
Product Advertising API

Developer Guide

API Version 2013-08-01



Product Advertising API: Developer Guide

Copyright © 2015 Amazon.com and its affiliates. All rights reserved.

Table of Contents

Welcome	1
Overview of Product Advertising API	1
Audience	1
Required Knowledge and Skills	2
How This Guide Is Organized	2
Programming Guide	3
Getting Started	4
Becoming an Associate	4
Becoming a Product Advertising API Developer	5
Reading the Licensing Agreement	7
Using the Product Advertising API Scratchpad	8
E-Commerce and Web Services	9
What is E-commerce?	9
What is a Web Service?	9
What is XML?	9
What is a WSDL?	11
What is a Schema?	18
Product Advertising API Terminology and Basic Concepts	19
The Marketplace	19
What Is the Product Advertising API?	19
How Do I Make Money Using the Product Advertising API?	20
Items for Sale	20
Summary of Product Advertising API Operations	23
Visual Introduction to Product Advertising API	24
Tools to Find Items	25
Tools to Find Out More About an Item	25
Tools to Implement a Shopping Cart	26
Organization of Items on Amazon	26
Browse Nodes	27
Search Indices	33
Variations	35
Accessories	39
Related Items	40
Requests	40
Anatomy of a REST Request	41
Request Limitations	44
REST Syntax	44
SOAP Requests	46
Request Authentication	48
Batch Requests	61
Parameters Common to All Product Advertising API Requests	63
Responses	66
Paging and Sorting Through Responses	70
Find Items to Buy	72
Search Operations	73
Items You Cannot Buy	73
Items Sold by Merchants	75
Common ItemSearch Parameters	76
Power Searches	81
Use Search Bins to Find Items	85
Find Items with Browse Nodes	91
Request and Response Examples	95
Motivate Customers to Buy	127
Promotions	128
Reviews	131

Suggest Similar Items to Buy	133
Top Sellers	140
New Releases	143
Sort by Popularity, Price, or Condition	144
Returning Price and Availability Information	145
Returning Prices	146
Determining Availability	151
Working With Remote Shopping Carts	154
Shopping Cart Concepts	154
Remote Shopping Cart Tasks	158
Add to Cart Form	166
Purchase Items in a Remote Shopping Cart	167
Order Pipeline	167
Implement the Proceed to Checkout Button	168
PurchaseURL	168
Troubleshooting	169
Syntax and Parameter Errors	169
Retrieving Errors	171
Troubleshooting Applications	171
Error Codes and Messages	172
Best Programming Practices	178
Read the Product Advertising API Terms and Conditions	179
Use the Latest API Version	179
Understand Available Operations	179
Use the Right Response Group	179
Use Your Associate Tag in Product Advertising API Requests	180
Handling Errors	180
Use Caches Carefully	180
Use the Correct AWSAccessKeyId	180
Locale Considerations	180
Locales	181
Associate IDs	182
General Differences	182
Shipping Restrictions	182
API Reference	184
Operations	184
ItemSearch	185
BrowseNodeLookup	194
ItemLookup	197
SimilarityLookup	203
CartAdd	207
CartClear	212
CartCreate	214
CartGet	220
CartModify	224
Common Request Parameters	228
Response Groups	230
Accessories	231
AlternateVersions	232
BrowseNodeInfo	234
BrowseNodes	236
Cart	238
CartNewReleases	241
CartTopSellers	242
CartSimilarities	244
EditorialReview	245
Images	247
ItemAttributes	249

ItemIds	253
Large	254
Medium	259
MostGifted	263
MostWishedFor	264
NewReleases	265
OfferFull	267
OfferListings	269
Offers	271
OfferSummary	275
PromotionSummary	277
RelatedItems	279
Request	281
Reviews	283
SalesRank	285
SearchBins	286
Similarities	291
Small	293
TopSellers	294
Tracks	296
Variations	297
VariationImages	300
VariationMatrix	302
VariationOffers	304
VariationSummary	307
Response Elements Common to All Response Groups	309
Response Elements	310
Locale Reference	328
Locale Information for the BR Marketplace	329
Locale Information for the CA Marketplace	331
Locale Information for the CN Marketplace	345
Locale Information for the DE Marketplace	358
Locale Information for the ES Marketplace	378
Locale Information for the FR Marketplace	394
Locale Information for the IN Marketplace	410
Locale Information for the IT Marketplace	423
Locale Information for the JP Marketplace	437
Locale Information for the MX Marketplace	455
Locale Information for the UK Marketplace	462
Locale Information for the US Marketplace	480
ItemSearch Sort Values	500
Sort Values	500
Resources	504
Document History	506
Index	508

Welcome

Topics

- [Overview of Product Advertising API \(p. 1\)](#)
- [Audience \(p. 1\)](#)
- [How This Guide Is Organized \(p. 2\)](#)

Overview of Product Advertising API

Amazon has developed a world-class web service that millions of customers use every day. As a developer, you can build Product Advertising API applications that leverage this robust, scalable, and reliable technology. You get access to a lot of the data used by Amazon including the items for sale, customer reviews, seller reviews, as well as most of the functionality you see on Amazon.com, such as finding items, displaying customer reviews, and product promotions. Product Advertising API operations open the doors to Amazon's databases so that you can take advantage of Amazon's sophisticated e-commerce data and functionality. Build your own web store to sell Amazon items or your own items.

Best of all, Product Advertising API is free. By signing up to become a Product Advertising API developer, you join the tens of thousands of developers who are already realizing financial gains by creating Product Advertising API-driven applications and web stores.

How Product Advertising API works

1. Your application uses the Product Advertising API to display item descriptions, images, and reviews to your customers.
2. Customers shop and add items to their shopping cart.
3. When the customer is ready to purchase the items, your application sends an HTML form to Product Advertising API and Amazon completes the purchase by getting purchase information, such as payment method and shipping address. Amazon then fulfills the order by shipping the items.

Audience

This guide is intended for developers who want to build an e-commerce storefront that sells items listed on Amazon.com, or an application that helps others build e-commerce storefronts.

Required Knowledge and Skills

Use of this guide assumes you are familiar with the following:

- XML (For an overview, see [W3 Schools XML Tutorial](#).)
- Basic understanding of web services (For an overview, see [W3 Schools Web Services Tutorial](#).)

If you are an Amazon Associate and are looking general information, see [Amazon Associates Tools for Every Site](#).

How This Guide Is Organized

Information	Topics
Learn the basics about the Product Advertising API	Terminology and Basic Concepts (p. 19)
See the Programming Reference, which provides task-oriented descriptions of how to use and implement Product Advertising API operations.	Programming Reference (p. 3)
See the API Reference to learn more about Product Advertising API operations, response groups, locales, and parameter constraints.	API Reference (p. 184)
See the latest documentation changes.	Document History (p. 506)

Programming Guide

The Programming Guide provides task-oriented descriptions of how to use and implement Product Advertising API operations. For a complete description of these operations, refer to the [API Reference \(p. 184\)](#).

The following table describes the sections in the programming guide.

Section	Description
Getting Started (p. 4)	Describes the tasks you must complete before using Product Advertising API.
E-Commerce and Web Services (p. 9)	Introduces technologies central to Product Advertising API, such as e-commerce, WSDL, and schemas.
Product Advertising API Terminology and Basic Concepts (p. 19)	Introduces terms and concepts central to Product Advertising API.
Visual Introduction to Product Advertising API (p. 24)	Introduces Product Advertising API functionality by visually comparing it to functionality seen on Amazon's retail web site.
Organization of Items for Sale on Amazon (p. 26)	Describes how items are organized in Amazon databases.
Requests (p. 40)	Introduces you to making Product Advertising API requests and processing Product Advertising API responses.
Finding Items to Buy (p. 72)	Describes the operations and response groups required to find items to buy.
Motivating Customers to Buy (p. 127)	Describes the operations and response groups required to motivate customers to buy. Topics include displaying images, customer review, and seller reviews.
Returning Price And Availability Information (p. 145)	Describes the operations and response groups required to get pricing and availability information of the items for sale.
Working With Remote Shopping Carts (p. 154)	Describes the operations and response groups required to create a shopping cart, add items to it, and modify the number of items in the cart.

Section	Description
Purchasing the Items in a Remote Shopping Cart (p. 167)	Describes the operations and response groups required to purchase the items in a shopping cart.
Handling Errors and Troubleshooting Applications (p. 169)	Describes errors returned by Product Advertising API.
Best Programming Practices (p. 178)	Provides programming tips to maximize your Product Advertising API application's functionality.
Locale Considerations (p. 180)	Addresses locale-specific issues.

Getting Started

Topics

- [Becoming an Associate \(p. 4\)](#)
- [Becoming a Product Advertising API Developer \(p. 5\)](#)
- [Reading the Licensing Agreement \(p. 7\)](#)
- [Using the Product Advertising API Scratchpad \(p. 8\)](#)

You need to join the Amazon Associates program and then register to become a Product Advertising API developer.

Becoming an Associate

Associates earn commissions by using their own websites to refer sales to Amazon.com. To get a commission, an Associate must have an Associate ID, also known as an Associate tag. The Associate ID is an automatically generated unique identifier that you will need to make requests through the Product Advertising API.

To become an associate

1. Using the table below, click the Amazon Associates URL for your locale.
2. Follow the instructions to create an Amazon Associates account.
3. When you register as an Amazon Associate, an Associate ID is sent you in email. When you sign in Amazon Associates for your locale, the home page displays a message that says: **Signed in as YourAssociateID**.



Note

One of the requirements for becoming an Associate is that you provide the URL of your site. If your site is not yet public but you want to test against the API, you must still provide a URL during registration.

Your Associate ID works *only* in the locale in which you register. If you want to be an Associate in more than one locale, you must register separately for each locale.

Locale	URL
Brazil	https://associados.amazon.com.br/
Canada	https://associates.amazon.ca/
China	https://associates.amazon.cn/gp/advertising/api/detail/main.html
France	https://partenaires.amazon.fr/
Germany	http://partner.net.amazon.de
India	https://affiliate-program.amazon.in/
Italy	https://programma-affiliazione.amazon.it/gp/advertising/api/detail/main.html
Japan	https://affiliate.amazon.co.jp/
Mexico	https://afiliados.amazon.com.mx/gp/associates/join/landing/main.html
Spain	https://afiliados.amazon.es
United Kingdom	https://affiliate-program.amazon.co.uk
United States	http://affiliate-program.amazon.com/

Becoming a Product Advertising API Developer

The Product Advertising API allows developers to advertise products from the following Amazon sites:

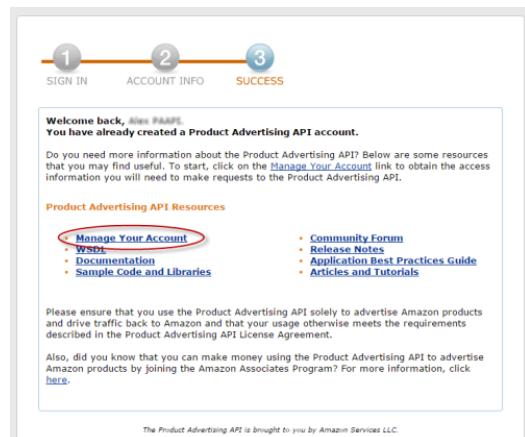
Locale	URL
Brazil	http://www.amazon.com.br
Canada	http://www.amazon.ca
China	http://www.amazon.cn
France	http://www.amazon.fr
Germany	http://www.amazon.de
India	http://www.amazon.in
Italy	http://www.amazon.it
Japan	http://www.amazon.co.jp
Mexico	http://www.amazon.com.mx
Spain	http://www.amazon.es
United Kingdom	http://www.amazon.co.uk
United States	http://www.amazon.com

After you register for Amazon Associates, register to become a Product Advertising API developer.

1. On the Amazon Associates page, click the **Product Advertising API** link. Use the same email address that you used for your Associates account. You can also use the table at the bottom of this page to locate the Product Advertising API URL for your locale.



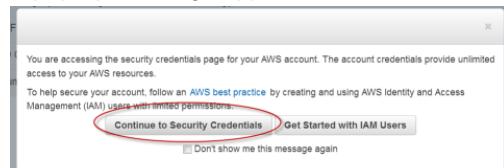
2. Follow the instructions to register as a Product Advertising API developer.
3. On the Success page, click **Manage Your Account**. We automatically create an AWS account for you. You will use your AWS account security credentials to make calls to the Product Advertising API.



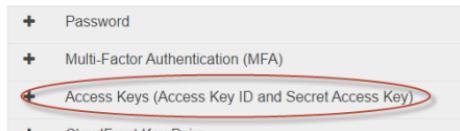
4. On the **Manage Your Account** page, click the **AWS Security Credentials Console** link and sign in to your AWS account using the same email address and password.



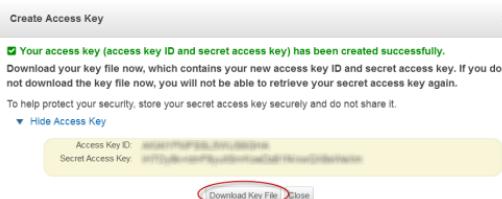
5. A pop-up message appears. Click **Continue to Security Credentials**.



6. Click **Access Keys (Access Key ID and Secret Key)**.



7. Click **Create New Access Key**, and then click **Show Access Key or Download Key File** to retrieve the credentials.



8. Save the access key information in a safe location. You will use these credentials to make calls to the Product Advertising API.

Important

You can access the secret access key *only* when you first create an access key pair. For security reasons, it cannot be retrieved at any later time. Ensure that you save both the access key ID and its matching secret key. If you lose them, you must create a new access key pair. IAM roles are not currently supported. You must use the root account credentials.

Locale	Product Advertising API URL
Brazil	http://associados.amazon.com.br/gp/associates/apply/main.html
Canada	https://associates.amazon.ca/gp/flex/advertising/api/sign-in.html
China	https://associates.amazon.cn/gp/advertising/api/detail/main.html
France	https://partenaires.amazon.fr/gp/flex/advertising/api/sign-in.html
Germany	https://partner.net.amazon.de/gp/flex/advertising/api/sign-in.html
India	http://affiliate-program.amazon.in/gp/associates/apply/main.html
Italy	https://programma-affiliazione.amazon.it/gp/advertising/api/detail/main.html
Japan	https://affiliate-program.amazon.com/gp/flex/advertising/api/sign-in-jp.html
Mexico	https://afiliados.amazon.com.mx/gp/advertising/api/detail/main.html
Spain	https://afiliados.amazon.es/gp/flex/advertising/api/sign-in.html
United Kingdom	https://affiliate-program.amazon.co.uk/gp/flex/advertising/api/sign-in.html
United States	https://affiliate-program.amazon.com/gp/flex/advertising/api/sign-in.html

Reading the Licensing Agreement

You will need to review and accept the terms and conditions of the license agreement to become a Product Advertising API developer. To read the Product Advertising API licensing agreement, go to the license agreement link for your locale:

Locale	License Agreement URL
Brazil	https://associados.amazon.com.br/gp/associates/agreement/
Canada	https://associates.amazon.ca/gp/advertising/api/detail/agreement.html
China	https://associates.amazon.cn/gp/advertising/api/detail/agreement.html
France	http://partenaires.amazon.fr/gp/advertising/api/detail/agreement.html
Germany	https://partner.net.amazon.de/gp/advertising/api/detail/agreement.html
India	http://affiliate-program.amazon.in/gp/advertising/api/detail/agreement.html
Italy	https://programma-affiliazione.amazon.it/gp/advertising/api/detail/agreement.html
Japan	https://affiliate.amazon.co.jp/gp/advertising/api/detail/agreement.html
Mexico	https://afiliados.amazon.com.mx/gp/advertising/api/detail/agreement.html
Spain	https://afiliados.amazon.es/gp/advertising/api/detail/agreement.html
United Kingdom	https://affiliate-program.amazon.co.uk/gp/advertising/api/detail/agreement.html
United States	https://affiliate-program.amazon.com/gp/advertising/api/detail/agreement.html

Note

If you plan to use the Product Advertising API to advertise Amazon products from a locale other than the one where you registered, be sure to review the license agreement for that locale. The terms and conditions for each locale apply to any use of the Product Advertising API in that locale.

Review the [Application Best Practices Guide](#) to make sure your application is compliant, scalable, and efficient.

Important

In addition to the Product Advertising API License Agreement, be sure to read your locale's Associates Program Operating Agreement for information about usage guidelines, policies, and requirements.

Using the Product Advertising API Scratchpad

Use the Scratchpad to easily submit requests to the Product Advertising API. The Scratchpad tool returns sample code to help you understand how the Product Advertising API works.

To use the Scratchpad tool

1. Go to [Product Advertising API Scratchpad](#).
2. Choose an operation, for example **ItemSearch**.
3. Under **Common parameters**, choose your marketplace and type your credentials that you retrieved from the previous procedures.
4. Choose your request parameters, and click **Run request**.
5. Under **Request URL**, your unsigned and request URL appear.
6. Under **Response**, locate your response examples.

Tip

You can also download the Scratchpad user guide for your reference.

E-Commerce and Web Services

This chapter is an introduction to e-commerce and web services. If you are familiar with these topics, proceed to the next chapter.

What is E-commerce?

E-commerce is the practice of using computers to buy, sell, and market goods and services across a network, such as the Internet.

In Product Advertising API, e-commerce is facilitated by web services.

What is a Web Service?

This guide assumes that you are familiar with the notion of using your computer (the client) to request that some other computer (the server's) perform some task and respond over the Internet. The work a server does is called a service. The service might be returning a weather forecast, or, in the case of Product Advertising API, returning information about items for sale on Amazon. Some of these request and response interactions over the Internet are considered web services provided by a web server.

A web service is any piece of software that uses REST and/or a standardized XML messaging system, described by a WSDL (Web Service Definition Language) (pronounced "wiz-dal"), to exchange data between applications or systems. A web service must have some simple mechanism for interested parties to locate the service and its public interface using standard network protocols, such as, but not limited to, SOAP over HTTP. Software applications written in various programming languages and running on various platforms can use web services transparently to exchange data over computer networks, like the Internet, because the WSDL serves as the definition of the language used by the computers.

In Product Advertising API:

- Requests and responses occur across the Internet.
- Client and server use REST, SOAP, or XML as the means of communication.
- Client and server agree on the grammar and syntax used in the requests and responses by specifying a WSDL.
- Requests and responses are not tied to a single operating system or programming language.

A simple example of a Product Advertising API request uses the `ItemSearch` operation.

1. You submit a request with the `ItemSearch` operation. The operation asks Amazon's web servers to find descriptions of an item in its databases.
2. The web servers carry out their service by finding the item descriptions.
3. The web servers send the item data back to the requester.

All Product Advertising API requests are based on REST or SOAP and all of the responses are based on XML.

Note

For more information about REST, see [Anatomy of a REST Request \(p. 41\)](#).

What is XML?

XML (Extensible Markup Language) is a standard defined by the World Wide Web Consortium (W3C). XML uses tags (identifiers enclosed in brackets, for example, `<para>`) to label content in text documents.

These tags, collectively called the "markup," can be read by humans and computers. XML tags, like SGML tags, encode not only the meaning of the content but also its structure.

XML looks like HTML but XML has nothing to do with the display of the content, which is the central focus of HTML. The following XML tags label the content they enclose as a paragraph.

```
<para>This is a paragraph.</para>
```

The `<para>` tag has nothing to do with the display of the sentence. The tag, for example, does not take attributes such as font, size, or style.

