

```
@PutMapping("/{id}")
public Product updateProduct(@PathVariable("id") int id, @RequestBody Product p) throws EntityNotFoundException {
    return service.updateProduct(id, p);
}
```

```
@Transactional
public Product updateProduct(int id, Product p) throws EntityNotFoundException {
    Product product = productDao.findById(id).get();
    product.setPrice(p.getPrice()); // just modifying price; ==> DIRTY Checking --> update
    return product;
}
```

Which product to update

PUT http://localhost:8080/api/products/2

Params Authorization Headers (10) Body Pre-request Script Tests Settings

Headers 8 hidden

	Key	Value
<input checked="" type="checkbox"/>	Accept	application/json
<input checked="" type="checkbox"/>	Content-Type	application/json
	Key	Value

PUT http://localhost:8080/api/products/2

Params Authorization Headers (10) Body Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL JSON

```

1 {
2   ... "price": 99009.99
3 }
```