Parsing Requests in Node.js

(using Express.js)

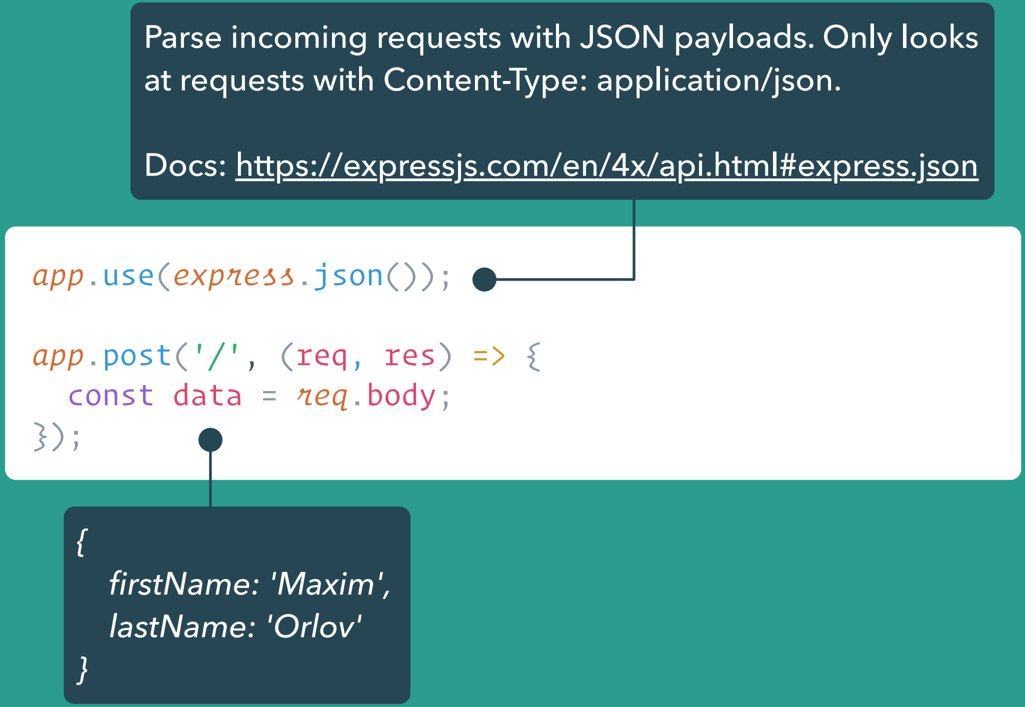
ⓘ Printable in A4 format (landscape)

Frontend

- application/json

```
const data = {
  firstName: 'Maxim',
  lastName: 'Orlov',
};

await fetch('/', {
  method: 'POST',
  headers: {
    'Content-Type': 'application/json',
  },
  body: JSON.stringify(data),
});
```

Backend



Parse incoming requests with JSON payloads. Only looks at requests with Content-Type: application/json.

Docs: https://expressjs.com/en/4x/api.html#express.json

```
app.use(express.json());

app.post('/', (req, res) => {
  const data = req.body;
});
```