Unlike HTML, there is not a single set of tags used in all XML documents. XML enables designers to create their own set of tags that are appropriate to their business. For example, Product Advertising API encodes its APIs using an XML document called a WSDL. Other companies create their own XML tags to define their APIs. The tags in these two WSDLs most likely would be completely different. The syntax and rules by which the XML elements are defined in the WSDL, however, are the same. Because a web server handles multiple requests that implement multiple WSDLs, each request must specify the WSDL it is using..

XML Syntax

XML tools enforce XML syntax. XML syntax is very similar to HTML syntax, except, like XHTML, the syntax is strictly enforced. The syntax is:

- Tags are enclosed within angle brackets, for example, `<para>`
- Opening tags must be paired with closing tags, for example, `<para>Sentence</para>`
- Opening and closing tags must be nested correctly, for example, the following example is well formed

```
<note><para>Sentence</para></note>
```

The following example is not well formed

```
<note><para>Sentence</note></para>
```

If these syntax rules are followed, the XML document is said to be well formed. That does not mean, however, that the tags in the XML document are necessarily the ones defined in the associated WSDL. If, for example, the WSDL defines `<para>`, an XML document that uses `<Para>` would cause an error because "Para" is not defined in the WSDL. When an XML document is well formed and the tags it uses conform to tags defined in the WSDL, the document is said to be valid. Product Advertising API responses always contain valid XML.

XML Structures

Some data in XML documents are one dimensional, for example, `<para>Sentence</para>`. Some data, however, is structured, as shown in the following XML snippet.

```
<SubTotal>
  <Amount>2998</Amount>
  <CurrencyCode>USD</CurrencyCode>
  <FormattedPrice>$29 . 98</FormattedPrice>
</SubTotal>
```

This example shows that the subtotal is comprised of three pieces of data: *Amount*, *CurrencyCode*, and *FormattedPrice*. The example also shows that there can be a hierarchy of tags. In this example, *Amount* is a child tag of *Subtotal*. These structures represent arrays of values.

Using XML

So, what do you do with the XML response? The good news is that the Product Advertising API returns a well-defined set of tags for each kind of request. That means that you can use parsing mechanisms to pull out of the XML responses the data that you would like to submit in a second request, or, display on a web page. For instance, in the preceding example, you might parse the XML document to find *FormattedPrice* so that you could display the \$29.98 on a web page that shows an item for sale.

XML is not only used to create requests and responses, it is also used to create a WSDL, which defines the XML tags.

What is a WSDL?

Topics

- [Restricted Parts of the WSDL \(p. 11\)](#)
- [Anatomy of a WSDL \(p. 12\)](#)
- [WSDL Location \(p. 18\)](#)

The goal of this section is to give you enough information so that you can read and use the Product Advertising API WSDL. You typically read a WSDL to understand value types, operation definitions, and request and response formats.

A WSDL (Web Service Description Language) is an XML document that defines the operations, parameters, requests, and responses used in web service interactions. You can think of a WSDL as the contract that defines the language and grammar used by web service clients and servers. When you look at the Product Advertising API WSDL, for example, you find in it all of the Product Advertising API operation names, parameters, request and response structures.

There is not a single WSDL. Product Advertising API, for example, has many different versions of its WSDL—the latest one and all of its previous versions. Not only can one company use different versions of a WSDL, every company can use its own WSDL based on its own APIs or business metrics. For that reason, web service requests must identify the WSDL they use so the web servers know how to interpret the requests.

In practice, Product Advertising API developers use the same version of the Product Advertising API WSDL in every request in a session. Over time, they might send requests to the latest Product Advertising API WSDL. The Product Advertising API WSDL is upgraded regularly.

Restricted Parts of the WSDL

The WSDL defines all Product Advertising API operation requests and responses. The majority of the WSDL is generic. There are, however, some small portions of the WSDL that are partner specific. That is, the use of some Product Advertising API operations is restricted to specific Amazon partners. Those operations are marked as restricted by a notation similar to the following.

```
<xss:element name="RestrictedOperation" minOccurs="0" maxOccurs="unbounded">
  <xss:annotation>
    <xss:appinfo>
      <aws-se:restricted>
        <aws-se:excludeFrom>public</aws-se:excludeFrom>
        <aws-se:excludeFrom>partner</aws-se:excludeFrom>
```

```
</aws-se:restricted>  
</xs:appinfo>
```

If you try to use a restricted operation and you are not the Amazon partner, Product Advertising API returns an error message.

Anatomy of a WSDL

Topics

- [Definitions \(p. 12\)](#)
- [Request Definitions \(p. 14\)](#)
- [Response Definitions \(p. 15\)](#)
- [Binding \(p. 16\)](#)
- [Service \(p. 17\)](#)

Typically, the Binding and Service segments do not change from one WSDL release to the next. In the Definitions segment, only the WSDL version changes. For that reason, when you read a WSDL, you will typically spend most of your time reading the Request and Response Definitions segments.

Definitions

Topics

- [Namespaces \(p. 13\)](#)
- [Versioning \(p. 13\)](#)

The Definitions section of the WSDL defines the namespaces used throughout the WSDL, and the name of the service, as shown in the following snippet of the Product Advertising API WSDL.

```
<?xml version="1.0" encoding="UTF-8" ?>  
<definitions  
    xmlns="http://schemas.xmlsoap.org/wsdl/"  
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"  
    xmlns:xs="http://www.w3.org/2001/XMLSchema"  
    xmlns:tns="http://webservices.amazon.com/  
    AWSECommerceService/2013-08-01"  
    targetNamespace="http://webservices.amazon.com/  
    AWSECommerceService/2013-08-01">
```

This example shows that the:

- Default namespace is `xmlns="http://schemas.xmlsoap.org/wsdl/"`
- SOAP namespace used is `xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"`
- Schema used is `xmlns:xs="http://www.w3.org/2001/XMLSchema"`
- Product Advertising API WSDL namespace is
`"http://webservices.amazon.com/AWSECommerceService/2013-08-01"`
The date at the end is the version number. It is the date the WSDL became public.
- TargetNamespace is `"http://webservices.amazon.com/AWSECommerceService/2013-08-01"`
The TargetNamespace is an XML schema convention that enables the WSDL to refer to itself (as the target). The TargetNamespace value is the Product Advertising API WSDL namespace

Namespaces

Namespaces are collections of parameters and operations in which their names are unique. The advantage of using namespaces is that the WSDL can define terms, like string, just by referring its namespace, xs. Also, prepending the namespace to a parameter ensures that there is no danger of name collisions.

Each namespace declaration starts with "xmlns:" (XML namespace:) and is followed by the abbreviation for the namespace. For example, in the following namespace declaration, xs becomes the abbreviation for the URL of the schema.

```
xmlns:xs="http://www.w3.org/2001/XMLSchema"
```

Throughout the remainder of the WSDL you will see parameters defined in terms of namespace abbreviations, for example:

```
type="xs:string"  
ref="tns:HTTPHeaders"
```

These abbreviations provide the namespace in which the parameters are defined.

Versioning

Product Advertising API enables you to specify the version of the WSDL you want to use. This functionality ensures that future enhancements and changes to Product Advertising API WSDLs will not be intrusive to your applications. For example, when Product Advertising API adds new elements to its WSDL, applications that validate against an older Product Advertising API WSDLs will not be affected.

Product Advertising API WSDL version names are based on the date that they become active. The version of the WSDL is specified in the Product Advertising API WSDL namespace declaration. In the preceding example, the version of the WSDL is 2013-08-01.

```
xmlns:tns="http://webservices.amazon.com/  
AWSECommerceService/2013-08-01"
```

In reality, the date, here, is the WSDL's file name.

The `AWSECommerceService` directory contains all of the Product Advertising API WSDL versions. You use the `Version` parameter in REST requests to specify the version of the WSDL you want to use. The default version is 2011-08-01. If you want to use a different WSDL version, including the latest, you must specify it in each request, for example, in REST.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
&Operation=ItemSearch&  
SearchIndex=Books&  
Author=Steve%20Davenport&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Note

This version of the *Product Advertising API Developer Guide* describes all of the functionality of the WSDL as of the guide's publication date.

SOAP requests always specify a namespace, which includes the WSDL version. To avoid problems due to future WSDL changes, be sure to specify a WSDL version in your SOAP application.

Request Definitions

The Request Definitions segment of the WSDL defines Product Advertising API operation requests, as shown in the following WSDL snippet.

```
<xs:complexType name="ItemSearchRequest">
  <xs:sequence>
    <xs:element name="Actor" type="xs:string" minOccurs="0" />
    <xs:element name="Artist" type="xs:string" minOccurs="0" />
    <xs:element name="Availability" minOccurs="0">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="Available" />
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element ref="tns:AudienceRating" minOccurs="0" maxOccurs="unbounded" />
```

This snippet shows some of `ItemSearch`'s input parameters, including `Actor`, `Artist`, `Availability`, and `AudienceRating`. The element declarations specify that these parameters are valid in an `ItemSearch` request. Most of the parameters in this example are strings. The type of one, however, `Availability`, is a variation on the base class, string. In this case, the variation puts a restriction on the strings that can be valid values for `Availability`. For that reason, the restriction keyword is used. The restriction is that the valid values for `Availability` are defined by an enumeration. The enumeration, however, has only one valid value, "Available," which means that the parameter, `Availability` can be set to only one value.

`minOccurs` refers to the minimum number of times the parameter must appear in an `ItemSearch` request. If the value is zero, the associated parameter is optional. If the value is 1, the associated parameter is required to be included once in every request involving that operation. The default value is 1, that is, if `minOccurs` is not included in an element declaration, `minOccurs` is 1.

`maxOccurs` defines the maximum number of times the parameter can appear in a request. The default is 1, that is, if `maxOccurs` is not included in an element declaration, `maxOccurs` is 1 and the parameter can only appear once in a request. In the preceding example, `maxOccurs` is "unbounded," which means that the `AudienceRating` parameter can appear any number of times in an `ItemSearch` request.

In the preceding example, the parameter types are declared to be simpleTypes. A simple type cannot have child elements or attributes. Complex types can. In practice, any parameter that can take multiple values, such as an array, must be defined as a complex type.

The following snippet shows an example of a complex type.

```
<xs:element name="ItemSearch">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="MarketplaceDomain" type="xs:string" minOccurs="0" />
      <xs:element name="AWSAccessKeyId" type="xs:string" minOccurs="0" />
      <xs:element name="SubscriptionId" type="xs:string" minOccurs="0" />
```

This definition snippet shows three of the parameters that can be part of an `ItemSearch` request.

Response Definitions

The response section defines the responses returned by default by each operation. The following snippet shows some of the specifications of an `ItemSearch` response.

```
<xs:element name="ItemSearchResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="tns:OperationRequest" minOccurs="0" />
      <xs:element ref="tns:Items" minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

The response section shows that an `ItemSearch` response contains two optional (`minOccurs=0`) elements, `OperationRequest` and `Items`. Both of these elements are references (`ref=`), which means that they are defined further down in the WSDL.

Further down in the WSDL, `OperationRequest` is defined, as follows.

```
<xs:element name="OperationRequest">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="tns:HTTPHeaders" minOccurs="0" />
      <xs:element name="RequestId" type="xs:string" minOccurs="0" />
      <xs:element ref="tns:Arguments" minOccurs="0" />
      <xs:element ref="tns:Errors" minOccurs="0" />
      <xs:element name="RequestProcessingTime" type="xs:float" minOccurs="0" maxOccurs="1" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

This definition also contains several references. One is `Arguments`, which is defined further down in the WSDL. To fully understand the definition of the parts of a request, you keep digging down through the layers of refs. You know that you have reached the end of the definition hierarchy when you no longer have "ref" in the element's definition. Instead, the element definition will have a "name," the name of the element, and "type," which specifies the element's type. The type will be a base type, such as, `string`, which is defined in the schema (`xs:`), as shown.

```
<xs:element name="RequestId" type="xs:string" minOccurs="0" />
```

This line defines `RequestId` to be of type `string`, which is defined by the W3C schema.

When you look at a sample response, shown in the following example, you can see how the definition of `RequestId` is carried out.

```
<ItemSearchResponse xmlns="http://webservices.amazon.com/AWSECommerceService/2013-08-01">
  ...
  <OperationRequest>
    ...
    <RequestId>0VFY0HFBRTJGRE6KES74</RequestId>
```

First, you see that the value for *RequestId* is string. Secondly, the name of the element is *RequestId*. Third, you can see, in the XML hierarchy, how the definition of *RequestId* is nested inside the *OperationRequest* element, which is nested inside of *ItemSearchResponse*. Remember, it was the "ref" keyword that created the nesting in the WSDL.

Response Group Definitions

Response groups, except in the case of a request error, always form part of a response. Each response group is defined in the WSDL. The following snippet from the WSDL shows the definition of the response group, [Images \(p. 247\)](#).

```
<xs:complexType name="Image">
<xs:sequence>
<xs:element name="URL" type="xs:string" />
<xs:element name="Height" type="tns:DecimalWithUnits" />
<xs:element name="Width" type="tns:DecimalWithUnits" />
<xs:element name="IsVerified" type="xs:string" minOccurs="0" />
</xs:sequence>
</xs:complexType>
```

As you can see, the *Image* response group returns the elements *URL*, *Height* (height of the image), *Width*, and *IsVerified*. All are required in the response except *IsVerified*, which is optional (*minOccurs=0*). You can see how these elements are displayed in a response.

```
<MediumImage>
<URL>http://ecl.images-amazon.com/images/P/
B00005TNFV.01._SCMZZZZZZ_.jpg</URL>
<Height Units="pixels">140</Height>
<Width Units="pixels">99</Width>
```

This example shows how *URL*, *Height*, and *Width* are child elements.

PortType

The association between operation names and their request and response definitions is created by the *PortType* element in the WSDL, for example:

```
<portType name="AWSECommerceServicePortType">
...
<operation name="ItemSearch">
<input message="tns:ItemSearchRequestMsg" />
<output message="tns:ItemSearchResponseMsg" />
</operation>
```

In this example, the operation, *ItemSearch*, is associated with its request and response definitions, *ItemSearchRequestMsg* and *ItemSearchResponseMsg*. The keywords, *input* and *output*, identify the operation's request and response definitions, respectively.

Binding

The binding segment of the WSDL specifies how operation requests and responses, defined in *PortType*, are actually transmitted over the wire using underlying transport protocols. While this is an interesting portion of the WSDL, it is a section that rarely changes so you need not pay much attention to it.

Binding values include HTTP GET, HTTP POST, and SOAP. SOAP is not tied to a specific transport. SMTP, FTP, HTTP are just some of the options that can transport a SOAP request. HTTP, however, is

most commonly used. While both HTTP GET and HTTP POST are allowed, HTTP POST is preferred because many servers place character limits on HTTP GET requests.

Product Advertising API uses SOAP, as shown in the following Product Advertising API WSDL snippet.

```
<binding name="AWSECommerceServiceBinding"
  type="tns:AWSECommerceServicePortType">
  ...
  <operation name="ItemSearch">
    <soap:operation soapAction="http://soap.amazon.com" />
    <input>
      <soap:body use="literal" />
    </input>
    <output>
      <soap:body use="literal" />
    </output>
  </operation>
```

This binding shows that Product Advertising API uses two SOAP extensions: soap:operation and soap:body.

The soap:operation element specifies that the Product Advertising API operation, ItemSearch , in this case, is bound to a specific SOAP implementation. The soapAction attribute specifies that the SOAPAction HTTP header is used to identify the Product Advertising API service, which is the URI value of soapAction, <http://soap.amazon.com>.. soapAction enables Amazon web servers to determine the intent of the SOAP request without having to examine the message portion of the SOAP payload. Specifying this URI is required to access Product Advertising API web servers.

The soap:body element specifies the input and output details. The value in the Product Advertising API WSDL is "literal," which means that instead of encoding the input and output as a SOAP struct, a literal XML document is used. You have seen that Product Advertising API responses are XML documents.

Service

The Service segment of the WSDL specifies the web service used, which, in this case, is Product Advertising API, as shown in the following WSDL snippet:

```
<service name="AWSECommerceService">
  <port name="AWSECommerceServicePort"
    binding="tns:AWSECommerceServiceBinding">
    <soap:address location="http://soap.amazon.com/onca/soap?Service=
AWSECommerceService" />
  </port>
</service>
```

This information changes very rarely and so you need not pay much attention to it.

Every Product Advertising API request includes this service declaration, as shown in the following example.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
Operation=ItemSearch&
SearchIndex=Books&
Keywords=Saving%20Miss%20Oliver's
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

WSDL Location

Product Advertising API publishes its API through a WSDL. The default WSDL is located at <http://webservices.amazon.com/AWSECommerceService/AWSECommerceService.wsdl>. Note that the default WSDL and the latest WSDL may not be the same version. You can access a particular WSDL version by adding the version date in the URL, as in the following example:

```
http://webservices.amazon.com/AWSECommerceService/[VERSION]/AWSECommerceService.wsdl
```

For example, the 2013-08-01 WSDL can be found at <http://webservices.amazon.com/AWSECommerceService/2013-08-01/AWSECommerceService.wsdl>

The WSDL contains all the API endpoints. To select the required endpoint, see your SOAP framework.

What is a Schema?

A schema is similar to a WSDL in that both are XML documents. Whereas the WSDL defines the web service language used by computers to converse, the schema defines the data types used in the WSDL.

You do not have to create schemas to use Product Advertising API. Those have already been created. It is helpful, however, to understand schemas so that you can determine the data types returned in responses.

The W3C defines the base data types, which include, for example, int, string, and float. While these data types are useful, they are not very descriptive. For example, defining every occurrence of text in an XML document as being of type string hides the differences between text that is, for example, a paragraph and a note. In such an application where paragraphs and notes are used, a schema would contain an extension of the string base class so that paragraph (<para>) and note (<note>) could be used as tags in XML documents.

Schemas enable you to create your own data types for the purpose of identifying the content in an XML document. All data types that you create must be based on the base data types defined by the W3C. This is the schema namespace defined in the WSDL example.

```
xmlns:xs="http://www.w3.org/2001/XMLSchema"
```

The data types that can be created are either simple or complex. Complex types can have sub elements and attributes; simple types cannot.

In the WSDL section of this chapter, you saw that complex types are declared as complexType. In the following example, the element, SearchBinSet, is defined as having two child elements, Bin and NarrowBy.

```
<xs:element name="SearchBinSet">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="tns:Bin" minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="NarrowBy" type="xs:string" use="required" />
  </xs:complexType>
</xs:element>
```

The NarrowBy attribute is defined in terms of a base type, string. The Bin parameter, however, is not. That means that Bin is defined elsewhere in the schema. Schema syntax, like WSDL syntax, calls for using the keyword "ref" if the element is defined elsewhere in a schema.

The next step in understanding Bin would be to see its definition in the schema. It might be that all of Bin's sub elements are defined by base types. In that case, the research would be over; you would have the full definition of SearchBinSet and its child elements. If, however, Bin contains more "ref's," you would repeat the search for the child elements until you reached element type definitions that used base types, as shown in the following example.

```
name="BinItemCount" type="xs:string"
```

Schema Location

Product Advertising API provides schemas for validating the XML in SOAP requests and for specifying item attribute types in responses. You can find the latest Product Advertising API XML schema at <http://webservices.amazon.com/AWSECommerceService/AWSECommerceService.xsd>.

Product Advertising API Terminology and Basic Concepts

Topics

- [The Marketplace \(p. 19\)](#)
- [What Is the Product Advertising API? \(p. 19\)](#)
- [How Do I Make Money Using the Product Advertising API? \(p. 20\)](#)
- [Items for Sale \(p. 20\)](#)
- [Summary of Product Advertising API Operations \(p. 23\)](#)

This chapter introduces you to Product Advertising API terminology and concepts. The concepts are briefly presented here, so that you have a broad understanding of Product Advertising API.

