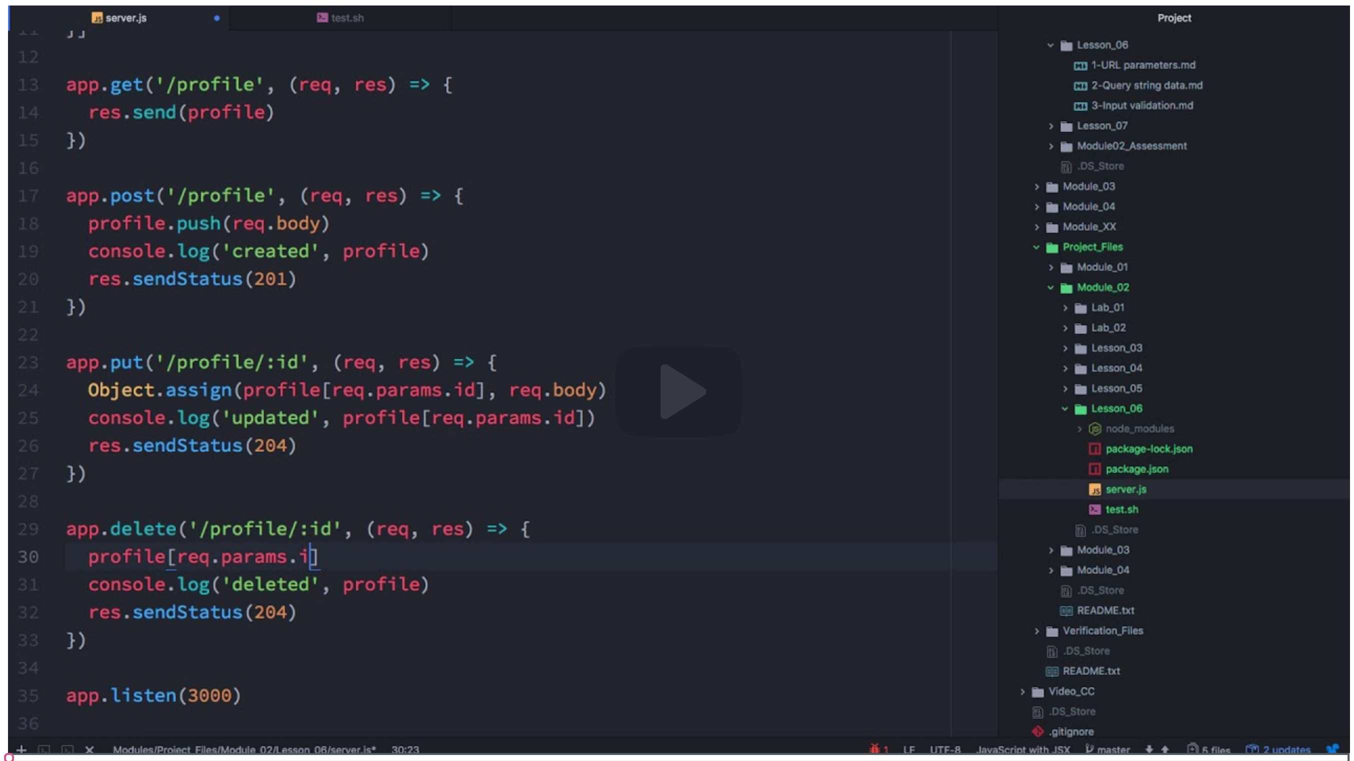
Course › Expres… › URL Pa… › URL pa…

URL parameters

Video: Accessing URL Parameters



```
})

app.get('/profile', (req, res) => {
  res.send(profile)
})

app.post('/profile', (req, res) => {
  profile.push(req.body)
  console.log('created', profile)
  res.sendStatus(201)
})

app.put('/profile/:id', (req, res) => {
  Object.assign(profile[req.params.id], req.body)
  console.log('updated', profile[req.params.id])
  res.sendStatus(204)
})

app.delete('/profile/:id', (req, res) => {
  profile[req.params.i]
  console.log('deleted', profile)
  res.sendStatus(204)
})

app.listen(3000)
```

0:00 / 0:00 1.50x

Video

Download video file

Transcripts

Download SubRip (.srt) file

Download Text (.txt) file

Accessing URL Parameters

To access URL parameters such as for IDs, names or other information, define the parameter in the URL pattern string with a colon `:` and then access the parameter with `req.params`.

For example, for requests like this one which has an user ID after the string `/users/`:

```
GET /users/572611d856b11dcec61651bb
```

Use a URI segment parameterized by prefixing it with a semi-colon:

```
app.get('/users/:id', (request, response) => {
  const userId = request.params.id
  fetchUser(userId, (error, user) => {
    if (error) return response.status(500).send(error)
    response.send(user)
  })
})
```

Multiple URL Parameters

Express supports multiple URL parameters in a route. Simply define all of them in the URL pattern. For example for the request GET /users/:id/transactions/:transactionId/:filter, the route will look like:

```
app.get('/users/:id/transactions/:transactionId/:filter', (req, res)
  const usersId = request.params.id,
    transactionId = request.params.transactionId,
    filter = request.params.filter
  res.status(200).send()
})
```