

Tutorial: Compare Shipping Rates

In this tutorial, you'll compare shipping rates of three different carriers for one package.

Tools You'll Need

There's [many great developer tools](#) that you can use to follow along with this tutorial. We'll use `curl` for all of our examples, since it's already pre-installed on most modern operating systems. If you're not comfortable working in the terminal, or you just prefer GUI tools, then we recommend [Postman](#) or [Insomnia](#). Both of these tools allow you to import `curl` commands, so you can still follow this tutorial.

Preparing the HTTP Request

API Endpoint

To validate an address, you send an HTTP `POST` request to `https://api.shipengine.com/v1/rates/`. From now on we'll refer to this as the `/v1/rates` API endpoint to keep things short.

Headers

There's two [HTTP headers](#) that you need to set in your request:

| one.

Request Body

The body of your HTTP request is where you specify the information to compare rates. We'll use the following information in the table below for our request body:

Field	Value
carrier_ids	UPS: se-108562 ;FedEx: se-108563 ;Stamps.com: se-108487
ship_to	Jane Doe 525 S Winchester Blvd San Jose, CA 95128
phone	555-555-5555
ship_from	John Doe 4009 Marathon Blvd Austin, TX 78756
weight	22.0 ounces

Putting it All Together

Now that we have all of the information, let's do some example requests to compare shipping rates.

Example #1: Compare Rates of UPS, FedEx, and Stamps.com

This example will get the rates from three carriers: UPS, FedEx, and Stamps.com for one package. Many ShipEngine customers want to see all available carrier rates and then decide which service is best for their shipment. So this scenario will return a long list of rates and services for UPS, FedEx, and Stamps.com. Below the response are smaller parts of what's returned to review things in bitesize pieces.

Request

Copy and paste the following `POST` request into curl and hit **Enter**. Don't forget to put your own API key

```
    "rate_key": "BOOK_RATE_KEY",
    -d '
    {
    "rate_options": {
    "carrier_ids": [
      "se-108562",
      "se-108563",
      "se-108487",
    ]
    },
    "shipment": {
    "validate_address": "no_validation",
    "ship_to": {
      "name": "Jane Doe",
      "phone": "555-555-5555",
      "company_name": "",
      "address_line1": "525 S Winchester Blvd",
      "city_locality": "San Jose",
      "state_province": "CA",
      "postal_code": "95128",
      "country_code": "US"
    },
    "ship_from": {
      "name": "John Doe",
      "phone": "111-111-1111",
      "company_name": "",
      "address_line1": "4009 Marathon Blvd",
      "address_line2": "",
      "city_locality": "Austin",
      "state_province": "TX",
      "postal_code": "78757"
    }
  }
}
```

Response

A successful response will return a lot of information including the rate options:

```
TTP/1.1 200 OK
Date: Mon, 26 Aug 2019 22:35:04 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 37536
Connection: keep-alive
x-shipengine-requestid: dfcaf75f-2451-41ca-8cc5-f501a44a8eaf
```

```
{
  "rate_response": {
    "rates": [
      {
        "rate_id": "se-11274318",
        "rate_type": "shipment",
        "carrier_id": "se-108562",
```

```
"currency": "usd",
"amount": 0.0
},
"other_amount": {
  "currency": "usd",
  "amount": 0.00
},
"zone": null,
"package_type": null,
```

Notice the response contains three different rates. The first one is for UPS:

```
{
  "rate_response": {
    "rates": [
      {
        "rate_id": "se-11274318",
        "rate_type": "shipment",
        "carrier_id": "se-108562",
        "shipping_amount": {
          "currency": "usd",
          "amount": 12.03
        },
        "insurance_amount": {
          "currency": "usd",
          "amount": 0.0
        },
        "confirmation_amount": {
          "currency": "usd",
          "amount": 0.0
        },
        "other_amount": {
          "currency": "usd",
          "amount": 0.00
        },
        "zone": null,
        "package_type": null,
        "delivery_days": 4,
        "guaranteed_service": true,
        "estimated_delivery_date": "2019-08-30T23:00:00Z",
        "carrier_delivery_days": "Friday 8/30 by 11:00 PM",
        "ship_date": "2019-08-26T00:00:00Z",
        "negotiated_rate": false,
        "service_type": "UPS® Ground",