The Marketplace

Sellers come together to sell their items on Amazon.com, in what is called the marketplace. Individual sellers can list their items for sale on Amazon.com.

What Is the Product Advertising API?

The Product Advertising API gives you much of the functionality that you see in action on Amazon's retail web site, Amazon.com. That functionality includes:

- **Find items to buy.**

These items are for sale by Amazon or other sellers.

- **Find information about those items.**

Item with an item, the author of a book, or the composer of an album.

- **Get customer reviews of items.**

Show customers how others reviewed the items.

- **Create a fully-functional shopping cart.**

Add items that are immediately available or ones that will become available, such as in a pre-sale of a book.

- **Add, remove, and modify the items in the shopping cart.**

Have full control over the contents of a shopping cart.

- **Get information about the company selling the item.**

Show customers what others think about the merchant selling the item.

- **Find similar items for sale.**

Generate additional sales by suggesting similar items that customers are buying.

- **Purchase the items in the shopping cart.**

When the customer buys an item, Amazon handles the shipping, payment, and order fulfillment, or notifies the seller.

How Do I Make Money Using the Product Advertising API?

Use the Product Advertising API to advertise Amazon products with Amazon Associates and earn referral fees when customers buy qualifying products.

You need to register for an Amazon Associates account and ensure that you include your Associate Tag in all requests. Product Advertising API returns URLs that have your Associate Tag. When a customer visits an Amazon marketplace through a tagged link and buys a product, you earn referral fees. For more information, see [Becoming an Associate \(p. 4\)](#).

Items for Sale

Amazon.com lists millions of items for sale. Most of these items are warehoused and sold by Amazon or other large merchants. All of these items are part of the Amazon marketplace.

In addition to large merchants selling items, individuals and companies also sell items that can be found using Amazon.com. Most sellers list their items on the Amazon marketplace.

Product Advertising API operations only enable you to retrieve items that are stored by Amazon. Product Advertising API does not enable you to add items for sale to Amazon. Sellers, such as Amazon Advantage sellers, who do add items to Amazon's catalog do so under a separate agreement. For more information, see [Amazon Marketplace Web Services \(Amazon MWS\)](#).

Item Identifiers

All items for sale on Amazon.com have identifiers. There are two major kinds. The first identifies items regardless of whether or not they can be purchased:

- **ASIN — Amazon Standard Item Number**

An alphanumeric token that uniquely identifies items in the Amazon marketplace.

All items in the marketplace have an ASIN. This is by far the most common identifier. ASINs are used by the majority of Product Advertising API operations.

The second identifies items that can actually be purchased:

- **OfferListingId —** An alphanumeric token that uniquely identifies items in the Amazon marketplace that can be purchased

All items in the marketplace that can be purchased have an OfferListingId. This is by far the most common identifier for items that can be purchased.

Other Item Identifiers

Amazon has other item identifiers but these are rarer and they can be used in fewer Product Advertising API operations:

- **UPC** — Universal Product Code

A 12-digit item identifier used in the US and CA locales. The UPC is the identifier used in bar codes.

- **EAN** — European Article Number

A 13-digit equivalent of the UPC that is used in Europe for products and bar codes.

- **JAN** — Japanese Article Number

The equivalent of the EAN that is used in Japan for products and bar codes.

- **ISBN** — International Standard Book Number

An alphanumeric token that uniquely identifies a book. To use ISBN-13, use the EAN IdType.

- **SKU** — Stock Keeping Unit

A merchant-specific identifier for a purchasable good, like a shirt or chair. Amazon's version of the SKU is the ASIN.

Amazon assigns items with SKUs an ASIN. If you had to search by SKU only, you would have to search each merchant's items independently because SKUs are not unique across all merchants. Several Product Advertising API operations enable you to search by SKU but the merchant must also be identified in the same request.

- **EISBN** — Electronic International Standard Book Number

A token that uniquely identifies a digital book.

All items in the Amazon marketplace have an ASIN even if they also have one of the preceding identifiers. Some Product Advertising API operations can use these identifiers. That functionality is presented as a convenience function. You would typically only use one of these identifiers if you knew it but did not know the item's ASIN.

The validity of these identifiers varies by search index, as shown in the following table.

Search Index	UPC	EAN	SKU
Apparel	Y	Y	Y
Automotive	Y	Y	Y
Automotive	Y	Y	Y
Baby	Y	Y	Y
Beauty	Y	Y	Y
Books	Y	Y	Y
Classical	Y	Y	Y
DigitalMusic	Y	N	N

Product Advertising API Developer Guide
Items for Sale

Search Index	UPC	EAN	SKU
DVD	Y	Y	Y
Electronics	Y	Y	Y
HealthPersonalCare	Y	Y	Y
HomeGarden	Y	Y	Y
Industrial	Y	Y	Y
Jewelry	Y	Y	Y
Kitchen	Y	Y	Y
Magazines	N	Y	Y
Miscellaneous	Y	Y	Y
Music	Y	Y	Y
MusicalInstruments	Y	Y	Y
MusicTracks	N	N	N
OfficeProducts	Y	Y	Y
OutdoorLiving	Y	Y	Y
PCHardware	Y	Y	Y
PetSupplies	Y	Y	Y
Photo	Y	Y	Y
Software	Y	Y	Y
Tools	Y	Y	Y
Toys	Y	Y	Y
VHS	Y	Y	Y
Video	Y	Y	Y
Watches	Y	Y	Y

The following table shows the valid identifiers by locale.

Locale	Valid Item Identifiers
CA	ASIN, EAN, EISBN, SKU
CN	ASIN, EAN, EISBN, SKU, UPC
DE	ASIN, EAN, EISBN, SKU
ES	ASIN, EAN, EISBN, SKU, UPC
FR	ASIN, EAN, EISBN, SKU, UPC
IN	ASIN, EAN, EISBN, SKU, UPC

Locale	Valid Item Identifiers
IT	ASIN, EAN, ISBN, SKU, UPC
JP	ASIN, EAN, ISBN, JAN, SKU
MX	ASIN, EAN, ISBN, SKU, UPC
UK	ASIN, EAN, ISBN, SKU, UPC
US	ASIN, EAN, ISBN, SKU, UPC

The default value of `IdType` is ASIN. For non-ASIN searches, including searches by ISBN, JAN, SKU, UPC, EAN, and ISBN, a variety of additional parameters become mandatory, including a value for `IdType`. To use ISBN-13, use the EAN `IdType`.

Summary of Product Advertising API Operations

Topics

- [Find Items \(p. 23\)](#)
- [Find Out More About Specific Items \(p. 23\)](#)
- [Shopping Cart \(p. 24\)](#)
- [Other Operations \(p. 24\)](#)

The term operation is synonymous with the terms function and method. Product Advertising API operations are included in every request and they request Product Advertising API web servers to take some action, for example, to find items in Amazon's databases or to find similar items. Think of each operation as a different means of querying Amazon's databases for information. Product Advertising API does not offer any operations that upload information on Amazon databases.

The following sections provide an overview of Product Advertising API operations.

Find Items

Use the following Product Advertising API operations to return a list of items that satisfy your search criteria.

Product Advertising API Operation	Description
ItemSearch (p. 185)	Find items that are sold on Amazon.com.
SimilarityLookup (p. 203)	Find items that are similar to ones you've already found.

Find Out More About Specific Items

After you use the Product Advertising API operations to get a list of items, you can use the following operations to return additional information about specific items.

Product Advertising API Operation	Description
ItemLookup (p. 197)	Returns descriptions of specified items.

Shopping Cart

After customers find items they want to purchase, they put them in a shopping cart. The following Product Advertising API operations enable you to implement a fully-featured e-commerce shopping cart.

Product Advertising API Operation	Description
CartCreate (p. 214)	Create a shopping cart and add an item(s).
CartAdd (p. 207)	Add items to the shopping cart.
CartModify (p. 224)	Add to or remove items that are already in a shopping cart.
CartClear (p. 212)	Remove all of the items from a shopping cart.
CartGet (p. 220)	Retrieve the contents of a shopping cart with updated price and availability information.

Other Operations

The remaining operations cover a variety of functions.

Product Advertising API Operation	Description
BrowseNodeLookup (p. 194)	Amazon uses browse nodes as a means of organizing the millions of items in inventory. One browse node, for example, might be Carving Knives. This browse node might be the child of a more general browse node, Cutlery. <code>BrowseNodeLookup</code> enables you to find a browse node, its ancestors and progeny.

Visual Introduction to Product Advertising API

Topics

- [Tools to Find Items \(p. 25\)](#)
- [Tools to Find Out More About an Item \(p. 25\)](#)
- [Tools to Implement a Shopping Cart \(p. 26\)](#)

This chapter provides a series of screen shots which demonstrate how Product Advertising API functionality is labeled.

The labels show three kinds of implementations:

- Operations
- Response groups
- Elements in the response

Tools to Find Items

Usually, the first task a customer undertakes is search for an item. The following web page shows the implementation of this task using, in particular, ItemSearch and ItemLookup requests.

The screenshot shows the Amazon.com search results for the keyword "Saving Miss Oliver's". The search bar at the top contains the query. Below it, the results are displayed in a grid format. Each result includes a thumbnail image, the title, author, price, and availability status. The results are sorted by relevance. On the left, there are navigation links for categories like Business & Investing, Biographies & Memoirs, and Entertainment. On the right, there is a sidebar for "Listmania!" featuring a user's profile and a link to "Create a Listmania list".

①	ItemSearch SearchIndex parameter	⑨	Images response group
②	ItemSearch Keywords parameter	⑩	CustomerReviews response group
③	BrowseNodeLookup	⑪	Availability element returned by ItemSearch Availability parameter
④	ItemLookup	⑫	SearchBins response group
⑤	Author ItemSearch parameter	⑬	ItemSearch Title parameter
⑥	Offers response group	⑭	ItemSearch Condition parameter
⑦	ItemSearch Sort parameter		
⑧	Child Browsenodes		

The top entry fields show a drop down menu equal to search indices, and a text entry box for a keyword. Both of these parameter values are required for an ItemSearch request. The result, shown in this page, displays many of the elements returned by the ItemAttributes response group.

Tools to Find Out More About an Item

If the customer clicks the first image in the previous screen shot, the equivalent of a Product Advertising API ItemLookup request is sent. The following page shows the elements returned by such a request and some of the functionality implemented to move the customer to the next stage, such as getting seller information and adding the item to a Product Advertising API remote shopping cart.

Product Advertising API Developer Guide

Tools to Implement a Shopping Cart



1	MediumImage element	6	Availability element
2	Title element	7	ItemLookup with Images response group
3	Author element	8	TotalItems element
4	FormattedPrice element	9	SmallImage element
5	CartAdd operation	10	SimilarLookup

Tools to Implement a Shopping Cart

If the customer clicked the **Add to Cart** button in the previous screen shot, their cart would be displayed, as shown in the following figure. Most of the functionality on this page deals with modifying (`CartModify`) the items in the cart.



1	CartModify, Action=SaveForLater	4	PurchaseURL
2	CartModify, Quantity=0	5	CartModify, Quantity=1
3	CartModify		

Organization of Items on Amazon

Product Advertising API operations and response groups give you the tools you need to find the items in the store that you want quickly. Before understanding the search mechanisms that Product Advertising API operations provide and the filtering mechanisms that Product Advertising API response groups provide, it is important to first understand the way in which Amazon groups items for sale.

Topics

- [Browse Nodes \(p. 27\)](#)
- [Search Indices \(p. 33\)](#)

- [Variations \(p. 35\)](#)
- [Accessories \(p. 39\)](#)
- [Related Items \(p. 40\)](#)

Note

You will sometimes see in responses another organizing tool called `ProductGroup`. This is an older concept in Amazon's database design and it has been superseded by browse nodes and search indices. None of the Product Advertising API operations use `ProductGroup` as an input parameter

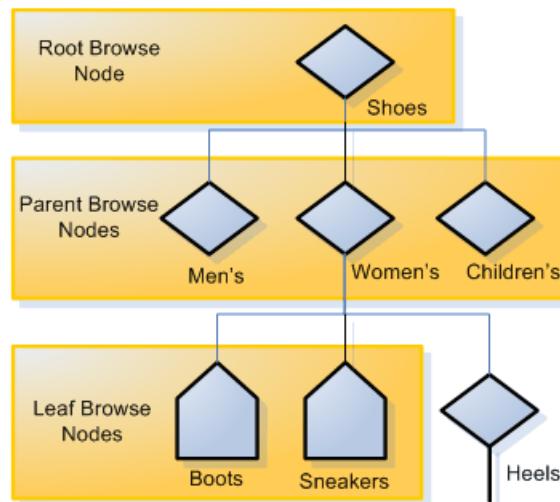
Browse Nodes

Topics

- [Browse Node Properties \(p. 28\)](#)
- [Browse Nodes and Items \(p. 29\)](#)
- [Browse Node IDs \(p. 30\)](#)
- [Browse Nodes and Search Indices \(p. 30\)](#)
- [Find Browse Nodes \(p. 31\)](#)

Amazon uses a hierarchy of nodes to organize its items for sale. Each node represents a collection of items for sale, such as Harry Potter books, not the items themselves. Product Advertising API calls the nodes, browse nodes because the customer can browse through the nodes to find the collection of items that interests them. For example, the customer might be interested in the browse nodes Literature & Fiction, Medicine, Mystery & Thrillers:, Nonfiction:, or Outdoors & Nature.

Browse nodes are related in a hierarchical structure; each browse node can be a leaf node or a parent node. A leaf node has no children nodes, a parent node does, as shown in the following figure.



Note

The figure is representational in nature and should not be construed to be the real browse node hierarchy used by Amazon.

As you can see in this example, the different levels of the hierarchical tree of nodes provides an organizational principle that is used to catalog and find items. The nodes progress from general to specific. For example, a top level browse node might be "Shoes". Its child browse nodes might be "Men's Shoes", "Women's Shoes", and "Children's Shoes". Child browse nodes are subsets of the parent's product

category. Navigating down the tree refines the search for items from the general to the specific. Going up the tree generalizes the search from the child browse node toward the root node.

Browse Node Properties

Browse nodes properties include:

Name

Describes the items associated with the browse node, such as "Mystery & Thrillers".

ID

A positive integer, for example, Literature & Fiction: (17), Medicine: (13996), Mystery & Thrillers: (18), Nonfiction: (53), Outdoors & Nature (290060).

Child nodes

Nodes that are subsets of the current node.

Parent nodes

Nodes that are supersets of the current node.

For example, the following XML response shows a browse node whose ID is 163357 and name is "Comedy".

```
<BrowseNode>
  <BrowseNodeId>163357</BrowseNodeId>
  <Name>Comedy</Name>
  <Children>
    <BrowseNode>
      <BrowseNodeId>599826</BrowseNodeId>
      <Name>Boxed Sets</Name>
    </BrowseNode>
    <BrowseNode>
      <BrowseNodeId>538712</BrowseNodeId>
      <Name>African American Comedy</Name>
    </BrowseNode>
  </Children>
  <Item>
    <Ancestors>
      <BrowseNode>
        <BrowseNodeId>549726</BrowseNodeId>
        <Name>Performing Arts</Name>
      </BrowseNode>
    </Ancestors>
  </Item>
```

In this example, the Comedy browse node has two child browse nodes, "Boxed Sets" and "African American Comedy", and one parent node, "Performing Arts".

Root Category

Search results can return items listed under multiple browse nodes. However, some nodes are more relevant than others. The `IsCategoryRoot` response tag identifies which browse node is the most relevant for an item in a specific marketplace. This functionality helps vendors classify items for sale.

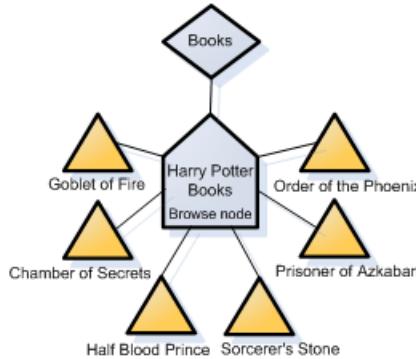
The `IsCategoryRoot` element applies to the marketplace specified in the request. So, it is possible for the `IsCategoryRoot` value to be different across marketplaces.

Note

The `IsCategoryRoot` value is not useful in the Books search index.

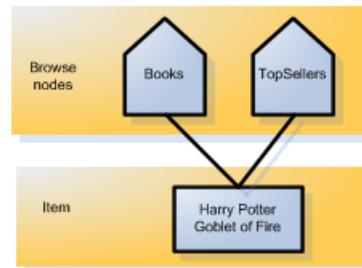
Browse Nodes and Items

Browse nodes are categories into which items for sale are organized. A single node might have many items associated with it, as shown in the following figure.



In this example, six items (Harry Potter book titles) are associated with the browse node: Harry Potter Books. This node is a leaf node and a child of the Books parent node. As the figure shows, many items can be associated with a single browse node. What all of the items have in common is described by the name of the browse node.

On the other hand, a single item can belong to more than one browse node, for example, a book might belong to the Books and TopSellers browse nodes.



At a later time, when the book comes off of the top sellers list, the book will be removed from its association with the TopSellers browse node. In this way, you can see that the association between items and browse nodes is dynamic.

Browse nodes are created and deleted according to item demand. For example, when a new toy or group of books starts selling briskly, a node would be created for it. For example, when pet rocks were popular, a node would have been created for pet rock items. When the sales of pet rocks declined significantly, the node would have been deleted. As you can see, some nodes are volatile by nature. For example, the items associated with the browse node, "Top Sellers", change frequently according to sales figures. Other browse nodes, such as Pet Rocks, exist only for a brief time.

Some browse nodes, however, are much longer lived. Top level nodes, for example, "Books" and "Apparel", have remained unchanged for years. So are the browse nodes associated with cities, as shown in the following table.

City	Browse Node
Boston	917982
Chicago	917984

City	Browse Node
New York	917976
San Francisco	917980
Seattle	917978
Washington, D.C.	917986

Browse Node IDs

Browse node IDs are positive integers that uniquely identify product sets, for example, Literature & Fiction: (17), Medicine: (13996), Mystery & Thrillers: (18), Nonfiction: (53), Outdoors & Nature: (290060). Amazon uses over 120,000 browse node IDs in the US locale alone.

While top level browse node values tend to remain the same, the values of others change often. Browse nodes are created and eliminated without notification. For that reason, it is advisable not to hard code browse node IDs into applications.

Browse node IDs are unique in one locale only, that is, the same browse node ID might be used in two locales but the names and purposes of those browse nodes might be unrelated.

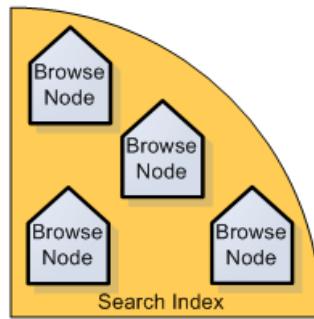
Browse node IDs are assigned internally by Amazon. There are no general rules for number assignments. You cannot assume, for example, that all IDs in the 2000s are related in any way.

