

# **Express #5:**

# **Route Regular Expressions**

# Advanced Routing: Regular Expressions

- In addition to regular characters, routes can also use special characters to match more than just the specified route
- \* - Matches any and all characters after it
  - `"/user/*"` matches `/user/will`, `/user/tom`, `/user/betty` etc.
- ? - Makes a character or set of characters optional
  - `"/home(page)?"` matches `/home` or `/homepage`
- + - Matches any amount of a character
  - `"/sleep/z+"` matches `/sleep/zzz`, `/sleep/zzzzzzzzzz` etc.

# Regular Expressions (cont.)

- The main one we use is `*`, as we often want to match any path after some amount, but may not use it as a param
- On top of that, since routes are resolved in order, we can add a `"*"` route at the end to catch all incorrect paths
- A route like this would typically go at the bottom of your `app.js`

```
// After all other routes...
app.get("*", function(req, res) {
  res.send("This is not a valid page, go away!");
});
```

# Exercise: 404 Page

- Using what we just learned, create a route that catches all routes that you haven't specified
- Have it set the status to 404, and render a 404 page in your template
- 404 pages are often funny, artistic, or otherwise interesting, so take the rest of class to make a really cool 404
- Some 404 examples are [AirBnB](#), [MailChimp](#), or [Bloomberg](#).

# Additional Reading

- Express' Guide to Route Paths
- Express Route Tester - Try out complex routes here!