The **path()** function is passed four arguments, two required: **route** and **view**, and two optional: **kwargs**, and **name**. At this point, it's worth reviewing what these arguments are for.

path() argument: route

route is a string that contains a URL pattern. When processing a request, Django starts at the first pattern in **urlpatterns** and makes its way down the list, comparing the requested URL against each pattern until it finds one that matches.

Patterns don't search GET and POST parameters, or the domain name. For example, in a request to **https://www.example.com/myapp/**, the URLconf will look for **myapp/**. In a request to **https://www.example.com/myapp/?page=3**, the URLconf will also look for **myapp/**.

---

path() argument: view

When Django finds a matching pattern, it calls the specified view function with an **HttpRequest** object as the first argument and any "captured" values from the route as keyword arguments. We'll give an example of this in a bit.

---

path() argument: kwargs

Arbitrary keyword arguments can be passed in a dictionary to the target view. We aren't going to use this feature of Django in the tutorial.

---

path() argument: name

Naming your URL lets you refer to it unambiguously from elsewhere in Django, especially from within templates. This powerful feature allows you to make global changes to the URL patterns of your project while only touching a single file.

When you're comfortable with the basic request and response flow, read part 2 of this tutorial to start working with the database.