If you have a browse node ID, you can find the name of the browse node by putting the ID into one of the following URLs, depending on the locale:

- [http://www.amazon.com/exec/obidos/tg/browse/-/\[Browse Node ID\]](http://www.amazon.com/exec/obidos/tg/browse/-/[Browse Node ID])
- [http://www.amazon.com.br/exec/obidos/tg/browse/-/\[Browse Node ID\]](http://www.amazon.com.br/exec/obidos/tg/browse/-/[Browse Node ID])
- [http://www.amazon.ca/exec/obidos/tg/browse/-/\[Browse Node ID\]](http://www.amazon.ca/exec/obidos/tg/browse/-/[Browse Node ID])
- [http://www.amazon.cn/exec/obidos/tg/browse/-/\[Browse Node ID\]](http://www.amazon.cn/exec/obidos/tg/browse/-/[Browse Node ID])
- [http://www.amazon.de/exec/obidos/tg/browse/-/\[Browse Node ID\]](http://www.amazon.de/exec/obidos/tg/browse/-/[Browse Node ID])
- [http://www.amazon.es/exec/obidos/tg/browse/-/\[Browse Node ID\]](http://www.amazon.es/exec/obidos/tg/browse/-/[Browse Node ID])
- [http://www.amazon.fr/exec/obidos/tg/browse/-/\[Browse Node ID\]](http://www.amazon.fr/exec/obidos/tg/browse/-/[Browse Node ID])
- [http://www.amazon.in/exec/obidos/tg/browse/-/\[Browse Node ID\]](http://www.amazon.in/exec/obidos/tg/browse/-/[Browse Node ID])
- [http://www.amazon.it/exec/obidos/tg/browse/-/\[Browse Node ID\]](http://www.amazon.it/exec/obidos/tg/browse/-/[Browse Node ID])
- [http://www.amazon.co.jp/exec/obidos/tg/browse/-/\[Browse Node ID\]](http://www.amazon.co.jp/exec/obidos/tg/browse/-/[Browse Node ID])
- [http://www.amazon.com.mx/exec/obidos/tg/browse/-/\[Browse Node ID\]](http://www.amazon.com.mx/exec/obidos/tg/browse/-/[Browse Node ID])
- [http://www.amazon.co.uk/exec/obidos/tg/browse/-/\[Browse Node ID\]](http://www.amazon.co.uk/exec/obidos/tg/browse/-/[Browse Node ID])

Browse Nodes and Search Indices

A search index is a more general classification than a browse node. For example, one search index is Books, and a browse node within that search index might be Harry Potter Books.



There can be many browse nodes within each search index. Some Product Advertising API search operations require a search index to limit the scope of the search. Specifying a browse node in addition to a search index returns more targeted search results. For example, a search operation, ItemSearch, that looked in the Books search index for book titles and descriptions that contained the word "dragon", would return thousands of titles. The same ItemSearch request with the Harry Potter browse node specified would return only those Harry Potter books that had dragons in them.

Find Browse Nodes

Product Advertising API offers several means of finding browse node IDs:

- **BrowseNodes** response group
 - Used with the ItemLookup, ItemSearch, and SimilarityLookup operations.
- **BrowseNodeInfo** response group
 - Used with the BrowseNodeLookup operation.

The **BrowseNodes** response group returns the browse node that an item belongs to as well as the ancestry of that browse node. The following response shows the ancestral browse nodes of High School. As you read down the response, the browse nodes ascend the browse node hierarchy. The last browse node, Books, in the response is the root browse node.

```
<Item>
  <ASIN>0976925524</ASIN>
  <BrowseNodes>
    <BrowseNode>
      <BrowseNodeId>69825</BrowseNodeId>
      <Name>High School</Name>
      <Ancestors>
        <BrowseNode>
          <BrowseNodeId>10605</BrowseNodeId>
          <Name>Education</Name>
          <Ancestors>
            <BrowseNode>
              <BrowseNodeId>53</BrowseNodeId>
              <Name>Nonfiction</Name>
            <Ancestors>
              <BrowseNode>
                <BrowseNodeId>1000</BrowseNodeId>
                <Name>Subjects</Name>
              <Ancestors>
                <BrowseNode>
```

```
<BrowseNodeId>283155</BrowseNodeId>
<Name>Books</Name>
```

There is, however, one caveat. When a node has more than one parent node, the `BrowseNodes` response group only returns one of the parents. There is no logic that determines which of the parent nodes it follows up the ancestral tree. Running the request multiple times, therefore might return a different set of ancestors for a node.

The `BrowseNodeInfo` response group returns browse node names, IDs, children and parent browse nodes, as shown in the following response snippet.

```
<BrowseNodeId>11232</BrowseNodeId>
<Name> Social Sciences</Name>
<Ancestors>
  <BrowseNode>
    <BrowseNodeId>53</BrowseNodeId>
    <Name>Nonfiction</Name>
    <Ancestors>
      <BrowseNode>
        <BrowseNodeId>1000</BrowseNodeId>
        <Name>Subjects</Name>
        <Ancestors>
          <BrowseNode>
            <BrowseNodeId>283155</BrowseNodeId>
            <Name>Books</Name>
          </BrowseNode>
        </Ancestors>
      </BrowseNode>
    </Ancestors>
  </BrowseNode>
</Ancestors>
<Children>
  <BrowseNode>
    <BrowseNodeId>11233</BrowseNodeId>
    <Name>Anthropology</Name>
  </BrowseNode>
  <BrowseNode>
    <BrowseNodeId>11242</BrowseNodeId>
    <Name>Archaeology</Name>
  </BrowseNode>
  <BrowseNode>
    <BrowseNodeId>3048861</BrowseNodeId>
    <Name>Children's Studies</Name>
  </BrowseNode>
</Children>
```

This response shows that the Social Sciences browse node has three child browse nodes: Anthropology, Archaeology, and Children's Studies, and an ancestry that starts with the parent node, Nonfiction and ends with the root browse node, Books.

Use `BrowseNodeLookup` to navigate the browse node tree. With every response, you use the browse node IDs returned to refine your search until you reach the desired browse node. You might, for example, navigate down the tree to refine a search or retrieve the root browse node to return, for example, the top sellers in the product category. To do that, you would use the ID of the root browse node found in the `BrowseNodeInfo` response as the value for the `BrowseNode` parameter in an `ItemSearch` request. That request would include the `TopSellers` response group so that the top sellers of the product category are returned.

Note

If a browse node has multiple ancestors, only one of them is returned in the response.

In contrast, `BrowseNodeLookup` only returns child browse nodes that are the direct descendant of the browse node in the request. You could use any of the browse node IDs to either find additional, related browse nodes by using another `BrowseNodeLookup` request, or to focus an `ItemSearch` request.

Use `ItemSearch` and `ItemLookup` and specify the `BrowseNodes` response group.

Find a browse node ID that is similar to the one you want and use `BrowseNodeLookup` to investigate its child and ancestor browse nodes.

Related Topics

- [BrowseNodes \(p. 236\)](#)

Search Indices

Topics

- [Search Indices and Locales \(p. 34\)](#)
- [Combined Search Indices \(p. 34\)](#)

On Amazon, you can find just about anything—from a book to a \$100,000 piece of jewelry. The job of finding the items you want to buy is made more difficult by having so many items for sale. Returning too many items in a response is almost as bad as returning no items at all. Amazon addresses this difficulty by placing every item it sells into a product category, called a search index.

You can think of search indices as dividing into groups everything for sale on Amazon, as shown in the following figure.



By selecting one slice in a request, you avoid returning items in all of the other slices. This functionality makes requests more targeted and reduces search times. For example, when looking for a specific book, it is best to specify the Books search index in the request.

```
SearchIndex=Books
```

Titles and keywords used to find items often occur in multiple search indices. For example, if you were searching using "Harry%20Potter" as your keyword, you would get results in many search indices, including Books, DVD, Video, and Music. By specifying the search index you are interested in, your request becomes much better targeted.

Specifying the wrong search index leads to no results or results that do not meet the customer's search criteria. For example, if your keyword was carburetor and the search index was Kitchen, you might not

get any items that match the search criteria. Change the search index to Automotive and you'd get many matches. For a complete list of search indices that are valid per locale, see [Locale Reference \(p. 328\)](#).

Search Indices and Locales

Some search indices work in all locales while others do not. Supported search indices in a locale can change.

As more sellers and items are added to the marketplace, more search indices will be supported in each locale.

When you specify a search index in a request, ensure it's supported in the locale of interest. For a list of the search indices supported in different locales, see [Locale Reference \(p. 328\)](#).

Combined Search Indices

Topics

- [All Search Index \(p. 34\)](#)

As a convenience, some of the search indices are combinations of other search indices, for example:

All

Searches through all search indices. Only five pages of items can be returned where each page contains up to five items.

Blended

Combines the following search indices: DVD, Electronics, Toys, VideoGames, PCHardware, Tools, SportingGoods, Books, Software, Music, GourmetFood, Kitchen, and Apparel search indices.

Music

Combines Classical, DigitalMusic, and MusicTracks search indices.

Video

Combines DVD and VHS search indices.

These search index combinations are helpful when you are not sure which search index to specify. The downside of using one of these combined search indices is that they might return quite a few items in the response.

More commonly, you will specify individual search indices in requests, for example,

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
Condition=All&  
ResponseGroup=Images&  
SearchIndex=Books&  
Title=Harry%20Potter  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

All Search Index

You can use the All search index to do an ItemSearch search through all search indices. There are, however, a number of restrictions placed on this request. The only parameter that you can use in the

request is `Keywords`. You cannot, for example, sort results. Results are restricted to the first five pages of results. Each page can have up to five results.

Note

The list of all available search indices by locale can be found on the search indices page.

Variations

Topics

- [Variation Parents \(p. 35\)](#)
- [Return Variations \(p. 36\)](#)
- [Variation Dimensions \(p. 38\)](#)

Often, an item comes in a variety of sizes and colors. A shirt, for example, might come in four different sizes and six different colors.



Each color and size combination is called a variation. Each variation, such as a medium, blue shirt, is an item that a customer can buy. For that reason, each variation has its own ASIN. For example, if a shirt came in four sizes and six colors, there would be 24 variations, each with a unique ASIN.

Variation Parents

The abstraction of the variations is called the variation parent. The title element of the variation parent names the variations, for example, "Long Sleeve Classic Pocket Tee". Because the parent ASIN is an abstraction, it cannot be purchased, that is, it is not associated with an offer. The following request uses the `Offers` response group in an `ItemLookup` of a parent variation.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=B00006XYAB&  
IdType=ASIN&  
Condition=All  
ResponseGroup=ItemAttributes,Offers  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

As you can see in the following response snippet, no offers are returned.

```
<Item>  
<ASIN>B00006XYAB</ASIN>  
<ItemAttributes>
```

```
<Binding>Apparel</Binding>
<Brand>Eddie Bauer</Brand>
<Department>mens</Department>
<FabricType>leather</FabricType>
<Feature>Cotton.</Feature>
<Feature>Single chest pocket.</Feature>
<Feature>Side seamed to prevent twisting.</Feature>
<Feature>Coverstitched seams add strength.</Feature>
<ProductGroup>Apparel</ProductGroup>
<Title>LongSleeve Classic Pocket Tee</Title>
</ItemAttributes>
<OfferSummary>
  <TotalNew>0</TotalNew>
  <TotalUsed>0</TotalUsed>
  <TotalCollectible>0</TotalCollectible>
  <TotalRefurbished>0</TotalRefurbished>
</OfferSummary>
<Offers>
  <TotalOffers>0</TotalOffers>
  <TotalOfferPages>0</TotalOfferPages>
</Offers>
</Item>
```

Return Variations

By default, ItemSearch only returns parent variations.

To return the associated variations and their images

- Include in the request the Variations and VariationImages response groups.

For example, when you add Variations to the preceding request, the response includes, in addition to what is shown, variations and their offers, as shown in the following response snippet.

```
<Offer>
  ...
  <OfferListing>
    <OfferListingId>4L2h0ymPzs3lDap8fU9Kw8hmKiPs5qVknFiifrvixAkRDZivZpZ
    sSFte0YMLtbo6MVWHTCzyKjT9UZI%2BpvF8Ax%2BFeSG29QGqQ4eF6ZspRD5fdxj8zm
    Rp%2Bw%3D%3D</OfferListingId>
  ...
</Offer>
<Offer>
  ...
  <OfferListing>
    <OfferListingId>PCOIjALkrkCts8UYi3pqf%2BqBaH5xbmJfiko
    ab5fNh1F7yRMeh19NjxBEBBNz9GDFMMYtH0EI06Y1Vbar7raOf6ieUnzzST94RN8wwiXpEn2jaW
    mZ%2F%2BYc7Q%3D%3D</OfferListingId>
```

Each of these offers is associated with an Item element and each of these items is a variation. The following snippet, which comes from the same response, shows how each variation is differentiated by size.

```
<Item>
  <ItemAttributes>
```

```
<Brand>Eddie Bauer</Brand>
...
</ItemAttributes>
<VariationAttributes>
  <VariationAttribute>
    <Name>ClothingSize</Name>
    <Value>Small Regular</Value>
  </VariationAttribute>
  <VariationAttribute>
    <Name>ClothingSize</Name>
    <Value>Medium Regular</Value>
  </VariationAttribute>
  <VariationAttribute>
    <Name>ClothingSize</Name>
    <Value>Large Regular</Value>
  </VariationAttribute>
  <VariationAttribute>
    <Name>ClothingSize</Name>
    <Value>X-Large Regular</Value>
  </VariationAttribute>
</VariationAttributes>
...
</Item>
```

Return Variation Dimension Information Only

The Variations and VariationSummary response groups return a great deal of information about each item in the response.

To see the values of the variation dimensions

- Use the VariationMatrix response group in an ItemSearch request.

```
<Item>
  <ASIN>B0008G23PQ</ASIN>
  <Variations>
    <VariationDimensions>
      <VariationDimension>ClothingSize</VariationDimension>
      <VariationDimension>Color</VariationDimension>
    </VariationDimensions>
  <Item>
    <ASIN>B0008EOA9U</ASIN>
    <VariationAttributes>
      <VariationAttribute>
        <Name>Color</Name>
        <Value>Grey</Value>
      </VariationAttribute>
      <VariationAttribute>
        <Name>ClothingSize</Name>
        <Value>29W x 30L</Value>
      </VariationAttribute>
    </VariationAttributes>
  </Item>
  <Item>
    <ASIN>B0008EO9J6</ASIN>
    <VariationAttributes>
```

```
<VariationAttribute>
  <Name>Color</Name>
  <Value>Navy</Value>
</VariationAttribute>
<VariationAttribute>
  <Name>ClothingSize</Name>
  <Value>29W x 30L</Value>
</VariationAttribute>
</VariationAttributes>

</Item>
```

This response snippet shows that the VariationMatrix response group returns the names of the dimensions along with their values for each returned item. For more information, see [VariationMatrix Response Group \(p. 302\)](#).

Related Topics

- [Variations Response Group \(p. 297\)](#)
- [VariationSummary Response Group \(p. 307\)](#)
- [VariationMatrix Response Group \(p. 302\)](#)

Variation Images

Each variation has its own set of images and those images are returned by the VariationImages response group, for example:

```
<SmallImage>
  <URL>http://images.amazon.com/images/P/B9999999A.01._SCTHUMBZZZ_.jpg</URL>

  <Height Units="pixels">60</Height>
  <Width Units="pixels">60</Width>
</SmallImage>

<MediumImage>
  <URL>http://images.amazon.com/images/P/B9999999A.01._SCMZZZZZZ_.jpg</URL>

  <Height Units="pixels">140</Height>
  <Width Units="pixels">140</Width>
</MediumImage>
```

These images are small and medium size images of the same child variation. For more information about image sets and variation images, see [Motivating Customers to Buy \(p. 127\)](#).

Variation Dimensions

Variations can differ from one another in a variety of ways. Size and color are common ways for apparel variations to differ. The ways in which variations differ are called dimensions. Parent variations relay that information with the following response elements:

- [VariationDimensions](#)
- [VariationDimension](#)

The values encapsulated by these elements in the parent variation response specify the variation dimensions for the child variations, for example:

```
<VariationDimensions>
  <VariationDimension>ClothingSize</VariationDimension>
  <VariationDimension>Color</VariationDimension>
</VariationDimensions>
```

The following response snippet from one of the associated child variations shows that the dimensions specified in the parent variation are used as variation attributes in the child variation.

```
<Item>
  ...
  <VariationAttributes>
    <VariationAttribute>
      <Name>Color</Name>
      <Value>Grey</Value>
    </VariationAttribute>
    <VariationAttribute>
      <Name>ClothingSize</Name>
      <Value>29W x 30L</Value>
    </VariationAttribute>
  </VariationAttributes>
</Item>
```

Accessories

Some items have associated accessories. For example, a camera might have a camera case, flash card, and battery.



Each of these accessories has an item ID, such as an ASIN, as shown in the following response snippet. In this example, the main item, B00008OE6I, the camera, returned in the response comes with two accessories, B00003G1RG, a compact flash card, and B00004WCCT, a leather camera case.

```
<Item>
  <ASIN>B00008OE6I</ASIN>
  <Accessories>
    <Accessory>
      <ASIN>B00003G1RG</ASIN>
      <Title>Viking 128 MB CompactFlash Card (CF128M)</Title>
    </Accessory>
    <Accessory>
      <ASIN>B00004WCCT</ASIN>
      <Title>Canon Soft Leather Case for Canon Digital ELPH Camer
as(Black)</Title>
    </Accessory>
  </Accessories>
</Item>
```

The following figure shows those items.



As you can see in this example, Amazon groups accessories with the main item for sale. Product Advertising API makes it easy to retrieve all of the accessories associated with a main item by using the Accessories response group in an [ItemLookup](#) or [ItemSearch](#) request. As you can see in the previous response snippet, each accessory listed in the response includes the accessory's title and item identifier, such as an ASIN.

Note

When you have the item ID of the main item, the Accessories response group returns the item's accessories. The reverse, however, is not true, that is, if you have the item ID of an accessory, you cannot use the Accessories response group to return the main item or the other accessories associated with the main item.

Related Items

The `RelatedItems` response group returns information about items related to the one specified in an `ItemLookup` request. The item is specified in an `ItemLookup` request. Digital items include downloadable music (search index: `MP3Downloads`), downloadable video (search index: `UnboxVideo`), and digital books (search index: `KindleStore`). A related item could be, for example, all of the shows in a TV series that are available separately, or, for example, all of the songs on a CD.

The basis upon which the item(s) are related is specified by the `RelationshipType` parameter. The `RelatedItems` response group requires that you include in the `ItemLookup` request the `RelationshipType` parameter. Sample values include `Episode`, `Season`, `Tracks`, and `Variation`. For a complete list of values, see [ItemLookup \(p. 197\)](#).

Each `ItemLookup` request can return, at most, ten related items. To return additional items, use the `RelatedItemPage` parameter. A value of 2, for example, returns the second set of ten related items.

Requests

Topics

- [Anatomy of a REST Request \(p. 41\)](#)
- [Request Limitations \(p. 44\)](#)
- [REST Syntax \(p. 44\)](#)
- [SOAP Requests \(p. 46\)](#)
- [Request Authentication \(p. 48\)](#)
- [Batch Requests \(p. 61\)](#)
- [Parameters Common to All Product Advertising API Requests \(p. 63\)](#)
- [Responses \(p. 66\)](#)
- [Paging and Sorting Through Responses \(p. 70\)](#)

The Product Advertising API web service supports REST requests for remotely calling Product Advertising API operations hosted by Amazon servers. REST requests are simple HTTP requests, using either the GET method with parameters in the URL, or the POST method with parameters in the POST body. The response is an XML document that conforms to a schema.

You may use REST requests because they are more intuitive than their SOAP counterpart or because a SOAP toolkit is not available for your platform. The example requests in this guide are in REST.

Anatomy of a REST Request

Product Advertising API REST requests are URLs, as shown in the following example.

