

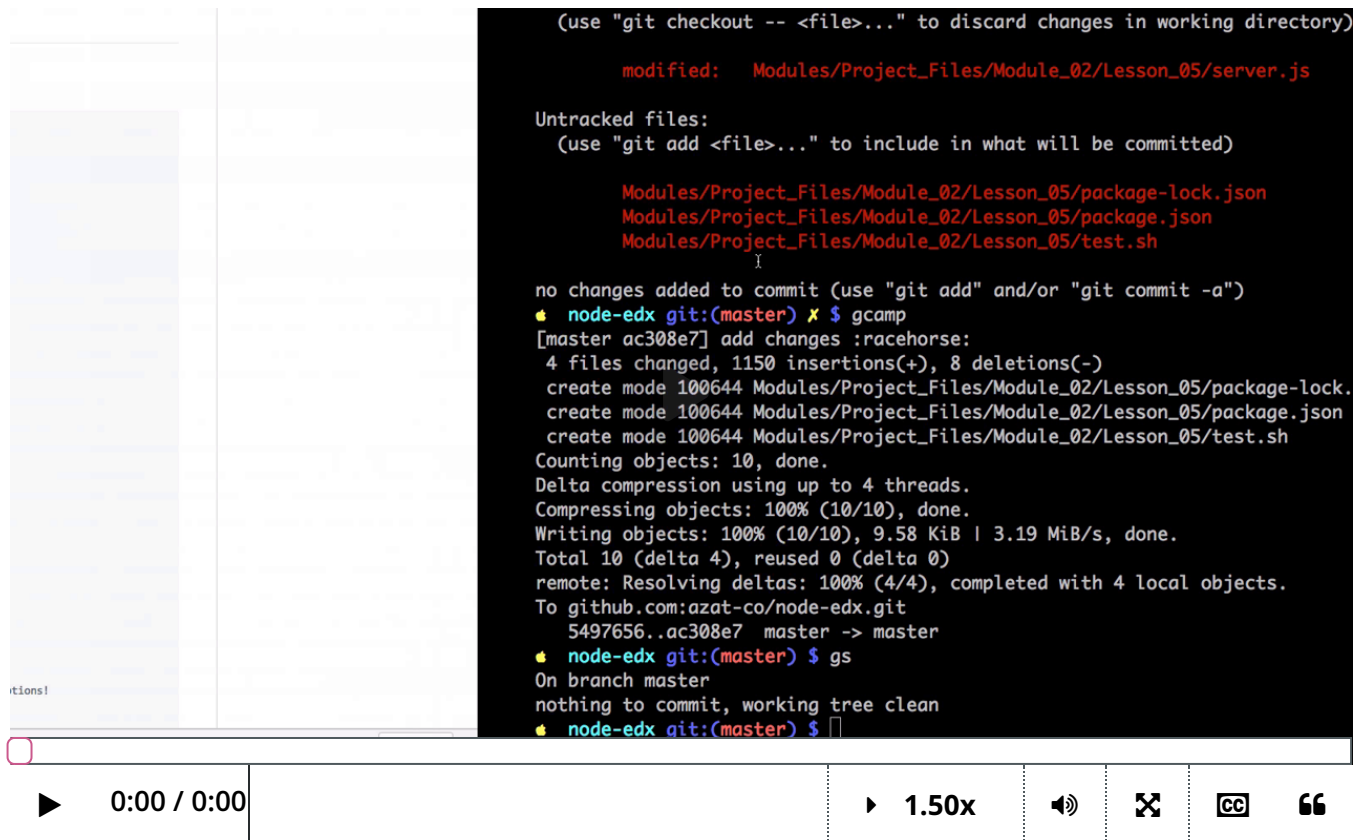
EdX and its Members use cookies and other tracking technologies for performance, analytics, and marketing purposes. By using this website, you accept this use. Learn more about these technologies in the [Privacy Policy](#).



[Course](#) > [Expres...](#) > [URL Pa...](#) > [Input v...](#)

Input validation

Video: Input Validation



The video player displays a terminal window with the following content:

```
(use "git checkout -- <file>..." to discard changes in working directory)

modified:   Modules/Project_Files/Module_02/Lesson_05/server.js

Untracked files:
(use "git add <file>..." to include in what will be committed)

Modules/Project_Files/Module_02/Lesson_05/package-lock.json
Modules/Project_Files/Module_02/Lesson_05/package.json
Modules/Project_Files/Module_02/Lesson_05/test.sh
I
no changes added to commit (use "git add" and/or "git commit -a")
node-edx git:(master) X $ gcamp
[master ac308e7] add changes :racehorse:
4 files changed, 1150 insertions(+), 8 deletions(-)
create mode 100644 Modules/Project_Files/Module_02/Lesson_05/package-lock.
create mode 100644 Modules/Project_Files/Module_02/Lesson_05/package.json
create mode 100644 Modules/Project_Files/Module_02/Lesson_05/test.sh
Counting objects: 10, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (10/10), done.
Writing objects: 100% (10/10), 9.58 KiB | 3.19 MiB/s, done.
Total 10 (delta 4), reused 0 (delta 0)
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
To github.com:azat-co/node-edx.git
5497656..ac308e7 master -> master
node-edx git:(master) $ gs
On branch master
nothing to commit, working tree clean
node-edx git:(master) $
```

Video player controls: 0:00 / 0:00, 1.50x, volume, full screen, CC, and quote icons.

Video

[Download video file](#)

Transcripts

[Download SubRip \(.srt\) file](#)

[Download Text \(.txt\) file](#)

Input Validation

It is very important to validate the incoming data. Never trust the client. The data can be malformed causing your app to crash or just malicious on purpose if a client is an attacker.

A manual validation can be done in each route which accepts data. If it's in the request body, you can use an if/else statement:

```
app.post('/login', (req, res) => {  
  if (!req.body.email || !req.body.password)  
    return res.send({  
      error: 'Please enter your email and password.'  
    })  
  if (!validateEmail(req.body.email) || ! validatePassword(req.body.password))  
    return res.send({  
      error: 'Invalid format for email and/or password.'  
    })  
  login(req.body.email, req.body.password)  
})
```

A better way is to use [express-validator](#) because it allows you to use a schema.