If you're not getting `id` and `token` from `req.query`, there might be a few reasons for this issue. Here are some things you can check:

1. \*\*URL Parameters in the Route Definition:\*\*

Ensure that the route definition in your Express application includes these parameters. For example, in your `routes.js` or wherever you define your routes, make sure you have something like this:

```javascript

// routes.js

const express = require('express');

const router = express.Router();

const emailVerificationController = require('./controllers/emailVerificationController');

// Define routes

router.get('/email/verify/:id/:token', emailVerificationController);

module.exports = router;

```

With this definition, you should be able to access `req.params.id` and `req.params.token` in your controller.

2. \*\*Check the URL Format:\*\*

Ensure that the URL you are using to make the request has the correct format with the `id` and `token` parameters. For example:

```

http://yourdomain.com/email/verify/123456789abcdef/yourtoken

```

3. \*\*Query Parameters Instead of Route Parameters:\*\*

If you intend to use query parameters (e.g., `/email/verify?id=123&token=yourtoken`), then update your route definition and access them using `req.query`:

```javascript

// routes.js

router.get('/email/verify', emailVerificationController);

```

In your controller:

```javascript

const { id, token } = req.query;

```

Adjust the route definition based on how you want to structure your URLs.

Double-check these aspects to ensure that the route parameters or query parameters are set up correctly and that you're using the correct URL format when making the request.