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService&Operation=ItemSearch&AWSAccessKeyId=[Access Key ID]&AssociateTag=[ID]&SearchIndex=Apparel&Keywords=Shirt&Timestamp=[YYYY-MM-DDThh:mm:ssZ]&Signature=[Request Signature]
```

If you substituted real IDs in this request and put the entire example in a browser, you are sending Product Advertising API a request.

Although the previous example is the form you enter in a browser, it is difficult to read. For this reason, this guide presents the same request as follows:

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
Operation=ItemSearch&  
AWSAccessKeyId=[Access Key ID]&  
AssociateTag=[ID]&  
SearchIndex=Apparel&  
Keywords=Shirt  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

For more information about `Timestamp` and `Signature`, see [Authentication Parameters \(p. 50\)](#).

For information about signing a request using example AWS access identifiers, see [Example REST Requests \(p. 55\)](#).

General Request Format

Part of every Product Advertising API request is always the same. The other part of the request varies according to the parameters used in the request, as shown in the following figure.



Request Terms that Remain the Same

The first two lines in the preceding example have the following:

- The endpoint, <http://webservices.amazon.com/onca/xml>.
- The service name, **AWSECommerceService**.

The service name in the request specifies that the request should be sent by the web servers to Product Advertising API. This line is always the same in every Product Advertising API request, regardless of locale.

```
Service=AWSECommerceService&
```

The endpoint value varies by locale, but there are only two endpoints per locale. One endpoint in a locale is the secure version of the other endpoint. The following table lists the endpoints for Product Advertising API requests.

Locale	Endpoint
BR	http://webservices.amazon.com.br/onca/xml https://webservices.amazon.com.br/onca/xml
CA	http://webservices.amazon.ca/onca/xml https://webservices.amazon.ca/onca/xml
CN	http://webservices.amazon.cn/onca/xml https://webservices.amazon.cn/onca/xml
DE	http://webservices.amazon.de/onca/xml https://webservices.amazon.de/onca/xml
ES	http://webservices.amazon.es/onca/xml https://webservices.amazon.es/onca/xml
FR	http://webservices.amazon.fr/onca/xml https://webservices.amazon.fr/onca/xml
IN	http://webservices.amazon.in/onca/xml https://webservices.amazon.in/onca/xml
IT	http://webservices.amazon.it/onca/xml https://webservices.amazon.it/onca/xml
JP	http://webservices.amazon.co.jp/onca/xml https://webservices.amazon.co.jp/onca/xml
MX	http://webservices.amazon.com.mx/onca/xml https://webservices.amazon.com.mx/onca/xml
UK	http://webservices.amazon.co.uk/onca/xml https://webservices.amazon.co.uk/onca/xml
US	http://webservices.amazon.com/onca/xml https://webservices.amazon.com/onca/xml

The endpoint remains the same for all Product Advertising API requests, as shown in the following example.

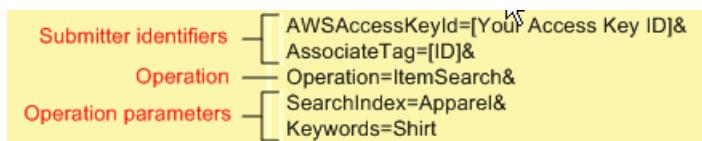
```
http://webservices.amazon.com/onca/xml
```

The third and fourth lines identify the request submitter. The AWS access key ID is required. It identifies the request submitter. You receive an AWS access key ID when you register for Product Advertising API.

The identifier `AssociateTag` is required. This is the ID for your Associate account. If you are an Associate, you must include your Associate tag in each request to be eligible to receive a referral fee for a customer's purchase.

Request Terms that Change

The remaining terms in the request vary for each operation. The terms, however, follow a pattern, as shown in the next figure.



The `Operation` parameter is required. Its value is one of the Product Advertising API operations. These operations are described throughout this guide.

The last lines, operation parameters, are representative of required and optional parameters. Requests can contain zero or more operation (up to ten) parameters.

Response Groups

An optional parameter for all Product Advertising API operations is `ResponseGroup`. Response groups control the type of information returned by the request. For example, the Large response group returns a great deal of information about the items included in a response, whereas the Medium and Small response groups return less.

There are response groups that return specific item information. For example, if you want to return images of the items, include the `Image` response group in the request. If you want price information, include the `Offer` response group in the request. To get browse node information, include the `BrowseNode` response group.

Each Product Advertising API operation can work with only a subset of all Product Advertising API response groups. The valid response groups that each Product Advertising API operation can use are listed in the [API Reference \(p. 184\)](#).

All Product Advertising API operations use some response group by default. So specifying additional response groups is optional. Every Product Advertising API operation uses the `Request` response group, which echoes the operation name and the input parameters sent in the request. The other response groups used by default vary by operation.

For example, `CartCreate`, `CartAdd`, and `CartModify` use, by default, the `Cart` response group, which provides detailed information about the items in a cart.

When you see request examples in this guide, note the `Operation` parameter and all of the required and optional parameters.

Request Limitations

Some Product Advertising API operations have many parameters. You can combine multiple single requests in one, longer batch request. The upper limit is bounded by the maximum number of characters that can be in a request. The maximum number differs by browser. For example, the limit for a URL in Internet Explorer is a little more than 2000 characters. It is unusual to have a request that approached this upper boundary.

REST Syntax

Topics

- [Spaces in Requests \(p. 44\)](#)
- [Separator Characters \(p. 44\)](#)
- [Setting Parameter Values \(p. 44\)](#)

One of the values of using REST is that its syntax is simple, which makes REST requests easy to read. This section summarizes all of the REST syntax rules when you create a REST request.

Spaces in Requests

Because a REST request is a URL, there can be no spaces between the parts of a request. A browser will stop reading when it runs across the first space. For example, if the last parameter read, `Keywords=Blue Shirts`, the request will end on "Blue". "Shirts" won't be read. If you have key words that have spaces, you must URL-encode the space using %20. For the preceding example to work, include a URL-encoded space.

```
Keywords=Blue%20Shirts
```

The same problem occurs if you have spaces between the parameters in a request, as shown in the following example.

```
SearchIndex=Apparel& Keywords=Shirt
```

In this example, the request ends at "Apparel&". This type of mistake returns an error, because parameters required by the operation are not read. For best practices, remove all spaces within a request.

Separator Characters

The question mark (?) and ampersand (&) separate the terms in a REST request. The first term in the request must always be the endpoint, which in the previous example is <http://webservices.amazon.com/onca/xml>. A question mark always follows the endpoint. The question mark tells the Product Advertising API web servers to start parsing the request for parameters.

Ampersands separate all other parameter name-value pairs in the request. The order of the parameter name-value pairs is not important, as long as they are after the question mark.

Setting Parameter Values

Topics

- [Parameter Names and Values are Case Sensitive \(p. 45\)](#)
- [Compound Parameters \(p. 45\)](#)

Request parameter values are set using the format.

```
ParameterName=value
```

The following example is a parameter/value pair.

```
Operation=ItemSearch
```

Parameter values must be URL-encoded. There are some characters, such as an asterisk or space, that cannot go into a URL. There are equivalents of these characters that you use in requests instead. For example, the URL encoded equivalent of a space is %20. So, instead of writing Name=John Smith, write Name=John%20Smith.

Parameter Names and Values are Case Sensitive

Parameter names and values are case sensitive. For example, the following declaration is correct.

```
SearchIndex=Apparel
```

The following examples return errors because the capitalization is incorrect.

```
Searchindex=Apparel  
SearchIndex=apparel
```

In these examples, parameter names and values start with capitals. If the name or value is a compound word, the beginning of each new word is capitalized. For example, in the parameter SearchIndex, the "I" is capitalized.

Compound Parameters

Most parameters can be specified in a REST request with just the name of the parameter and an appropriate value. The value needs to be URL-encoded to make the request a valid URL, as shown in the following example.

```
Author=Steve%20Davenport
```

However, some parameters can be repeated in a request. In that case, the parameter names are differentiated by adding a period (.) after the parameter name and then a sequence number, as shown in the following example.

```
Item.1=1234&  
Item.2=2345
```

Other parameters can be repeated but, in addition, have associated parameters. These parameters extend the preceding example by adding another period and the associated parameter name, as shown in the following example.

```
Item.1.ASIN=3456789123&  
Item.1.Quantity=2
```

In this example, the item being added to a shopping cart has an identifier, the ASIN, and a quantity value. The equivalent expression in an XML document is, as shown in the following example.

```
<Item>
  <ASIN>3456789123</ASIN>
  <Quantity>2</Quantity>
</Item>
```

The sequence number associates the identifier and its quantity value. The following example shows two compound parameters.

```
Item.1.ASIN=3456789123&
Item.1.Quantity=2&
Item.2.ASIN=123456&
Item.2.Quantity=1
```

The sequence numbers associate the ASIN and quantity values. This means that the compound parameters can be written in a different order without causing a problem.

```
Item.1.ASIN=3456789123&
Item.2.ASIN=123456&
Item.1.Quantity=2&
Item.2.Quantity=1
```

SOAP Requests

Product Advertising API supports the SOAP message protocol for calling Product Advertising API operations over an HTTP connection. The easiest way to use the SOAP interface with your application is to use a SOAP toolkit appropriate for your platform. SOAP toolkits are available for most popular programming languages and platforms.

The service's Web Services Definition Language (WSDL) file describes the operations and the format and data types of their requests and responses. Your SOAP toolkit interprets the WSDL file to provide your application access to the operations. For most toolkits, your application calls a service operation using routines and classes provided or generated by the toolkit. For more information, see [WSDL Location \(p. 18\)](#).

This section describes the structure of a SOAP request. For information about authenticating a SOAP request, see [Authenticating SOAP Requests \(p. 59\)](#).

The Structure of a SOAP Request

A SOAP request is an XML data structure generated by a SOAP toolkit that is sent to a web service. The root element of this structure is named after the operation and contains the values for the operation's parameters.

The root element of every request must have:

- A value for AWSAccessKeyId to authenticate the request.

For more information, see [Managing Your AWS Credentials \(p. 49\)](#).

- An endpoint, which is the destination for the request.

For more information, see [Request Terms that Remain the Same \(p. 41\)](#).

- A request element, which contains the values for the operation's parameters

SOAP Endpoints

SOAP requests use endpoints in their requests. The endpoint value varies by locale, but there are only two endpoints per locale. One endpoint in a locale is the secure version of the other endpoint. The following table lists the endpoints to use in Product Advertising API SOAP requests.

Locale	Endpoint
BR	http://webservices.amazon.br/onca/soap https://webservices.amazon.br/onca/soap
CA	http://webservices.amazon.ca/onca/soap https://webservices.amazon.ca/onca/soap
CN	http://webservices.amazon.cn/onca/soap https://webservices.amazon.cn/onca/soap
DE	http://webservices.amazon.de/onca/soap https://webservices.amazon.de/onca/soap
ES	http://webservices.amazon.es/onca/soap https://webservices.amazon.es/onca/soap
FR	http://webservices.amazon.fr/onca/soap https://webservices.amazon.fr/onca/soap
IN	http://webservices.amazon.in/onca/soap https://webservices.amazon.in/onca/soap
IT	http://webservices.amazon.it/onca/soap https://webservices.amazon.it/onca/soap
JP	http://webservices.amazon.co.jp/onca/soap https://webservices.amazon.co.jp/onca/soap
MX	http://webservices.amazon.com.mx/onca/soap https://webservices.amazon.com.mx/onca/soap
UK	http://webservices.amazon.co.uk/onca/soap https://webservices.amazon.co.uk/onca/soap
US	http://webservices.amazon.com/onca/soap https://webservices.amazon.com/onca/soap

The XML Message for an ItemSearch SOAP Request

The following example is the XML for a SOAP message that calls the `ItemSearch` operation. While you will probably not be building the SOAP message for a service request manually, it is useful to see what

your SOAP toolkit produces when provided with the appropriate values. Many SOAP toolkits require that you build a request data structure similar to the XML to make a request.

The `ItemSearch` element contains the parameters common to all requests. The `Request` element contains the `ItemSearch` parameters, `SearchIndex` and `Keywords`.

```
<?xml version="1.0" encoding="UTF-8" ?>
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <soapenv:Body>
        <ItemSearch
            xmlns="http://webservices.amazon.com/AWSECSCCom
merce/onca/soap">
            <AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</AWSAccessKeyId>
            <Request>
                <SearchIndex>Books</SearchIndex>
                <Keywords>Harry%20Potter</Keywords>
            </Request>
        </ItemSearch>
    </soapenv:Body>
</soapenv:Envelope>
```

For information about signatures and how to authenticate SOAP requests, see [Authenticating SOAP Requests \(p. 59\)](#).

Request Authentication

Topics

- [What is Authentication? \(p. 48\)](#)
- [Managing Your AWS Credentials \(p. 49\)](#)
- [HMAC-SHA256 Signatures for REST Requests \(p. 49\)](#)
- [Authenticating REST Requests \(p. 51\)](#)
- [Authenticating SOAP Requests \(p. 59\)](#)

This section covers the basics of authentication, how your AWS credentials are used to support authentication, and how to create an HMAC-SHA256 signature. This section also covers the request authentication requirements for Query and SOAP.

What is Authentication?

Authentication is a process by which the identity of the request sender is verified.

Overview of the authentication process

1. The sender obtains the required credentials. For the Product Advertising API, the credentials are the AWS access key ID and secret key.
2. The sender submits a request with the credentials to the recipient.
3. The recipient uses the credentials to verify the sender.
4. If the credentials are valid, the request is processed, and response information is returned. If the credentials are invalid, the recipient rejects the request and returns an error message.

Managing Your AWS Credentials

When you register as a Product Advertising API developer, an AWS account is created for you, along with a pair of credentials: an access key ID and secret access key. You use these credentials to submit requests to the Product Advertising API.

- Access key ID (a 20-character, alphanumeric sequence)
For example: AKIAIOSFODNN7EXAMPLE
- Secret access key (a 40-character sequence)
For example: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY

Note

If you haven't registered to become a Product Advertising API developer, see [Getting Started \(p. 4\)](#).

To manage your AWS credentials

1. Sign in to <http://aws.amazon.com> with your AWS account.
2. Click **Account Name**, and then click **Security Credentials**.
3. A pop-up message appears. Click **Continue to Security Credentials**.
4. Click **Access Keys (Access Key ID and Secret Access Key)** to locate your access key ID. Under **Actions**, you can delete your access key ID or deactivate it temporarily.
5. You can view your access key ID, but not the secret access key, from this page. If you lost or don't remember your secret access key, you'll need to create a new pair of credentials.
 - a. On the same page, click **Create New Access Key**, and then click **Show Access Key or Download Key File** to retrieve the credentials.
 - b. Save the access key information in a safe location.

Important

Only you and AWS should know your secret access key. It is important to keep it confidential to protect your account. Never include it in your requests to AWS. Never email it to anyone. Do not share it outside your organization, even if an inquiry appears to come from AWS or Amazon.com. No one representing Amazon will ever ask you for your secret access key.

Related Topics

- [HMAC-SHA256 Signatures for REST Requests \(p. 49\)](#)
- [Authenticating REST Requests \(p. 51\)](#)

HMAC-SHA256 Signatures for REST Requests

Contents

- [Authentication Parameters \(p. 50\)](#)
- [Basic Authentication Process \(p. 50\)](#)

This section describe how Product Advertising API uses HMAC-SHA256 signatures to authenticate REST requests.

Authentication Parameters

The following parameters are used by Product Advertising API for REST authentication:

Signature — Required

There is no default value. A signature is created by using the request type, domain, the URI, and a sorted string of every parameter in the request (except the Signature parameter itself) with the following format <parameter>=<value>&. After it's properly formatted, create a base64-encoded HMAC-SHA256 signature with your AWS secret key. For more information, see [Example REST Requests \(p. 55\)](#).

Timestamp — Required

There is no default value. The time stamp you use in the request must be a `dateTime` object, with the complete date, including hours, minutes, and seconds. This is a fixed-length subset of the format defined by ISO 8601, represented in Universal Time (GMT): `YYYY-MM-DDThh:mm:ssZ` (where T and Z are literals). For more information, see [Date and Time Formats](#).

Important

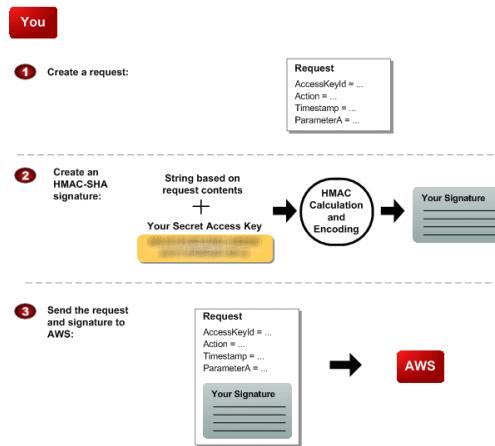
If you are using .NET, you should not send overly specific time stamps, due to differing interpretations of how extra time precision should be dropped. To avoid overly specific time stamps, manually construct `dateTime` objects with no more than millisecond precision.

Basic Authentication Process

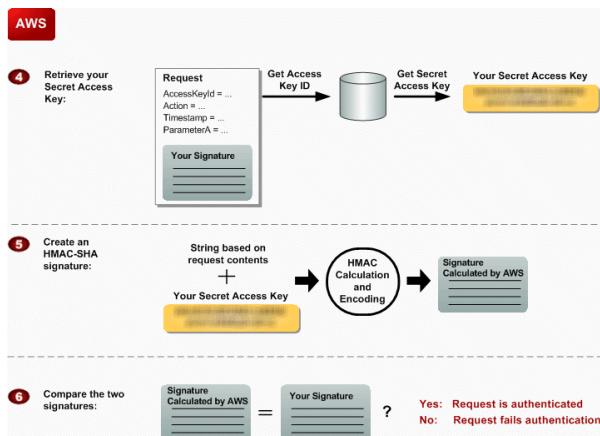
The following describes the steps required to authenticate requests to AWS using an HMAC-SHA256 request signature.

1. You construct a request to AWS.
2. You calculate a keyed-hash message authentication code (HMAC-SHA256) signature with your secret access key. For information about HMAC, see [RFC2104](#).
3. You include the signature and your access key ID in the request, and then send the request to AWS.
4. The Product Advertising API uses your access key ID to look up your secret access key.
5. Product Advertising API generates a signature from the request data and the secret access key with the same algorithm you used to calculate the signature you sent in the request.
6. If the signature generated by AWS matches the one you sent in the request, the request is considered authentic. If the comparison fails, the request is discarded, and AWS returns an error response.

Steps you perform



Steps AWS performs



Authenticating REST Requests

This section describes how to create a signature. The Product Advertising API supports only Signature Version 2.

To create the signature

1. Create the canonicalized query string that you need later in this procedure:
 - a. Sort the UTF-8 query string components by parameter name with natural byte ordering. The parameters can come from the GET URI or from the POST body (when Content-Type is application/x-www-form-urlencoded).
 - b. URL encode the parameter name and values according to the following rules:
 - Do not URL encode any of the unreserved characters that RFC 3986 defines. These unreserved characters are A-Z, a-z, 0-9, hyphen (-), underscore (_), period (.), and tilde (~).
 - Percent encode extended UTF-8 characters in the form %XY%ZA.
 - Percent encode the space character as %20 (and not +, as common encoding schemes do).
 - Percent encode all other characters with %XY, where X and Y are hex characters 0-9 and uppercase A-F.

Perl Note:

The commonly used URI::Escape CPAN module uses RFC 2396. This has five additional reserved characters: asterisk (*), left and right parenthesis ((and)), single quote (') and exclamation (!). To follow RFC 3986 use:

```
URI::Escape::uri_escape( $parameter_value, "^A-Za-z0-9\-\_\~" )
```

Java Note:

URLEncoder uses + for space, and won't encode asterisk (*), and encodes tilde (~) when not necessary. To follow RFC 3986 use:

```
URLEncoder.encode(value, UTF_8_Encoding).replace("+", "%20").replace("*", "%2A").replace("%7E", "~");
```

C# Note:

Use uppercase hex characters.

Tip

Currently all Product Advertising API service parameter names use unreserved characters, so you don't need to encode them. However, you might want to include code to handle parameter names that use reserved characters, for possible future use.

- c. Separate the encoded parameter names from their encoded values with the equals sign (=) (ASCII character 61), even if the parameter value is empty.
 - d. Separate the name-value pairs with an ampersand (&) (ASCII code 38).
2. Create the string to sign according to the following pseudo-grammar (the "\n" represents an ASCII newline).

