## **Matching Longer Paths**

## 2 min

Parameters are extremely helpful in making server routes dynamic and able to respond to different inputs. Route parameters will match anything in their specific part of the path, so a route matching /monsters/:name would match all the following request paths:

```
/monsters/hydra
/monsters/jörmungandr
/monsters/manticore
/monsters/123
```

In order for a request to match a route path, it must match the entire path, as shown in the diagram to the right. The request arrives for /expressions/1. It first tries to match the /expressions route, but because it has additional path segments after /expressions, it does not match this route and moves on to the next. It matches /expressions/:id because :id will match any value at that level of the path segment. The route matches, so the Express server calls the callback function, which in turn handles the request and sends a response.

## Instructions

Move on to the next exercise when you're ready.

```
Request

GET /expressions/1

checks the block and fails

incorrect app.get('/expressions', (req, res, next) => {

res.send(getAllExpressions());
});

checks the block and passes

correct action / correct route (has /1)

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

const expression = getExpressionById(req.params.id);

res.send(expression);
});

Sends
Response
```