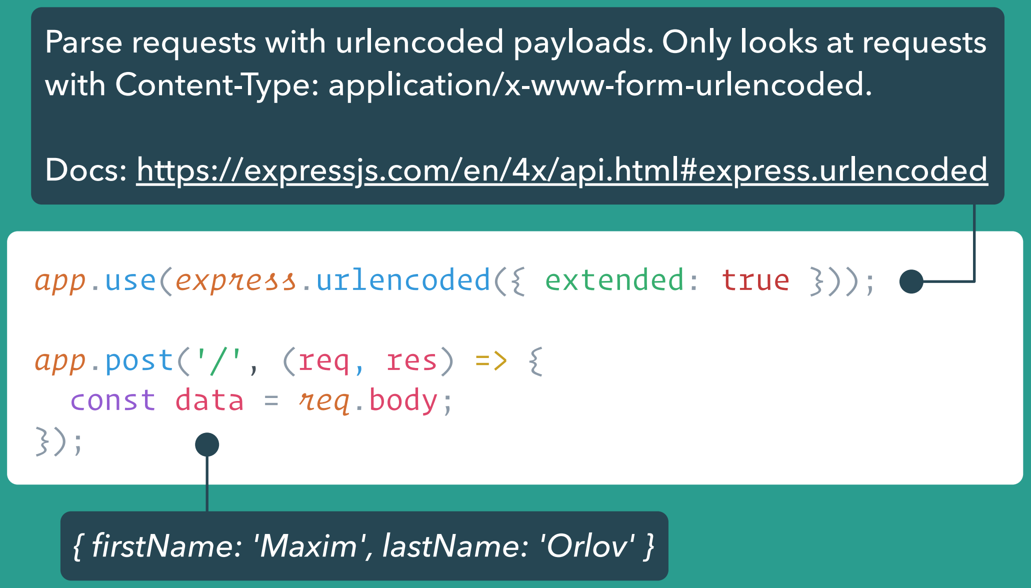
```
{
  firstName: 'Maxim',
  lastName: 'Orlov'
}
```

Made with 💛 by Maxim Orlov

🐦 @_maximization 🌐 https://maximorlov.com

- application/x-www-form-urlencoded

```html
<form action="/" method="POST">
  <input type="text" name="firstName" value="Maxim" />
  <input type="text" name="lastName" value="Orlov" />
  <input type="submit" value="Send" />
</form>
```



Parse requests with urlencoded payloads. Only looks at requests with Content-Type: application/x-www-form-urlencoded.

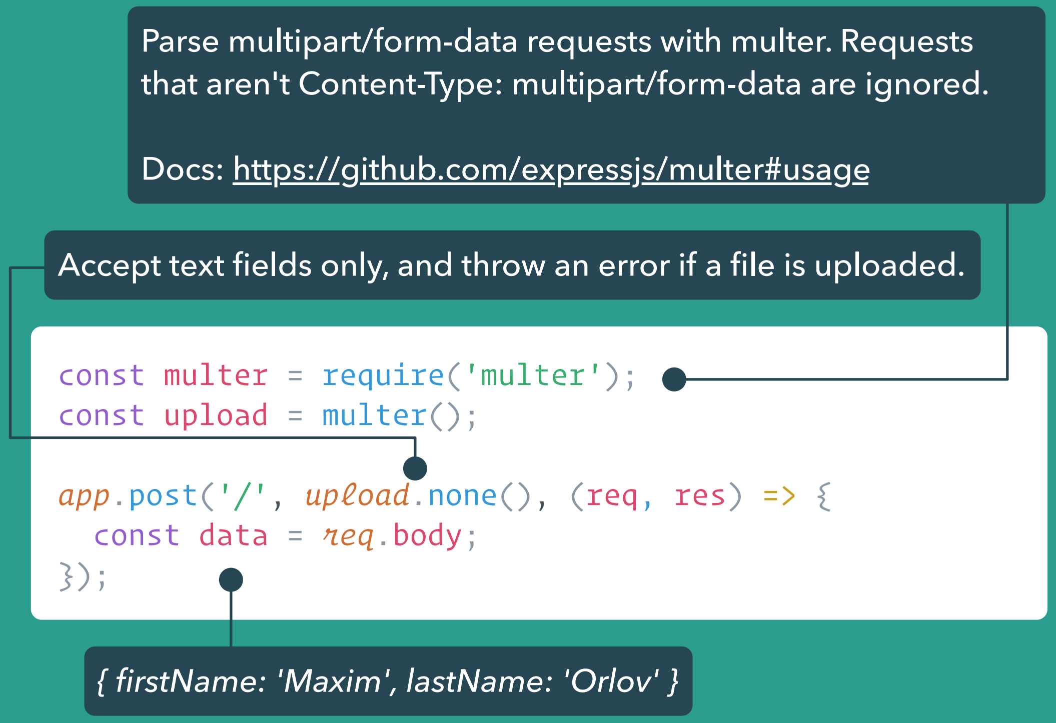
Docs: https://expressjs.com/en/4x/api.html#express.urlencoded

```js
app.use(express.urlencoded({ extended: true }));

app.post('/', (req, res) => {
  const data = req.body;
});
```