```
StringToSign = HTTPVerb + "\n" +
    ValueOfHostHeaderInLowercase + "\n" +
    HTTPRequestURI + "\n" +
    CanonicalizedQueryString <from the preceding step>
```

The HTTPRequestURI component is the HTTP absolute path component of the URI up to, but not including, the query string. If the HTTPRequestURI is empty, use a forward slash (/).

Note

HTTPRequestURI is always "/onca/xml" for Product Advertising API. HTTPVerb is either GET or POST.

3. Calculate an RFC 2104-compliant HMAC with the string you just created, your AWS secret access key as the key, and SHA256 as the hash algorithm.
For more information, see [RFC2104](#).
4. Convert the resulting value to base64.
5. Use the resulting value as the value of the `Signature` request parameter.

The final signature you send in the request must be URL encoded as specified in [RFC 3986](#). If your toolkit URL encodes your final request, then it handles the required URL encoding of the signature. If your toolkit doesn't URL encode the final request, then make sure to URL encode the signature before you include it in the request.

Important

Verify the signature is URL encoded **only once**. A common mistake is to URL encode it manually during signature formation, and then again when the toolkit URL encodes the entire request.

For examples of signed requests, see [Example REST Requests \(p. 55\)](#).

Java Sample Code for Calculating Signature Version 2 Signatures

The following Java code sample shows how to calculate a Signature Version 2. This sample has code for creating the canonical string, for base64 encoding, and for HMAC encoding for generating the signature.

```
package com.amazon.associates.sample;

import java.io.UnsupportedEncodingException;
import java.net.URLDecoder;
import java.net.URLEncoder;
import java.security.InvalidKeyException;
import java.security.NoSuchAlgorithmException;
```

```
import java.text.DateFormat;
import java.text.SimpleDateFormat;

import java.util.Calendar;
import java.util.HashMap;
import java.util.Iterator;
import java.util.Map;
import java.util.SortedMap;
import java.util.TimeZone;
import java.util.TreeMap;

import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;

import org.apache.commons.codec.binary.Base64;

public class SignedRequestsHelper {
    private static final String UTF8_CHARSET = "UTF-8";
    private static final String HMAC_SHA256_ALGORITHM = "HmacSHA256";
    private static final String REQUEST_URI = "/onca/xml";
    private static final String REQUEST_METHOD = "GET";

    private String endpoint = "webservices.amazon.com"; // must be lowercase
    private String awsAccessKeyId = "YOUR AWS ACCESS KEY";
    private String awsSecretKey = "YOUR AWS SECRET KEY";

    private SecretKeySpec secretKeySpec = null;
    private Mac mac = null;

    public SignedRequestsHelper() {
        byte[] secretKeyBytes = awsSecretKey.getBytes(UTF8_CHARSET);
        secretKeySpec =
            new SecretKeySpec(secretKeyBytes, HMAC_SHA256_ALGORITHM);
        mac = Mac.getInstance(HMAC_SHA256_ALGORITHM);
        mac.init(secretKeySpec);
    }

    public String sign(Map<String, String> params) {
        params.put("AWSAccessKeyId", awsAccessKeyId);
        params.put("Timestamp", timestamp());

        SortedMap<String, String> sortedParamMap =
            new TreeMap<String, String>(params);
        String canonicalQS = canonicalize(sortedParamMap);
        String toSign =
            REQUEST_METHOD + "\n"
            + endpoint + "\n"
            + REQUEST_URI + "\n"
            + canonicalQS;

        String hmac = hmac(toSign);
        String sig = percentEncodeRfc3986(hmac);
        String url = "http://" + endpoint + REQUEST_URI + "?" +
        canonicalQS + "&Signature=" + sig;

        return url;
    }
}
```

```
private String hmac(String stringToSign) {
    String signature = null;
    byte[] data;
    byte[] rawHmac;
    try {
        data = stringToSign.getBytes(UTF8_CHARSET);
        rawHmac = mac.doFinal(data);
        Base64 encoder = new Base64();
        signature = new String(encoder.encode(rawHmac));
    } catch (UnsupportedEncodingException e) {
        throw new RuntimeException(UTF8_CHARSET + " is unsupported!", e);
    }
    return signature;
}

private String timestamp() {
    String timestamp = null;
    Calendar cal = Calendar.getInstance();
    DateFormat dfm = new SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ss'Z'");
    dfm.setTimeZone(TimeZone.getTimeZone("GMT"));
    timestamp = dfm.format(cal.getTime());
    return timestamp;
}

private String canonicalize(SortedMap<String, String> sortedParamMap)
{
    if (sortedParamMap.isEmpty()) {
        return "";
    }

    StringBuffer buffer = new StringBuffer();
    Iterator<Map.Entry<String, String>> iter =
        sortedParamMap.entrySet().iterator();

    while (iter.hasNext()) {
        Map.Entry<String, String> kvpair = iter.next();
        buffer.append(percentEncodeRfc3986(kvpair.getKey()));
        buffer.append("=");
        buffer.append(percentEncodeRfc3986(kvpair.getValue()));
        if (iter.hasNext()) {
            buffer.append("&");
        }
    }
    String canonical = buffer.toString();
    return canonical;
}

private String percentEncodeRfc3986(String s) {
    String out;
    try {
        out = URLEncoder.encode(s, UTF8_CHARSET)
            .replace("+", "%20")
            .replace("*", "%2A")
            .replace("%7E", "~");
    } catch (UnsupportedEncodingException e) {
        out = s;
    }
}
```

```
        return out;
    }
}
```

Example REST Requests

This section shows the steps to sign a request with example AWS credentials.

- **AWS access key ID:** AKIAIOSFODNN7EXAMPLE
- **AWS secret key:** 1234567890

Follow these steps with these credentials to generate the same signature strings in the examples. This can help verify your request generation code. You cannot make actual requests with these example requests.

Tip

If you aren't familiar with REST requests, see [Anatomy of a REST Request \(p. 41\)](#) before continuing with this example.

The following is an example of an [ItemLookup \(p. 197\)](#) request:

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService&AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Operation=ItemLookup&ItemId=0679722769&ResponseGroup=ItemAttributes,Offers,Images,Reviews&Version=2013-08-01
```

Steps to Sign the Example Request

1. Enter the time stamp. For this example, we'll use the UTC time 2014-08-18T12:00:00Z.

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService&AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&AssociateTag=mytag-20&Operation=ItemLookup&ItemId=0679722769&ResponseGroup=Images,ItemAttributes,Offers,Reviews&Version=2013-08-01&Timestamp=2014-08-18T12:00:00Z
```

2. URL encode the request's comma (,) and colon (:) characters, so that they don't get misinterpreted. For more information about converting to RFC 3986 specifications, see documentation and code samples for your programming language.

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService&AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&AssociateTag=mytag-20&Operation=ItemLookup&ItemId=0679722769&ResponseGroup=Images%2CItemAttributes%2COffers%2CReviews&Version=2013-08-01&Timestamp=2014-08-18T12%3A00%3A00Z
```

Important

Do not double-escape any characters.

3. Split the parameter/value pairs and delete the ampersand characters (&). The linebreaks used in the following example follow Unix convention (ASCII 0A, "line feed" character).

```
Service=AWSECommerceService
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE
```

```
AssociateTag=mytag-20
Operation=ItemLookup
ItemId=0679722769
ResponseGroup=Images%2CItemAttributes%2COffers%2CReviews
Version=2013-08-01
Timestamp=2014-08-18T12%3A00%3A00Z
```

4. Sort your parameter/value pairs by byte value (not alphabetically, lowercase parameters will be listed after uppercase ones).

```
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE
AssociateTag=mytag-20
ItemId=0679722769
Operation=ItemLookup
ResponseGroup=Images%2CItemAttributes%2COffers%2CReviews
Service=AWSECommerceService
Timestamp=2014-08-18T12%3A00%3A00Z
Version=2013-08-01
```

5. Rejoin the sorted parameter/value list with ampersands. The result is the canonical string that we'll sign:

```
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&AssociateTag=mytag-
20&ItemId=0679722769&Operation=ItemLookup&ResponseGroup=Images%2CItemAttrib-
utes%2COffers%2CReviews&Service=AWSECommerceService&Timestamp=2014-08-
18T12%3A00%3A00Z&Version=2013-08-01
```

6. Prepend the following three lines (with line breaks) before the canonical string:

```
GET
webservices.amazon.com
/onca/xml
```

7. The string to sign:

```
GET
webservices.amazon.com
/onca/xml
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&AssociateTag=mytag-
20&ItemId=0679722769&Operation=ItemLookup&ResponseGroup=Images%2CItemAttrib-
utes%2COffers%2CReviews&Service=AWSECommerceService&Timestamp=2014-08-
18T12%3A00%3A00Z&Version=2013-08-01
```

8. Calculate an RFC 2104-compliant HMAC with the SHA256 hash algorithm using the string above with this example AWS secret key: 1234567890. For more information about this step, see documentation and code samples for your programming language.

```
j7bZM0LXZ9eXeZruTqWm2DlVdYVUU3wxPPpp+iXxzQc=
```

9. URL encode the plus (+) and equal (=) characters in the signature:

```
j7bZM0LXZ9eXeZruTqWm2DlVvDYVUU3wxPPpp%2BiXxzQc%3D
```

10. Add the URL encoded signature to your request, and the result is a properly-formatted signed request:

```
http://webservices.amazon.com/onca/xml?AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&AssociateTag=mytag-20&ItemId=0679722769&Operation=ItemLookup&ResponseGroup=Images%2CItemAttributes%2COffers%2CReviews&Service=AWSECommerceService&Timestamp=2014-08-18T12%3A00%3A00Z&Version=2013-08-01&Signature=j7bZM0LXZ9eXeZruTqWm2DlVvDYVUU3wxPPpp%2BiXxzQc%3D
```

Examples of Other Signed Requests

The previous section went through a detailed process for signing an `ItemLookup` request. This section presents examples of unsigned requests, the string to sign generated from them, and the final signed request. These examples can make excellent test cases for your software. All examples use the example AWS secret key 1234567890 used earlier in the [Steps to Sign the Example Request \(p. 55\)](#).

ItemSearch Examples

The following examples show the [ItemSearch \(p. 185\)](#) request without a signature, with the string to sign, and the signed request.

Original Unsigned Request

```
http://webservices.amazon.co.uk/onca/xml?Service=AWSECommerceService&Operation=ItemSearch&AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Operation=ItemSearch&Actor=Johnny%20Depp&ResponseGroup=ItemAttributes,Offers,Images,Reviews,Variations&Version=2013-08-01&SearchIndex=DVD&Sort=salesrank&AssociateTag=mytag-20
```

String to Sign

```
GET  
webservices.amazon.co.uk  
/onca/xml  
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Actor=Johnny%20Depp&AssociateTag=mytag-  
20&Operation=ItemSearch&Operation=ItemSearch&ResponseGroup=ItemAttributes%2COffers%2CImages%2CReviews%2CVariations&SearchIndex=DVD&Service=AWSECommerceService&Sort=salesrank&Timestamp=2014-08-18T17%3A34%3A34.000Z&Version=2013-08-01
```

Signed Request

```
http://webservices.amazon.co.uk/onca/xml?AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Actor=Johnny%20Depp&AssociateTag=mytag-20&Operation=ItemSearch&Operation=ItemSearch&ResponseGroup=ItemAttributes%2COffers%2CImages%2CReviews%2CVariations&SearchIndex=DVD&Service=AWSECommerceService&Sort=salesrank&Timestamp=2014-08-18T17%3A34%3A34.000Z&Version=2013-08-01&Signature=Gv4kWyAAD3xgSGI86I4qZlzi  
jAhZYs2H7CRTpeHLD1o%3D
```

CartCreate Examples

The following examples show the [CartCreate \(p. 214\)](#) request without a signature, with the string to sign, and the signed request.

Original Unsigned Request

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService&Operation=ItemSearch&AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Operation=CartCreate&Version=2013-08-01&Item.1.OfferListingId=j8ejq9wxDfSYWf20Cp6XQGDsVrWhl08GSQ9m5j%2Be8MS449BN1XGUC3DfU5Zw4nt%2FFBt87cspLowlQXzfvZpvzg%3D%3D&Item.1.Quantity=3&AssociateTag=mytag-20
```

String to Sign

```
GET  
webservices.amazon.com  
/onca/xml  
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&AssociateTag=mytag-20&Item.1.OfferListingId=j8ejq9wxDfSYWf20Cp6XQGDsVrWhl08GSQ9m5j%2Be8MS449BN1XGUC3DfU5Zw4nt%2FFBt87cspLowlQXzfvZpvzg%3D%3D&Item.1.Quantity=3&Operation=CartCreate&Operation=ItemSearch&Service=AWSECommerceService&Timestamp=2014-08-18T17%3A36%3A55.000Z&Version=2013-08-01
```

Signed Request

```
http://webservices.amazon.com/onca/xml?AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&AssociateTag=mytag-20&Item.1.OfferListingId=j8ejq9wxDfSYWf20Cp6XQGDsVrWhl08GSQ9m5j%2Be8MS449BN1XGUC3DfU5Zw4nt%2FFBt87cspLowlQXzfvZpvzg%3D%3D&Item.1.Quantity=3&Operation=CartCreate&Operation=ItemSearch&Service=AWSECommerceService&Timestamp=2014-08-18T17%3A36%3A55.000Z&Version=2013-08-01&Signature=LpEUnc9tT4WGneeUwH0LvwxxLLfbMEXgmjGX5GXQ1MEQ%3D
```

BrowseNodeLookup Examples

The following examples show the [BrowseNodeLookup \(p. 194\)](#) request without a signature, with the string to sign, and the signed request.

Original Unsigned Request

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService&Operation=ItemSearch&AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Operation=BrowseNodeLookup&Version=2013-08-01&BrowseNodeId=465600&AssociateTag=mytag-20&ResponseGroup=BrowseNodeInfo,TopSellers,NewReleases,MostWishedFor,MostGifted
```

String to Sign

```
GET  
webservices.amazon.com  
/onca/xml  
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&AssociateTag=mytag-20&BrowseNodeId=465600&Operation=BrowseNodeLookup&Operation=ItemSearch&ResponseGroup=BrowseNodeInfo%2CTopSellers%2CNewReleases%2CMostWishedFor%2CMostGifted&Service=AWSECommerceService&Timestamp=2014-08-18T17%3A38%3A12.000Z&Version=2013-08-01
```

Signed Request

```
http://webservices.amazon.com/onca/xml?AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&AssociateTag=mytag-20&BrowseNodeId=465600&Operation=BrowseNodeLookup&Operation=ItemSearch&ResponseGroup=BrowseNodeInfo%2CTopSellers%2CNewReleases%2CMostWishedFor%2CMostGifted&Service=AWSECommerceService&Timestamp=2014-08-18T17%3A38%3A12.000Z&Version=2013-08-01&Signature=t48XyuQKlcYROCm7w%2FNqo3mihqB%2FQF2B9b9SX3FIOnU%3D
```

SimilarityLookup Examples

The following examples show the [SimilarityLookup \(p. 203\)](#) request without a signature, with the string to sign, and the signed request.

Original Unsigned Request

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService&Operation=ItemSearch&AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&Operation=SimilarityLookup&ItemId=B0011ZK6PC,B000NK8EWI&Version=2013-08-01&AssociateTag=mytag-20&ResponseGroup=Offers,ItemAttributes&SimilarityType=Intersection&Condition>New&Merchant=Amazon
```

String to Sign

```
GET  
webservices.amazon.com  
/onca/xml  
AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&AssociateTag=mytag-20&Condition>New&ItemId=B0011ZK6PC%2CB000NK8EWI&Merchant=Amazon&Operation=ItemSearch&Operation=SimilarityLookup&ResponseGroup=Offers%2CItemAttributes&Service=AWSECommerceService&SimilarityType=Intersection&Timestamp=2014-08-18T17%3A39%3A22.000Z&Version=2013-08-01
```

Signed Request

```
http://webservices.amazon.com/onca/xml?AWSAccessKeyId=AKIAIOSFODNN7EXAMPLE&AssociateTag=mytag-20&Condition>New&ItemId=B0011ZK6PC%2CB000NK8EWI&Merchant=Amazon&Operation=ItemSearch&Operation=SimilarityLookup&ResponseGroup=Offers%2CItemAttributes&Service=AWSECommerceService&SimilarityType=Intersection&Timestamp=2014-08-18T17%3A39%3A22.000Z&Version=2013-08-01&Signature=nIlF7C6O1T3faoXIZgGVxYXd%2BD%2F39%2BFPSnwdfiQvy9g%3D
```

Authenticating SOAP Requests

Topics

- [Using SOAP without WS-Security \(p. 60\)](#)

The Product Advertising API accepts SOAP requests sent over an HTTPS connection only. You can authenticate SOAP requests with your AWS access key ID.

Using SOAP without WS-Security

This section describes how to authenticate SOAP requests without using WS-Security. The topics describe the basic requirements, the required authentication information, and where to place the information in the SOAP request.

General Requirements

If you plan to use SOAP without WS-Security:

- You can use SOAP 1.1 or SOAP 1.2.
- You must use HTTPS with your requests.

Required Authentication Information

Authentication of SOAP requests without WS-Security uses your AWS identifiers and an HMAC-SHA256 signature. The request must include the parameters listed in the following table.

Parameter	Description
AWSAccessKeyId	Your AWS access key ID. For more information, see Managing Your AWS Credentials (p. 49) .
Timestamp	This is a required parameter if you include the Signature parameter. Otherwise, it is optional. There is no default value. The time stamp you use in the request must be a <code>dateTime</code> object, with the complete date plus hours, minutes, and seconds. This is a fixed -length subset of the format defined by ISO 8601, represented in Universal Time (GMT): <code>YYYY-MM-DDThh:mm:ssZ</code> (where T and Z are literals). For more information, see Date and Time Formats . Important If you are using .NET you must not send overly specific time stamps, due to different interpretations of how extra time precision should be dropped. To avoid overly specific time stamps, manually construct <code>dateTime</code> objects with no more than millisecond precision.
Signature	The HMAC-SHA256 signature calculated from the concatenation of the Action and Timestamp parameters, using your AWS secret access key. For example, for a request to create a queue, the value of the Signature element would be the HMAC-SHA256 digest of a string like this: <code>ItemLookup2014-09-24T00:00:00Z</code> For more information about authentication with HMAC signatures, see HMAC-SHA256 Signatures for REST Requests (p. 49) .

To calculate the signature

1. Concatenate the values of the Action and Timestamprequest parameters, in that order.

The string you've just created is the string you'll use when generating the signature.
2. Calculate an RFC 2104-compliant HMAC-SHA256 signature, using the string you just created and your secret access key as the key.
3. Convert the resulting value to base64.
4. Pass this final value in the Signature parameter of the SOAP request.

Location of Authentication Information in the Request

With version 2013-08-01, you must provide the authentication information as elements in the SOAP header (using the namespace `http://security.amazonaws.com/doc/2007-01-01/`), as in the following example.

