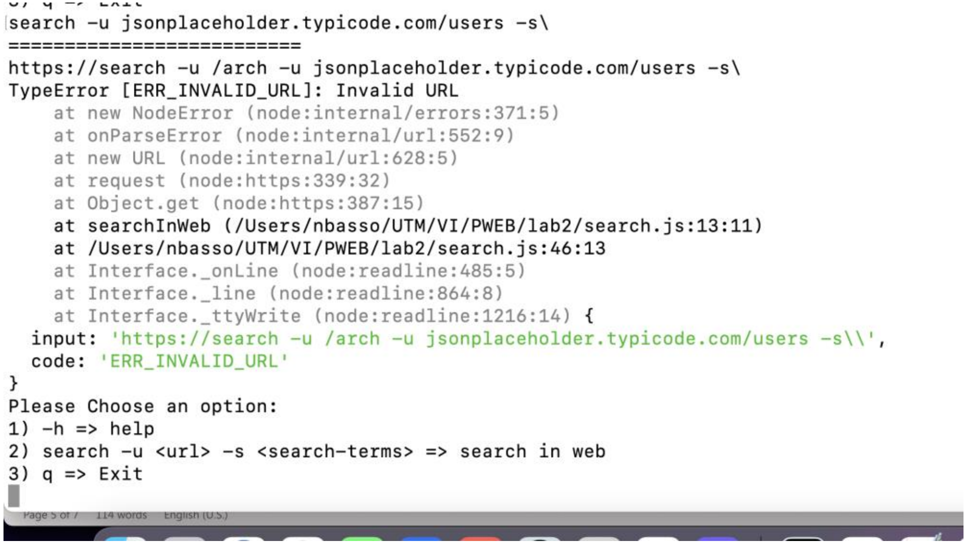
```
  website: 'conrad.com',
  company: {
    name: 'Yost and Sons',
    catchPhrase: 'Switchable contextually-based project',
    bs: 'aggregate real-time technologies'
  }
}
{
  id: 10,
  name: 'Clementina DuBuque',
  username: 'Moriah.Stanton',
  email: 'Rey.Padberg@karina.biz',
  address: {
    street: 'Kattie Turnpike',
    suite: 'Suite 198',
    city: 'Lebsackbury',
    zipcode: '31428-2261',
    geo: { lat: '-38.2386', lng: '57.2232' }
  },
  phone: '024-648-3804',
  website: 'ambrose.net',
  company: {
    name: 'Hoeger LLC',
    catchPhrase: 'Centralized empowering task-force',
    bs: 'target end-to-end models'
  }
}

No such option. Please enter another:
Please Choose an option:
1) -h => help
2) search -u <url> -s <search-terms> => search in web
3) q => Exit
```

Page 3 of 5   114 words   English (U.S.)

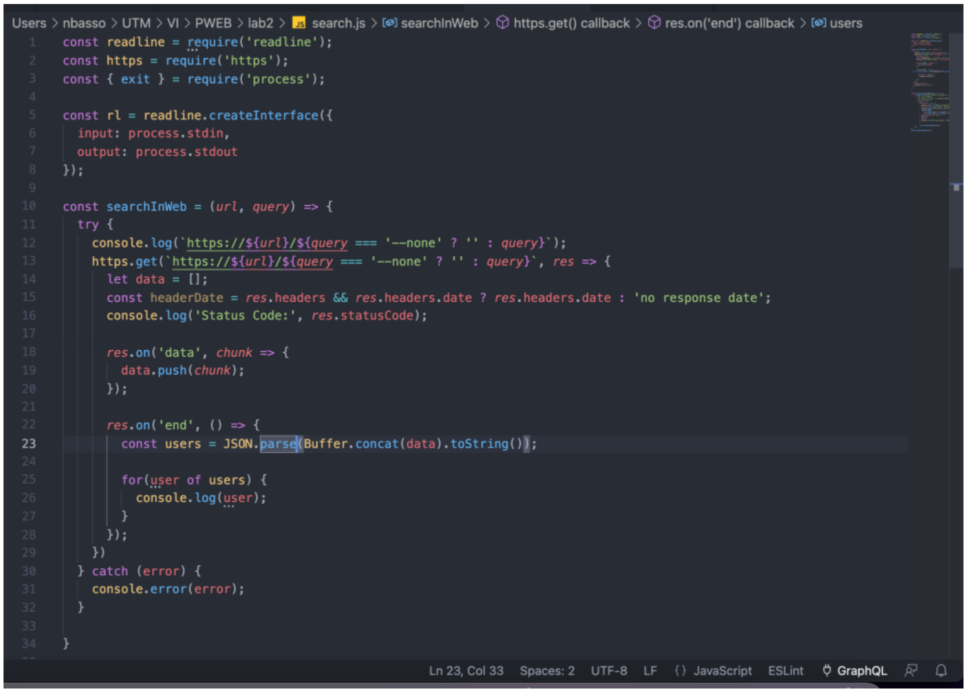


```
-h
1. Always provide with -u some_url_here
2. -s search_terms_here or -s --none if no more terms
Please Choose an option:
1) -h => help
2) search -u <url> -s <search-terms> => search in web
3) q => Exit
```

