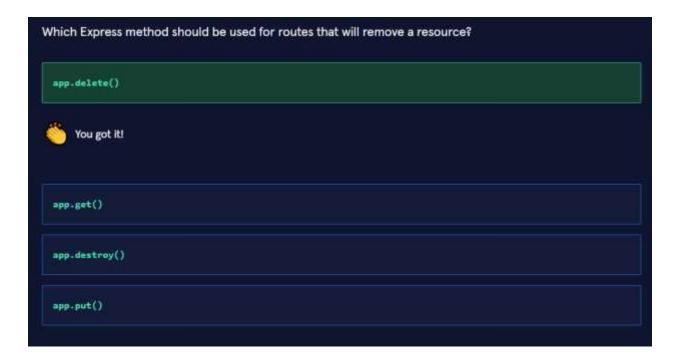
```
What will this Express server log in response to GET /fruits?
 const express = require('express');
 const app = express();
 const fruits = ['apple', 'tomato', 'pear'];
 app.get('/fruits/:id', (req, res, next) => {
   console.log(`request to /fruits/${req.params.id}`);
   res.send(fruits[req.params.id]);
 });
 app.get('/fruits', (req, res, next) => {
   console.log('request to /fruits, looking good!');
   res.send(fruits);
 app.get('/fruits', (req, res, next) => {
   console.log('request to /fruits, feeling good!');
   res.send(fruits);
 request to /fruits, looking good!
      You got it!
```



```
Which code will set a status code of 200 and send an empty response?

app.status(200).send();

res.status(200).send();

res.status(200).send();

You got it!
```

```
Which route will GET /dogs/retriever/1 match?

app.get('/dogs/:breed/:id', (req, res, next) => {
    ));

You got it!
```

```
What will req.query look like in a route that matches a GET request to /animals?color=green&age=4&eyes=2

['color', 'age', 'eyes']

'color-green&age=4&eyes=2'

{ color: 'green', age: '4', eyes: '2', route: '/animals' }

{ color: 'green', age: '4', eyes: '2')

You got it!
```

```
If this code has been run in Node on your computer, why will a set request to localhost:4001/ fail?

const express = require('express');
const app = express();
const port = 8000;
const expressIs = ['fun', 'rewarding', 'servertastic'];
app.get('/', (req, res, next) => {
    res.send(expressIs);
));
app.listen(port);

There is no route registered to handle that path.

The server is listening on port 8000, not 4001.
```

In the code snippet below, what value should the slothPath string have in order for the route to match a GET

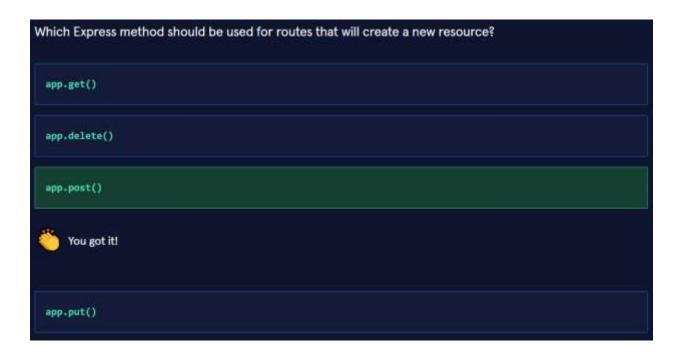
/animals/sloths request?

const express = require('express');
const app = express();
const router = express.Router();
const sloths = ['slowpoke', 'fluffles', 'sonic'];
app.use('/animals', router);
let slothPath = // assign slothPath here
router.get(slothPath, (req, res, next) -> {
 res.send(sloths);
});

'/animals/sloths'

'/sloths'

You got it!



```
Which Express method will start a server?

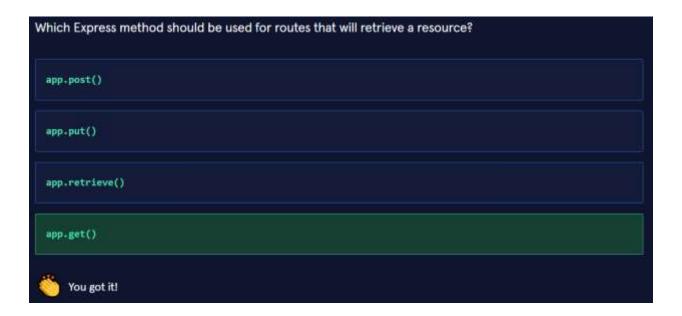
app.server()

app.run()

app.get()

app.listen()

You got it!
```



```
Which code will create an Express server instance and save it to the variable app?

const express = require('express');
const app = express();

You got it!
```

```
Which Express server method is used to update a resource?

app.get()

app.put()

You got it!
```

```
Which express method sends a response to the client?

app.send()

res.send()

res.respond()

req.send()
```

```
What will req.params equal in this route with a GET request to /users/user_1/purchases/100?

app.get('/users/:name/purchases/:purchaseId', (req, res, next) => {
});

{ ':name': 'user_1', ':purchaseId': '100')

{ name: 'user_1', purchaseId: 100 }

?name=user_1&purchaseId=100

{ name: 'user_1', purchaseId: '100' }

Correct! Both parameters are strings
```