```
<?xml version="1.0"?>
<soap:Envelope
    xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
    soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">

<soap:Header
    xmlns:aws="http://security.amazonaws.com/doc/2007-01-01/">
    <aws:AWSAccessKeyId>AKIAIOSFODNN7EXAMPLE</aws:AWSAccessKeyId>
    <aws:Timestamp>2008-02-10T23:59:59Z</aws:Timestamp>
    <aws:Signature>SZf1CHmQnrZbsrC13hCZS06lywsEXAMPLE</aws:Signature>
</soap:Header>
...
</soap:Envelope>
```

Batch Requests

Topics

- [Parameters That Differ \(p. 62\)](#)
- [Shared Parameters \(p. 63\)](#)
- [Performing Multiple ItemLookups in One Request \(p. 63\)](#)

The requests presented so far contain only one operation. Product Advertising API enables you to improve performance by submitting more than one request at the same time. There are two ways to do this:

Batch request

A request that uses one operation with up to two sets of parameters.

Multiple ItemIds

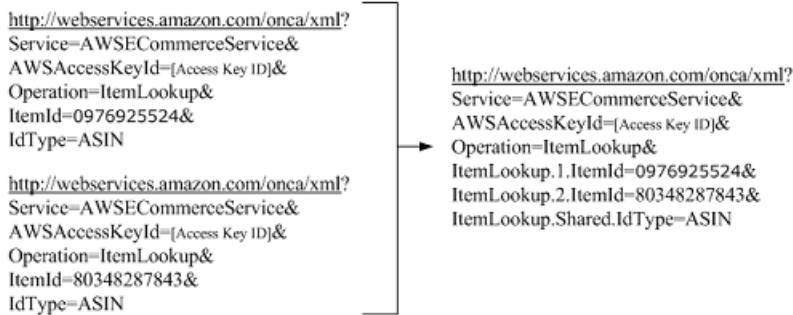
In an `ItemLookup` request, you can include up to ten comma-separated values for `ItemId`.

Note

Batch requests are not supported for the cart operations: `CartAdd`, `CartClear`, `CartCreate`, `CartGet`, and `CartModify`.

There may be occasions when you want to use the same operation in two requests, each one using different input parameters. For example, you might want to run an `ItemLookup` request several times, each time looking up a different item. To accomplish this task, you can submit two requests or one batch request.

The following figure shows how two simple requests can be combined into one batch request.



The new syntax introduced in the batch request centers on the ways in which the operation parameters are defined. Either the parameter values are different or they are shared between the simple requests in the batch request.

Batch requests can contain up to two sets of parameters.

Note

The Product Advertising API supports `ItemLookup` batch requests only when the `ItemType` is an EAN (European Article Number), ISBN (International Standard Book Number), EISBN (Electronic International Standard Book Number), or ASIN (Amazon Standard Item Number).

Parameters That Differ

Parameters that differ between the simple requests in a batch request are identified by a reference number. The following parameter declaration syntax shows where the reference number goes.

```
OperationName.ReferenceNumber.Parameter=Value
```

The following request snippet shows an example of a reference number, 1.

```
ItemLookup.1.ItemId=0976925524
```

The `ReferenceNumber` is a positive integer that associates the parameters in the simple request. For example, the following parameters are part of the same simple request because they use the same reference number.

```
ItemLookup.1.ItemId=0976925524&  
ItemLookup.1.ResponseGroup=Images
```

These parameters may be different from a second simple request, which is part of the same batch request.

```
ItemLookup.2.ItemId=0485935524&  
ItemLookup.2.ResponseGroup=Collections
```

So, if the previous two request snippets were part of one batch request, the `ItemLookup` operation is executed twice, each time with a different `ItemId` and `ResponseGroup`.

In one batch request, you can have up to two different reference numbers. For example, the following request snippet exceeds the number of allowed reference numbers in one batch request.

```
ItemLookup.1.ItemId=0976925524&  
ItemLookup.2.ItemId=0485935524&
```

```
ItemLookup.3.ItemId=0792335535&
```

Shared Parameters

There are times when the parameter values for the simple requests in a batch request are the same, for example, `ItemType=ASIN`. In this case, instead of using a `ReferenceNumber` for each simple request, you can substitute the special value, "Shared", as shown in the following example.

```
ItemLookup.Shared.ItemType=ASIN
```

The following parameter declarations show the equivalent of the preceding declaration.

```
ItemLookup.1.ItemType=ASIN&
ItemLookup.2.ItemType=ASIN
```

Both forms of the request work but using the special value, `Shared`, reduces the amount of typing required. Generally, you only use the `Shared` value with required parameters. Optional parameters do not need to be included in the request and their default values, if any, are assumed and thus shared across the simple requests in the batch request.

Performing Multiple ItemLookups in One Request

The other way to execute multiple `ItemLookup` requests in one request is to use a comma-separated list of `ItemIds`, for example:

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate ID]&
Operation=ItemLookup&
ItemId=B00008OE6I,B35987036I,B0002546I,B25468OE6I,B09788OE6I,B00453OE6I&
IdType=ASIN&
ResponseGroup=OfferFull&
Condition=All&
Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The example shows six `ItemIds` in the request. `ItemLookup` will be executed six times, one for each `ItemId`. You can specify up to ten `ItemIds`.

Tip

Be careful not to add spaces before or after the commas in the comma-separated list of `ItemIds`.

Parameters Common to All Product Advertising API Requests

Topics

- [Required Parameters \(p. 64\)](#)
- [Optional Parameters \(p. 64\)](#)
- [XML-Encoding Parameters \(p. 65\)](#)

- [Debugging Parameters \(p. 66\)](#)

Product Advertising API operations use a variety of required and optional parameters. For descriptions of each operation and available parameters, see the [API Reference \(p. 184\)](#).

Some parameters are available for all operations. These parameters can be grouped into the following types.

Required Parameters

These parameters are required in every request.

Parameter	Value	Description
Service	AWSECommerceService	Specifies the Product Advertising API service.
AWSAccessKeyId	Your AWS access key ID.	Every Product Advertising API request must contain an access key ID. To retrieve your AWS access key ID, see Becoming a Product Advertising API Developer (p. 5) .
AssociateTag	Amazon Associate Tag or ID.	The AssociateTag enables item URLs returned by Product Advertising API to be tagged as originating from your Associates website. For your request to succeed and to receive a referral credit for a sale, you must include an AssociateTag value in all requests. Verify the AssociateTag is correctly added. No error is returned for incorrect values.
Operation	Operation you want to perform, for example, ItemLookup.	One of the Product Advertising API operation types.

AssociateTag Parameter

An AssociateTag is an alphanumeric token distributed by Amazon that uniquely identifies an Associate. Amazon uses this ID to credit an Associate for a sale. The AssociateTag parameter becomes part of the PurchaseURL, which is the URL used to purchase the items in a remote shopping cart.

Important

You must include an AssociateTag value in all requests to the Product Advertising API. If a request does not have an AssociateTag value, Product Advertising API returns an error.

For information about how to get an AssociateTag, see [Becoming an Associate \(p. 4\)](#).

Optional Parameters

These parameters are optional for Product Advertising API requests.

Product Advertising API Developer Guide
Parameters Common to All Product Advertising API Requests

Parameter	Value	Description
MerchantId	A parameter used to filter search results and offer listings to only include items sold by Amazon. By default, the Product Advertising API will return items sold by various merchants including Amazon.	The only valid optional value is "Amazon".
Response Group	Various	Specifies what subset of data to return. For a list of all Response Groups, see Response Groups (p. 230) .
Version	Various	Version of the Product Advertising API WSDL. The default is 2013-08-01. If you want another version, you must specify it in your request.

XML-Encoding Parameters

You may find this parameter useful in some environments.

Parameter	Value	Description
XMLEscaping	"Single" or "Double".	XMLEscaping specifies whether responses are XML-encoded in a single pass or a double pass. By default, XMLEscaping is "Single" and Product Advertising API responses are encoded only once in XML. For example, if the response data includes an ampersand character (&), the character is returned in its regular XML encoding (&). If XMLEscaping is "Double", the same ampersand character is XML-encoded twice (&). The Double value for XMLEscaping is useful in some clients, such as PHP, that do not decode text within XML elements.

Debugging Parameters

Parameter	Value	Description
Validate	Boolean	<p>Use the <code>Validate</code> parameter to have Product Advertising API test your request without actually executing it. When present, <code>Validate</code> must equal "True".</p> <p>If the request is valid, the response will have an element called <code>IsValid</code> with a value of "True". If the request is invalid, the response returns <code>IsValid</code> with a value of "False" and errors messages.</p> <p>Note Since the request is not actually executed, only a subset of the errors for the request may be returned. This is because some errors (e.g., <code>no_exact_matches</code>) are only generated during execution of a request.</p>

We recommend you see the `Request` response group when debugging. The `Request` response group echoes back the parameters used to generate the response and allows you to verify if your request was received correctly.

Responses

Topics

- [Anatomy of a Response \(p. 67\)](#)
- [Version Information Segment \(p. 67\)](#)
- [Operation Request Segment \(p. 68\)](#)
- [Request Validation Segment \(p. 68\)](#)
- [Item Attributes Segment \(p. 70\)](#)

When the Product Advertising API receives a request, it returns an XML document that contains the results of the call in an XML data structure. This data conforms to a WSDL and schema.

For REST requests, this data structure is simply the body of the HTTP response. You can use a data binding method for REST responses, or use an XML parser directly to process the information.

For SOAP requests, the data structure is the SOAP message body of the response. SOAP toolkits typically convert the response data into structures for use with your programming language, or allow you to specify your own data bindings.

Other than the use of a message envelope in the SOAP response, REST and SOAP responses are identical. Both conform to the Product Advertising API schema. The SOAP WSDL imports an XSD file to define the response messages. REST users can access the XSD file directly. For more information, see [WSDL Location \(p. 18\)](#).

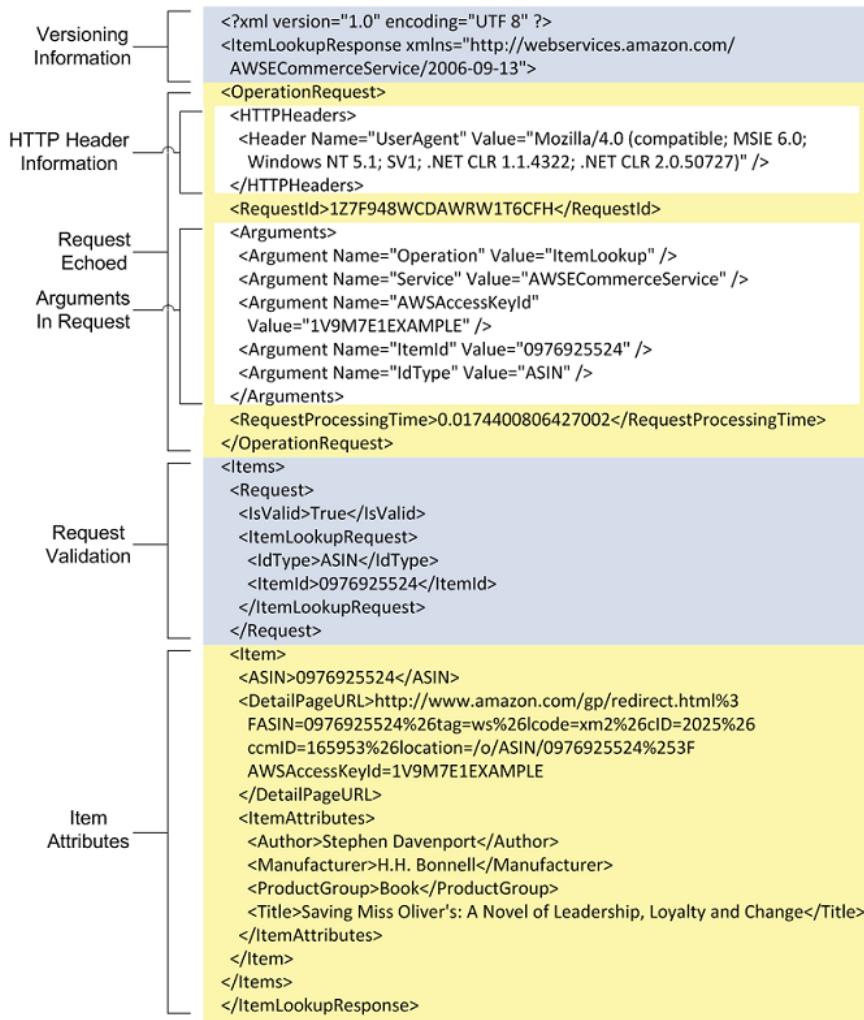
Important

All requests that you submit must be authenticated. For information about authenticating REST requests, see [Authenticating REST Requests \(p. 51\)](#).

For information about authenticating SOAP requests, see [Authenticating SOAP Requests \(p. 59\)](#).

Anatomy of a Response

Product Advertising API responses vary according to the operations and parameters submitted in the request. There are parts of the response that are common across all responses, as shown in the following figure.



The following sections describe each segment in this response.

Version Information Segment

The first two lines of every response contain version information. The first line details the XML version used. In this example, the version is 1.0. The first line also includes the character encoding. In this example, it's UTF-8.

The second line of every response has the name of the response, which includes the name of the operation of the request. In this example, the `ItemLookup` operation returns a response element named `ItemLookupResponse`.

The second line also contains the version of Product Advertising API used to create the response. If a version is not specified in a request, Product Advertising API uses the default and latest version, 2013-08-01.

Operation Request Segment

The second segment of a response is always the Operation Request. For the most part, it returns the operation's parameters, called arguments, that were used in the request.

The first large subsection is the HTTP Header section, which specifies the network agent used to send the request. Typically, the agent, called the user agent, is a web browser and the information returned describes that browser, such as its name, version, and the operating system of its host.

The second subsection contains the arguments, or operation parameters, used in the request. These values are unprocessed echoes of the values submitted in the request. This information is echoed for the benefit of troubleshooting requests.

Finally, in the Operation Request segment there is Product Advertising API processing information, including the Request ID and the time it took to process the request. This is good information for troubleshooting. You will not have to use either the Request ID or the processing time in future requests.

Request Validation Segment

The Request Validation segment has error information if there was an error in the request. It is also the last segment that Product Advertising API returns in a response if there was an error.

In the preceding example, the request was valid, as shown in the `IsValid` element.

```
<IsValid>True</IsValid>
```

The `IsValid` element pertains only to the validity of the request. It reports if all of the required elements of a request are present, parameter restrictions and constraints have been obeyed, and parameter values are legal. `IsValid` does not assure that a valid request will be processed.

For example, an `ItemId` such as 1234ABCD is in the correct form, but no items in Amazon's catalog match that `ItemId`. The request is valid, but will return an error.

There are many reasons why a request can fail.

Problem	Description
Incorrect parameters	<p>The request is missing parameters that an operation requires, or the combination of parameters is not allowed, or the <code>Service</code> parameter was omitted.</p> <p>In this case, the <code>IsValid</code> value is "False" and the request is not processed. Common mistakes that cause these errors are spaces in the URL that prevent Product Advertising API from reading the entire request, or misspelled and incorrectly capitalized parameter names or values.</p> <p>If you inadvertently enter a space in a request, your browser automatically enters the URL-encoded form of a space: <code>%20</code>. For example, if you paste, <code>ItemId= B12345</code>, in your browser as part of a request, the URL field in the browser shows, <code>ItemId=%20B12345</code>.</p>

Problem	Description
Invalid values	The request may be syntactically correct, but the values for one or more parameters may be invalid. For example, you might supply an <code>ItemId</code> value that does not match any item sold on Amazon. In this case, the <code>IsValid</code> value is "True", but an error message is returned that says that the value is not valid.
Empty result	The request is valid, but there is no data that matches the search criteria. In this case, the <code>IsValid</code> value is "True". Remember that parameter names and values are case sensitive.
Network or processing error	The request is valid, but a network outage or processing failure has caused Product Advertising API to return an empty response. In this case, the <code>IsValid</code> value is "True".
Partial request errors	Product Advertising API returns as much information as possible. For example, if you use a batch request and Product Advertising API finds items for one request but not the other, Product Advertising API returns all of the information for the found item and an error message for the other request.

When one of these errors occurs, the `IsValid` value is "False" and an error element is returned in the Request Validation segment. The error element includes:

- **Error code** — The title of the error.
- **Error message** — A brief description of the error.

The following is an example error message.

```
<Errors>
<Error>
  <Code>missing_service_parameter</code>
  <message>Your request is missing the Service parameter. Please add the Service parameter to your request and retry. Valid values for the Service parameter include AWSEcommerceService.</message>
</Error>
</Errors>
```

The more difficult error to assess is a request that is valid and processed, but contains an error in logic. For example, a request might use default values such that only items in new condition are returned when the requester meant to return items in all conditions.

The remainder of the Request Validation segment is a subsection named after the operation used in the request. In this example, the subsection is named `ItemLookup` Request. This subsection echoes the parameter values used in the request. These values are returned for the purpose of troubleshooting the request.

For a list of all error codes and messages, see [Error Messages \(p. 173\)](#).

Item Attributes Segment

If your request is valid, the final segment of the response has the item attributes. This segment is typically the part of the response that contains the most value. This segment contains all of the descriptive information about the items that satisfied the request.

In the previous example, one item was found that satisfied the search criteria. Many attributes are returned that describe that item, including, the title, "Saving Miss Oliver's: A Novel of Leadership, Loyalty and Change".

Item attributes returned in a response vary according to the response group used and the items found. Merchants do not always provide Amazon with a single set of item attributes. For example, one merchant might supply the name of a book's publisher, while another won't. A response with both books will have a different set of item attributes.

Item attributes that do not have values are not returned. For more information for item attributes that can be returned, see [Response Groups \(p. 230\)](#).

Paging and Sorting Through Responses

Topics

- [Paging Through Results \(p. 70\)](#)
- [Maximum Number Of Returned Pages \(p. 71\)](#)
- [Sorting Results \(p. 71\)](#)
- [Default Sort Values \(p. 72\)](#)

A constraint of having so many items is the possibility of receiving too many in a response. Product Advertising API handles this problem in several ways:

- Results are returned on page, generally, up to ten results per page.
- The `Sort` parameter orders results.

Paging Through Results

You can create a request that returns thousands of items in a response, but can be problematic. Returning all of the item attributes for those items will negatively impact the performance of Product Advertising API. Also, posting a thousand responses on a web page is impractical.

The Product Advertising API developed the strategy of returning results a little at a time. You can return any page of results. For example, the first request can return the last page of results. To do so, specify the desired page of results using one of the parameters that enable you to return result pages.

To page through results

Use the appropriate paging parameter in the request. Operations have their own paging parameters. For example, the following `ItemSearch` request uses `ItemPage` to ask for the fourth page of results.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&
```

```
Keywords=Potter&
SearchIndex=Books&
ItemPage=4
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The following response snippet shows the fourth page of results is returned.

```
<ItemSearchRequest>
  <ItemPage>4</ItemPage>
  <Keywords>Potter</Keywords>
  <SearchIndex>Books</SearchIndex>
</ItemSearchRequest>
</Request>
<TotalResults>9729</TotalResults> <TotalPages>973</TotalPages>
```

This example shows that 9729 items matched the search criteria. Also, it shows that those results are on 973 (~9729 / 10) pages. You might try putting in an `ItemPage` value over 10. If you do, Product Advertising API returns the following error.