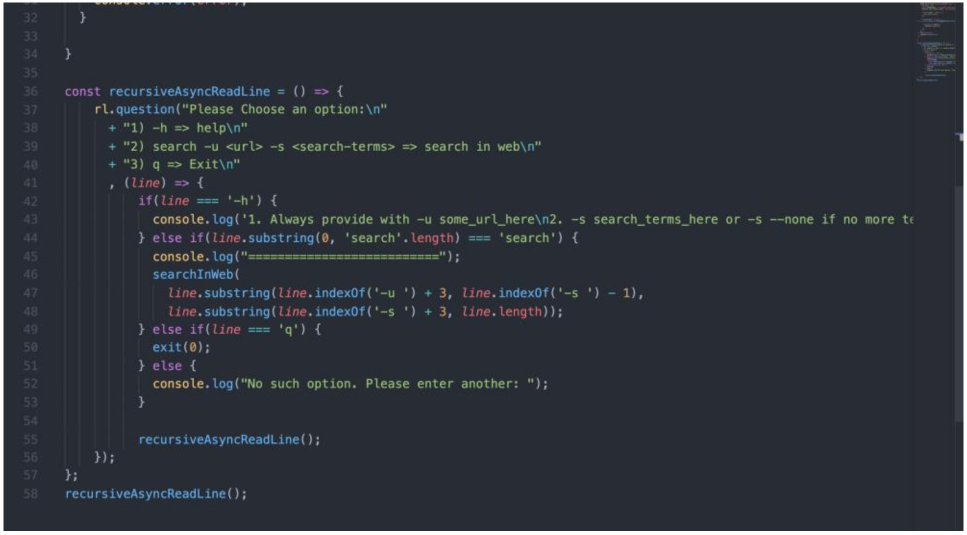
Page 4 of 7   114 words   Eng



```
search -u jsonplaceholder.typicode.com/users -s\
==========================
https://search -u /arch -u jsonplaceholder.typicode.com/users -s\
TypeError [ERR_INVALID_URL]: Invalid URL
    at new NodeError (node:internal/errors:371:5)
    at onParseError (node:internal/url:552:9)
    at new URL (node:internal/url:628:5)
    at request (node:https:339:32)
    at Object.get (node:https:387:15)
    at searchInWeb (/Users/nbasso/UTM/VI/PWEB/lab2/search.js:13:11)
    at /Users/nbasso/UTM/VI/PWEB/lab2/search.js:46:13
    at Interface._onLine (node:readline:485:5)
    at Interface._line (node:readline:864:8)
    at Interface._ttyWrite (node:readline:1216:14) {
  input: 'https://search -u /arch -u jsonplaceholder.typicode.com/users -s\\',
  code: 'ERR_INVALID_URL'
}
Please Choose an option:
1) -h => help
2) search -u <url> -s <search-terms> => search in web
3) q => Exit
```


Users > nbasso > UTM > VI > PWEB > lab2 > search.js > searchInWeb > https.get() callback > res.on('end') callback > users
Ln 23, Col 33
Spaces: 2
UTF-8
LF
{} JavaScript
ESLint
GraphQL

```javascript
const readline = require('readline');
const https = require('https');
const { exit } = require('process');

const rl = readline.createInterface({
  input: process.stdin,
  output: process.stdout
});

const searchInWeb = (url, query) => {
  try {
    console.log(`https://${url}/${query === '--none' ? '' : query}`);
    https.get(`https://${url}/${query === '--none' ? '' : query}`, res => {
      let data = [];
      const headerDate = res.headers && res.headers.date ? res.headers.date : 'no response date';
      console.log('Status Code:', res.statusCode);

      res.on('data', chunk => {
        data.push(chunk);
      });

      res.on('end', () => {
        const users = JSON.parse(Buffer.concat(data).toString());

        for(user of users) {
          console.log(user);
        }
      });
    })
  } catch (error) {
    console.error(error);
  }

}
```



```javascript
  }

}

const recursiveAsyncReadLine = () => {
    rl.question("Please Choose an option:\n"
      + "1) -h => help\n"
      + "2) search -u <url> -s <search-terms> => search in web\n"
      + "3) q => Exit\n"
      , (line) => {
        if(line === '-h') {
          console.log('1. Always provide with -u some_url_here\n2. -s search_terms_here or -s --none if no more te
        } else if(line.substring(0, 'search'.length) === 'search') {
          console.log("==========================");
          searchInWeb(
            line.substring(line.indexOf('-u ') + 3, line.indexOf('-s ') - 1),
            line.substring(line.indexOf('-s ') + 3, line.length));
        } else if(line === 'q') {
          exit(0);
        } else {
          console.log("No such option. Please enter another: ");
        }

        recursiveAsyncReadLine();
    });
};
recursiveAsyncReadLine();
```

3. Conclusion

Improved my skills at doing labs through pain of reluctance...

References:

NodeJS