```

```
"shipping_amount": {
  "currency": "usd",
  "amount": 105.78
},
"insurance_amount": {
  "currency": "usd",
  "amount": 0.0
},
"confirmation_amount": {
  "currency": "usd",
  "amount": 0.0
},
"other_amount": {
  "currency": "usd",
  "amount": 7.40
},
"zone": null,
"package_type": null,
"delivery_days": 1,
"guaranteed_service": false,
"estimated_delivery_date": "2019-08-27T08:00:00Z",
"carrier_delivery_days": "Tomorrow by 08:00 AM",
"ship_date": "2019-08-26T00:00:00Z",
"negotiated_rate": false,
"service_type": "FedEx First Overnight®",
"service_code": "fedex_first_overnight",
"trackable": true,
"carrier_code": "fedex",
"carrier_account": "FEDEX HOUSE ACCOUNT"
```

The last one is for Stamps.com (USPS):

```
{
  "rate_id": "se-11274332",
  "rate_type": "shipment",
  "carrier_id": "se-108487",
  "shipping_amount": {
    "currency": "usd",
    "amount": 10.23
  },
  "insurance_amount": {
    "currency": "usd",
    "amount": 0.0
  },
  "confirmation_amount": {
    "currency": "usd",
    "amount": 0.0
  },
  "other_amount": {
```

```
"ship_date": "2019-08-26T00:00:00Z",
"negotiated_rate": false,
"service_type": "USPS Priority Mail",
"service_code": "usps_priority_mail",
"trackable": true,
"carrier_code": "stamps_com",
.....
```

Example #2: Find Rates for One Carrier

This example use case is for users who only want a shipping rate for one specific carrier. We'll look up rates for UPS. Perhaps the user isn't certain if they want to ship it overnight, 2-Day or even UPS Ground. We'll use the following information in the request:

Field	Value
carrier_ids	se-108562
ship_to	Jane Doe 525 S Winchester Blvd San Jose, CA 95128
ship_to phone	555-555-5555
ship_from	John Doe 4009 Marathon Blvd Austin, TX 78756
ship_from phone	111-111-1111
weight	22.0 ounces

Request

Copy and paste the following **POST** request into curl and hit **Enter**. Don't forget to put your own API key in the `_YOUR_API_KEY_` placeholder on line #3.

```
curl -iX POST https://api.shipengine.com/v1/rates \
-H 'Content-Type: application/json' \
-H 'API-Key: _YOUR_API_KEY_' \
-d '{
{
"rate_options": {
"carrier_ids": [
"se-108562",
```

```
"city_locality": "San Jose",
"state_province": "CA",
"postal_code": "95128",
"country_code": "US"
},
"ship_from": {
"name": "John Doe",
"phone": "111-111-1111",
"company_name": "",
"address_line1": "4009 Marathon Blvd",
"address_line2": "",
"city_locality": "Austin",
"state_province": "TX",
"postal_code": "78756",
"country_code": "US"
}
```

Response

A successful response will return all of the UPS shipping rates:

```
HTTP/1.1 200 OK
Date: Tue, 27 Aug 2019 20:05:18 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 11173
Connection: keep-alive
x-shipengine-requestid: 8a7babe9-cea9-4f47-a35f-0634d9332405
```

```
{
"rate_response": {
  "rates": [
    {
      "rate_id": "se-11342583",
      "rate_type": "shipment",
      "carrier_id": "se-108562",
      "shipping_amount": {
        "currency": "usd",
        "amount": 12.03
      },
      "insurance_amount": {
        "currency": "usd",
        "amount": 0.0
      },
      "confirmation_amount": {
        "currency": "usd",
        "amount": 0.0
      },
      "other_amount": {
        "currency": "usd",
```

Ready to solve shipping?

Get a free developer account (no credit card required) to start building with our APIs. Or, contact a shipping expert for more info.

[Chat With an Expert](#)

[Get a Free API Key](#)

PRODUCTS

[Shipping](#)

[Orders](#)

PRICING

[Plans](#)

DEVELOPERS

[Documentation](#)

[API Status](#)

SOLUTIONS

[E-commerce Platforms](#)

[Third Party Logistics](#)

[Brands](#)

INTEGRATIONS

[Carriers](#)

[Marketplaces](#)

COMPANY

[About](#)

[Customers](#)

[Blog](#)

[Contact](#)

[Support](#)

[Privacy Policy](#)

[Terms of Service](#)

Search