{ firstName: 'Maxim', lastName: 'Orlov' }

- multipart/form-data (no file)

```html
<form
  action="/"
  method="POST"
  enctype="multipart/form-data"
>
  <input type="text" name="firstName" value="Maxim" />
  <input type="text" name="lastName" value="Orlov" />
  <input type="submit" value="Send" />
</form>
```



Parse multipart/form-data requests with multer. Requests that aren't Content-Type: multipart/form-data are ignored.

Docs: https://github.com/expressjs/multer#usage

Accept text fields only, and throw an error if a file is uploaded.

```js
const multer = require('multer');
const upload = multer();

app.post('/', upload.none(), (req, res) => {
  const data = req.body;
});
```

{ firstName: 'Maxim', lastName: 'Orlov' }

Made with 💛 by Maxim Orlov
@_maximization https://maximorlov.com

- ## multipart/form-data (single file)

```html
<form
  action="/"
  method="POST"
  enctype="multipart/form-data"
>
  <input type="text" name="firstName" value="Maxim" />
  <input type="text" name="lastName" value="Orlov" />
  <input type="file" name="avatar" />
  <input type="submit" value="Send" />
</form>
```

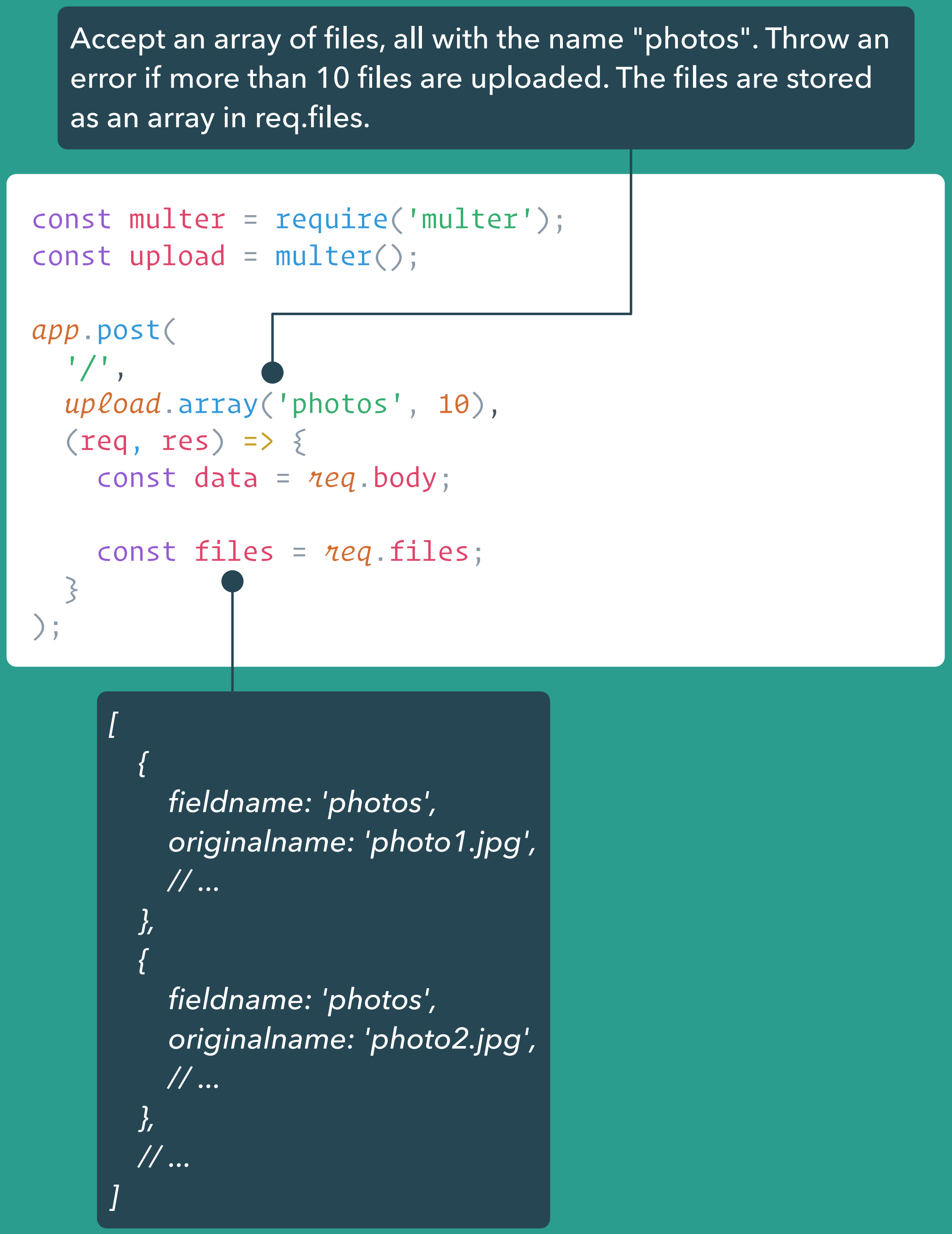
Accept a single file with the name "avatar". The file is stored in req.file and text fields in req.body.

```js
const multer = require('multer');
const upload = multer();

app.post('/', upload.single('avatar'), (req, res) => {
  const data = req.body;

  const file = req.file;
});
```

```
{
   fieldname: 'avatar',
   originalname: 'my-avatar.jpg',
   encoding: '7bit',
   mimetype: 'image/jpeg',
   size: 77434,
   buffer: <Buffer ff d8 ff e0 00 10 4a 46 49 46 00 01
01 00 00 48 00 48 00 00 ff e1 00 80 45 78 69 66 00
00 4d 4d 00 2a 00 00 00 08 00 04 01 1a 00 05 00
00 00 01 00 00 ... 77384 more bytes>
}
```

