



WEB DEVELOPMENT

Lesson 11

Building REST APIs With Django

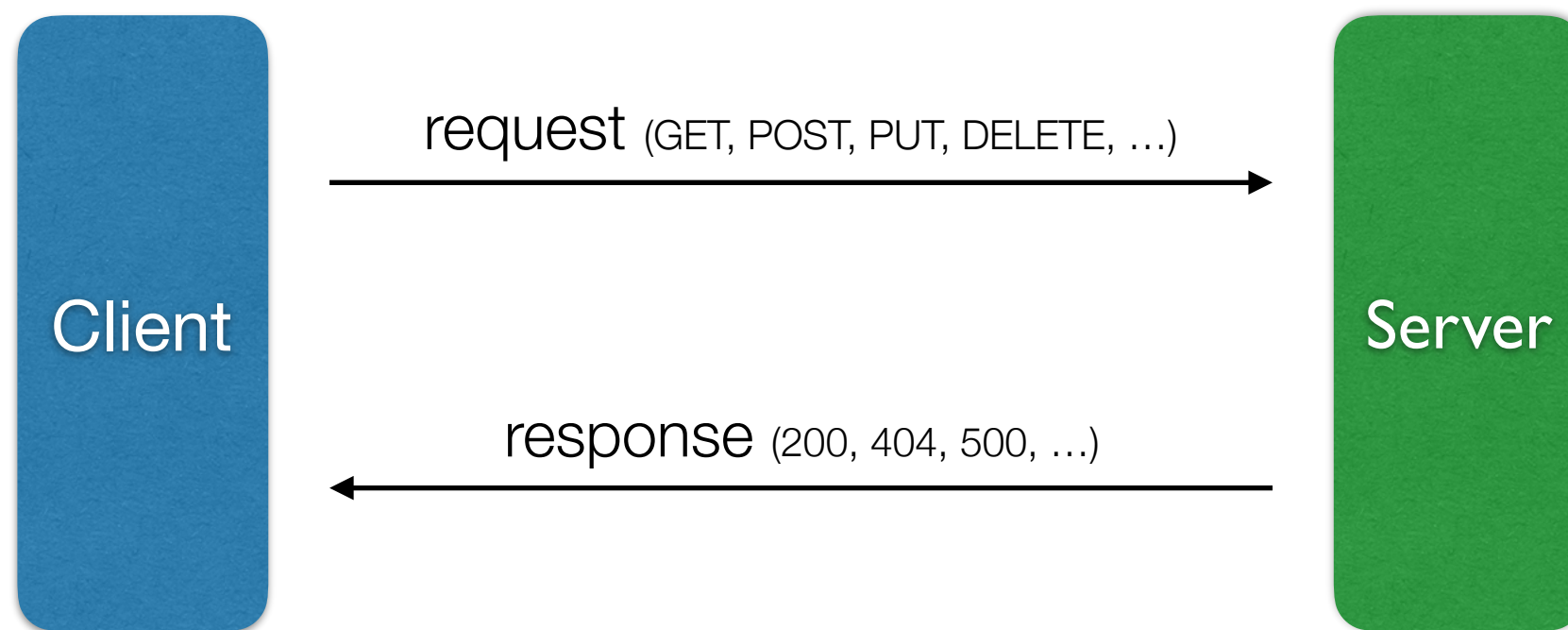
What is REST API?

Representational State Transfer (REST) Application Programming Interface (API)

HTTP method for each action

Purpose of Request	HTTP Method	Rough SQL equivalent
Create a new resource	POST	INSERT
Read an existing resource	GET	SELECT
Update an existing resource	PUT	UPDATE
Update part of an existing resource	PATCH	UPDATE
Delete an existing resource	DELETE	DELETE
Returns same HTTP headers as GET, but no body content	HEAD	
Return the supported HTTP methods for the given URL	OPTIONS	

Client Server Communication



Postman

<https://www.getpostman.com/>

Example with JSON response

Selecting Objects

```
>>> products = Product.objects.all()
```

```
SELECT * FROM products;
```

Filtering Data

```
>>> p = Product.objects.filter(name="Test name")
```

```
SELECT * FROM products  
WHERE name='Test name';
```

Filtering Data

- `contains / icontains`
- `startswith / endswith`
- `istartswith / iendswith`
- `exact / iexact`
- `in`
- `isnull`
- `gt / gte`
- `lt / lte`

```
>>> p = Product.objects.filter(name__contains="Test")
```

Retrieving Single Objects

```
>>> p = Product.objects.get(id=3)
```

```
SELECT * FROM product WHERE id=3;
```

Ordering Data

```
>>> p = Product.objects.order_by("name")
```

```
SELECT * FROM products ORDER BY name;
```

Ordering Data

```
>>> p = Product.objects.order_by("name", "address")
```

```
>>> p = Product.objects.order_by("-name")
```

Meta class

```
class Product(models.Model):  
    ...  
    class Meta:  
        ordering = ["name"]
```

Chaining Lookups

```
>>> p = Product.objects.filter(name="Laptop").order_by("-price")
```


Deleting Objects

```
>>> p = Product.objects.get(id=3)
```

```
>>> p.delete()
```

```
>>> products = Product.objects.all()
```

```
>>> products.delete()
```

CRUD API

Questions?