• multipart/form-data (multiple files, single field)

```html
<form
  action="/"
  method="POST"
  enctype="multipart/form-data"
>
  <input type="text" name="firstName" value="Maxim" />
  <input type="text" name="lastName" value="Orlov" />
  <input type="file" name="photos" multiple />
  <input type="submit" value="Send" />
</form>
```



Accept an array of files, all with the name "photos". Throw an error if more than 10 files are uploaded. The files are stored as an array in req.files.

```js
const multer = require('multer');
const upload = multer();

app.post(
  '/',
  upload.array('photos', 10),
  (req, res) => {
    const data = req.body;

    const files = req.files;
  }
);
```

```
[
  {
    fieldname: 'photos',
    originalname: 'photo1.jpg',
    // ...
  },
  {
    fieldname: 'photos',
    originalname: 'photo2.jpg',
    // ...
  },
  // ...
]
```

- **multipart/form-data (multiple files, multiple fields)**

```html
<form
  action="/"
  method="POST"
  enctype="multipart/form-data"
>
  <input type="text" name="firstName" value="Maxim" />
  <input type="text" name="lastName" value="Orlov" />
  <input type="file" name="avatar" />
  <input type="file" name="photos" multiple />
  <input type="submit" value="Send" />
</form>
```

Accept a mix of files specified by an array of objects with `name` and `maxCount`. An object with array of files is stored in req.files.

```js
const multer = require('multer');
const upload = multer();

app.post(
  '/',
  upload.fields([
    { name: 'avatar', maxCount: 1 },
    { name: 'photos', maxCount: 10 },
  ]),
  (req, res) => {
    const data = req.body;

    const files = req.files;
  }
);
```

```
{
  avatar: [
    { fieldname: 'avatar', originalname: 'my-avatar.jpg', ... }
  ],
  photos: [
    { fieldname: 'photos', originalname: 'photo1.jpg', ... },
    { fieldname: 'photos', originalname: 'photo2.jpg', ... },
    // ...
  ]
}
```

Made with 💛 by Maxim Orlov
@_maximization https://maximorlov.com