```
<Error>
  <Code>AWS.ParameterOutOfRange</Code>
  <Message>The value you specified for ItemPage is invalid. Valid values must be between 1 and 10.</Message>
</Error>
```

So, how do you get that 973rd page? You cannot. A better approach is to submit a new request that is more targeted and returns fewer items in the response.

Maximum Number Of Returned Pages

The previous example showed that Product Advertising API returns only so many pages for any one request. This is because performance must be optimized for the tens of thousands of Product Advertising API developers and customers. Many Product Advertising API operations have pagination parameters and associated maximum values, as shown in the following table.

Operation	Parameter Name	Maximum Page Number
ItemLookup	VariationPage	150
ItemSearch	ItemPage	10

Sorting Results

The `ItemSearch` operation has a `Sort` parameter that arranges results.

To sort results

1. To determine available sort values for your locale, see [Locale Reference \(p. 328\)](#).
2. Add the `Sort` parameter to a request that uses one of the previous operations.

For example, the following request returns books with "Harry Potter" in the title or description in alphabetical order.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
Keywords=Harry%20Potter&  
SearchIndex=Books&  
Sort=titlerank&  
ItemPage=4&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

A small snippet of the response shows that the book titles are returned in alphabetical order.

```
<Title>Aventuras de Harry Potter, Las</Title>  
...  
<Title>Beacham's Sourcebook For Teaching Young Adult Fiction: Exploring Harry  
Potter</Title>  
...  
<Title>Beatrix Potter to Harry Potter: Portraits of Children's Writers</Title>
```

Product Advertising API provides many different sorting criteria, for example, price (high to low, or low to high), salesrank (best to worst selling, or worst to best selling), publication date, review rank, and release date.

Available sort values vary by locale and search index. For example, the DigitalMusic search index can be sorted by UploadedDate. That same value will not make sense in the Automotive search index.

Default Sort Values

There are many sort values and the majority are not applied unless the `Sort` parameter is in the request. There are two sort values that are default.

- For `ItemSearch` requests that do not use the `BrowseNode` parameter, results are sorted by Relevance.
- For `ItemSearch` requests that do use the `BrowseNode` parameter, results are sorted by BestSeller ranking.

Find Items to Buy

Topics

- [Search Operations \(p. 73\)](#)
- [Items You Cannot Buy \(p. 73\)](#)
- [Items Sold by Merchants \(p. 75\)](#)
- [Common ItemSearch Parameters \(p. 76\)](#)
- [Power Searches \(p. 81\)](#)
- [Use Search Bins to Find Items \(p. 85\)](#)
- [Find Items with Browse Nodes \(p. 91\)](#)
- [Request and Response Examples \(p. 95\)](#)

In the previous chapter, you learned how Amazon groups and catalogs items for sale. In the following sections, learn how to use Product Advertising API operations to find those items.

Search Operations

The following table describes the Product Advertising API operations that find items.

Operation	Description
ItemSearch (p. 185)	To find items sold on Amazon. This operation is commonly used to find items for sale.
BrowseNodeLookup (p. 194)	To find items associated with browse nodes by navigating through the browse node hierarchy. This operation does not return items directly, only the browse nodes associated with items.

If you do not know whether an item is for sale by a merchant or seller, search first using `ItemSearch` for two reasons:

- Most of the items sold on Amazon are sold by merchants.
- Sellers often list items for sale in Amazon's marketplace as well as in their own stores. Anything sold in the marketplace can be found by `ItemSearch`.

Items You Cannot Buy

You might assume that any item returned by one of the search operations is for sale. This, however, is not the case. Some items, such as Variation parents and Collection parents, are never for sale. Before presenting items to customers, verify that the items can be purchased.

Product Advertising API provides several ways to determine if an item can be purchased, for example, `ItemSearch` has an *Availability* parameter, which specifies whether an item is available for purchase. The best way, however, to determine whether or not an item can really be purchased is by using one of the following response groups in a request:

- [Offers Response Group \(p. 271\)](#)
- [OfferFull Response Group \(p. 267\)](#)
- [Large Response Group \(p. 254\)](#)

Note

Large is a parent of the Offers response group.

All of the search operations listed, except `BrowseNodeLookup`, can use these response groups. These response groups return an `OfferListingId` element for each item that can be purchased. Items that do not have an `OfferListingId` cannot be purchased.

The following response snippet shows that an item can be purchased because the response contains an `OfferListingId`.

```
<Item>
  ...
  <Offer>
    ...
    <OfferListing>
```

```
<OfferListingId>g7CWSnBZmVESKy%2BCNS</OfferListingId>
<Price>
  <Amount>625</Amount>
  <CurrencyCode>USD</CurrencyCode>
  <FormattedPrice>$6.25</FormattedPrice>
</Price>
<Availability>Usually ships in 1-2 business days
</Availability>
</OfferListing>
</Offer>
</Item>
```

This response shows how the `OfferListingId` element is a child of the `Item` element.

The `Availability` element in the preceding example is not a reliable means of determining whether or not an item is truly available for purchase. Its message concerns how quickly an item can ship when it is in stock. A value returned for `Availability` does not guarantee that the item is in stock.

Tip

The next chapter talks about adding items to a customer's shopping cart. While it is possible to add an item to a cart using an ASIN, the preferred means of adding an item is by specifying an `OfferListingId` because an `OfferListingId`, not an ASIN, guarantees that an item can be purchased.

Availability Parameter

The `Availability` parameter filters out of `ItemSearch` results those items that are unavailable. The availability of an item can change rapidly. There is typically a discrepancy between an item's availability as reported by `ItemSearch` and the item's true availability, as reported by Amazon's web site. For this reason, the availability of items reported by `ItemSearch` and by Amazon's web site will be slightly different. Items that are "available" are classified on Amazon's retail web site as:

- Currently for sale
- Pre-orders
- Special orders
- New releases
- E-mail me when items become available
- Items available for in-store pickup
- Items for sale by third parties

Parameter Restrictions for Availability

The following table describes the `ItemSearch` parameters that must be included to return available items (only).

ItemSearch Parameter	Description
Availability	Must be set to "Available". When the <code>Availability</code> parameter is not set, <code>ItemSearch</code> returns available and unavailable items. "Available" is the only valid value for <code>Availability</code> . Setting it to another value returns an error message. Parameter values are case sensitive. When the <code>Availability</code> parameter is set to "Available", the only optional constraint that can be present is <code>MerchantId</code> .

ItemSearch Parameter	Description
MerchantId	An optional parameter that can be used to filter search results and offer listings to only include items sold by Amazon. By default, the API will return items sold by various merchants including Amazon. The only valid value for MerchantId is "Amazon".

The following search indices do not work with the `Availability` parameter:

- Items available for in-store pickup
- Items for sale by third parties

In both cases, because Amazon does not warehouse the items for sale, Amazon cannot determine the availability of them.

Items Sold by Merchants

By far, most items on Amazon are sold by merchants. Amazon itself is one. For that reason, [ItemSearch \(p. 185\)](#) is one of the most commonly used operations. Because there are so many items for sale on Amazon, ItemSearch has many input parameters that help target a request. ItemSearch requests can also use many response groups, which can tailor the type of information returned with each item.

At first, the number of ItemSearch input parameters might seem overwhelming. This section explains the parameters most commonly used. By mastering these, you can create highly targeted requests. For a description of all of the ItemSearch parameters, see [ItemSearch \(p. 185\)](#).

Required ItemSearch Parameters

ItemSearch can potentially return any item sold by a merchant and most sellers. Because there are so many items, ItemSearch requires that you specify a search index and at least one additional parameter, as shown in the following example.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=ItemSearch&  
SearchIndex=Books&Keywords=Saving%20Miss%20Oliver's  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

In this request, ItemSearch looks for "Saving Miss Oliver's" in the titles and descriptions of all of the items in the Books search index. The response would include all matches.

The previous example used the `Keywords` parameter as the second parameter, which is one of the most popular parameters to use. The following list shows all of the ItemSearch parameters that you could have used instead.

- | | | |
|---|--|---|
| <ul style="list-style-type: none">• Actor• Artist• AudienceRating• Author• Brand• BrowseNode | <ul style="list-style-type: none">• Composer• Conductor• Director• Keywords• Manufacturer• MusicLabel | <ul style="list-style-type: none">• Orchestra• Power• Publisher• Title |
|---|--|---|

A cursory look at the list of parameters shows you that most are very specific, which makes it easy to determine whether or not they should be used in a request. The Publisher parameter, for example, is used only to specify the publisher of a book. Once you know the item you are looking for, choosing which parameter to use becomes easy.

Parameter Support by Search Index

Each search index supports only a subset of all `ItemSearch` parameters. For example, in the US locale, when you specify the Blended search index, you can only use one parameter: `Keywords`, in a request. Using any of the other parameters in that request would return an error.

All other search indices support multiple parameters. For example, when you specify the Beauty search index, in the US locale, you can include one or more of the following parameters in the request.

- | | | |
|--|---|---|
| <ul style="list-style-type: none">• Brand• BrowseNode• Condition | <ul style="list-style-type: none">• ItemPage• Keywords• Manufacturer• MaximumPrice | <ul style="list-style-type: none">• MinimumPrice• Sort• Title |
|--|---|---|

The search index—parameter combinations, for the most part, make sense. For example, if you use the Automotive search index, the `Author` parameter is invalid in the request. This restriction makes intuitive sense. So, in general, it is not the case that you need to memorize all of the parameters that can be used in each search index.

Search Indices, Parameters, and Locales

There is one further restriction on the use of parameters in an `ItemSearch` request. The search index values that can be used in an [ItemSearch \(p. 185\)](#) request vary by locale. For example, the Baby and Beauty search indices are available in the US but not in the UK locale. An error is returned if you use a search index value that is not supported in a locale. The limitations on the use of search indices by locale therefore also limits the use of `ItemSearch` parameters.

For a complete list of `ItemSearch` parameters supported in each search index and in each locale, see [Locale Reference \(p. 328\)](#).

Common ItemSearch Parameters

Topics

- [Return Only Available Items \(p. 77\)](#)
- [Inaccurate Availabilities \(p. 78\)](#)
- [Availability Parameter Restrictions and Constraints \(p. 79\)](#)

- [Search Across Indices \(p. 79\)](#)

After you decide on a search index value and check [Locale Reference \(p. 328\)](#) to determine what [ItemSearch \(p. 185\)](#) input parameters are valid for that search index, you should set values for as many valid parameters as you can to increase the accuracy of the search results.

The most commonly used input parameters are the ones that can be used with the most search indices, including:

Availability

Specifies that the item must be available for purchase. The only valid value for the parameter is "Available".

BrowseNode

Enables you to search a specified browse node for associated items

Condition

Enables you to specify the condition of an item. Valid values are "All", "New", "Used", "Collectible", and "Refurbished". The default is "New". Condition does not restrict the total number of items returned. It does, however, restrict the offers returned to those items that are in the specified condition.

Keywords

A word or phrase (words separated by percent-encoded spaces, %20) used as a search criteria. The titles and descriptions of items are searched for keywords.

MaximumPrice

The maximum price that an item can cost.

MinimumPrice

The minimum price that an item can cost.

Title

A word or phrase (words separated by percent-encoded spaces, %20) used as a search criteria. The titles of items are searched for these words.

The definitions for all `ItemSearch` parameters are in the API Reference. The following sections describe in greater detail some of the commonly-used parameters.

Return Only Available Items

`ItemSearch` returns available and unavailable items. Unavailable items are, for example, items that are temporarily out of stock. By setting the `Availability` parameter to "Available," `ItemSearch` can filter out most of the items that are unavailable. The following `ItemSearch` request returns shirts that are available.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=ItemSearch&  
Condition=All&  
Availability=Available&  
SearchIndex=Apparel&  
Keywords=Shirt  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

"Available" is the only valid value for the parameter. Setting it to any other value returns an error message. By default, the value is not set.

Items that are available are classified on Amazon's retail web site as:

- Currently for sale
- Pre-orders
- Special orders
- New releases
- E-mail me when items become available
- Items available for in-store pickup
- Items for sale by third parties

The availability of an item can change rapidly. There is typically a discrepancy between an item's availability as reported by `ItemSearch` and the item's true availability, as reported by Amazon's web site. For this reason, the availability of items reported by `ItemSearch` and by Amazon's web site can be slightly different. These differences are most evident for items, such as software, whose availability changes often.

Note

Do not confuse the `Availability` parameter with the return element, `Availability`. The latter is an element in a response that describes how soon an item can be shipped, such as, "Ships in 48 hours". While this is a helpful value to display, it does not guarantee that an item is truly available to be purchased. For that determination, test for an `OfferListingId`, which is provided by the `Offers`, `OfferFull`, and `Large` response groups.

Inaccurate Availabilities

The `Availability` parameter does a good job of filtering out of the response items that are unavailable. The parameter, however, is not foolproof. It is possible for an item to be unavailable for purchase even if:

- The item is returned in a response
- The response has been filtered using the `Availability` parameter
- The `Availability` element in the response says the item can be shipped in twenty-four hours

For example, some items, including [Variations \(p. 35\)](#) parents, are never available for purchase because they do not represent real items; they are abstractions. These parent items, however, are often returned as being available. The following request returns a Collection parent item even though the `Availability` parameter is used.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=ItemLookup&  
ItemId=B0006PLAOE&  
Availability=Available&  
ResponseGroup=ItemAttributes,Offers&  
Condition=All  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

A snippet of the response shows that the Collection parent item is returned, so you might assume that it is available. However, the item is not associated with any offers, so it cannot be purchased.

```
<Item>
  <ASIN>B0006PLAOE</ASIN>
  ...
  <Offers>
    <TotalOffers>0</TotalOffers>
```

The contradiction of an item being available but not something you can buy shows that the `Availability` parameter is helpful but not the final arbiter in determining what items a customer can actually purchase. Instead, as was mentioned at the beginning of this chapter, a better test for availability is whether or not an item has an `OfferListingId`. Also, in the preceding example, you can see that you can test the value of `TotalOffers` to determine if an item has an offer.

Process to Include Item's Availability

1	Include in an ItemSearch request <code>Availability = Available</code> .
2	Include in the request <code>ResponseGroup = Offers</code> .
3	Test to see that the value in the response for <code>TotalOffers</code> is greater than zero.

Even though the `Availability` parameter is not foolproof, it does have value. The advantage gained by using the parameter is that it prevents the return of items that certainly do not have an `OfferListingId`. This filtering reduces the amount of testing you need to do to make sure the returned items have an `OfferListingId`.

Availability Parameter Restrictions and Constraints

The `Availability` parameter carries with it a restriction and some constraints. The restriction is that the `Availability` parameter cannot be used with the Blended search index. When the `Availability` parameter is set to "Available", the only optional constraint that can be present is `MerchantId=Amazon`.

Parameter	Description
<code>MerchantId</code>	An optional parameter that can be used to filter search results and offer listings to only include items sold by Amazon. By default, the API will return items sold by various merchants including Amazon. The only valid optional value for <code>MerchantId</code> is "Amazon".

Search Across Indices

ItemSearch requests require that you specify a search index. This is because searching across the millions of products in Amazon databases would take too long. Product Advertising API does, however, enable you to search across multiple search indices using the All or Blended search indices.

All Search Index

You can use the All search index to do an ItemSearch search through all search indices. There are, however, a number of restrictions placed on this request: the only parameter that you can use in the request is `Keywords`, and you cannot, for example, sort results.

Blended Searches

ItemSearch searches through a specified search index, or SearchIndex can be set to "Blended". A blended search always searches through the following search indices (only).

The indices that are searched are specific to a particular marketplace. The following tables list the search indices for each marketplace.

Note

Blended searches are not supported in CN, IT, and ES.

Blended search indices in CA

<ul style="list-style-type: none">• Books• DVD• Electronics• ForeignBooks	<ul style="list-style-type: none">• Kitchen• Music• Software• SoftwareVideoGames	<ul style="list-style-type: none">• SportingGoods• Tools• VHS• VideoGames
--	---	--

Blended search indices in DE

<ul style="list-style-type: none">• Apparel• Automotive• Books• DVD• Electronics• ForeignBooks• Grocery	<ul style="list-style-type: none">• HealthPersonalCare• Kitchen• Music• PCHardware• Shoes• Software	<ul style="list-style-type: none">• SoftwareVideoGames• SportingGoods• Tools• Toys• VHS• VideoGames
---	--	--

Blended search indices in FR

<ul style="list-style-type: none">• Apparel• Books• DVD• Electronics• ForeignBooks	<ul style="list-style-type: none">• HealthPersonalCare• Kitchen• Music• PCHardware• Shoes	<ul style="list-style-type: none">• Software• SoftwareVideoGames• SportingGoods• Toys• VHS• VideoGames
--	---	---

Blended search indices in JP

<ul style="list-style-type: none">• Apparel• Automotive• Books• DVD• Electronics• ForeignBooks• Grocery	<ul style="list-style-type: none">• HealthPersonalCare• Kitchen• Music• PCHardware• Shoes	<ul style="list-style-type: none">• Software• SportingGoods• Toys• VHS• VideoGames
---	---	--

Blended search indices in UK

<ul style="list-style-type: none">• Apparel• Automotive• Books• DVD• Electronics• Grocery	<ul style="list-style-type: none">• HealthPersonalCare• Kitchen• Music• PCHardware• Shoes• Software	<ul style="list-style-type: none">• SoftwareVideoGames• SportingGoods• Tools• Toys• VHS• VideoGames
--	--	--

Blended search indices in US

<ul style="list-style-type: none">• Apparel• Automotive• Books• DVD• Electronics• GourmetFood• Grocery	<ul style="list-style-type: none">• HealthPersonalCare• HomeGarden• Kitchen• Music• PCHardware• Shoes	<ul style="list-style-type: none">• Software• SportingGoods• Tools• Toys• VHS• VideoGames
--	--	--

Power Searches

Topics

- [Power Search Syntax \(p. 82\)](#)

ItemSearch's Power parameter, which can only be used when the search index equals Books, enables you to use simple or complex queries to perform book searches. A simple power search would be to return all books written by a specified author. While this query is valuable, it does not offer anything more than a similar search using the Author parameter in an ItemSearch request.

The true value of power searches comes in the ability to specify multiple criteria to search for books. For example, you could create a query that returned all books by a specific author, published during a specified year, in a specified language, and printed in a specified format.

Power Search Syntax

The individual parts of a power search query are of the following form

```
key:value
```

For example, the following query string searches for books authored by Davenport.

```
author:Davenport
```

Values are not case sensitive. So, "Davenport" returns the same books as "davenport".

The individual parts of a power search query are linked together using a variety of operators. The following example searches for all books authored by Davenport that were published during 2006.

```
author:Davenport and pubdate:during 2006
```

Other operators, such as parentheses, can be used to create additional specificity. For example, a query of subject:history and (spain or mexico) and not military and language:spanish would return a list of books in the **Spanish** language on the **subject** of either **Spanish** or **Mexican history**, excluding all items with **military** in their description.

These parameter values would be part of an `ItemSearch` request that used the `Power` parameter, as shown in the following example.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=ItemSearch&  
SearchIndex=Books&Power=subject:history%20and%20(spain%20or%20mex  
ico)%20and%20not%20military%20and%20language:spanish  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Note that spaces must be URL-encoded (%20) as shown.

Power Search Keywords

Power search queries are created using one or more keyword:value pairs. The keywords you can use in a power search are described in the following table.

Keyword	Description
after	Restricts the results to books published after the specified date. This keyword is used with pubdate, for example, pubdate:after 2006.
ASIN	An alphanumeric token that uniquely identifies a book.
author	Book's author.
author-begins	Author's name begins with this value.
author-exact	Authors name is exactly this value.

Keyword	Description
binding	Searches for books with the specified binding. Values for binding include abridged and large print.
during	Searches for books that were published during the time period specified. This keyword is used with pubdate, for example, pubdate:during 2006.
EISBN	Electronic International Standard Book Number of the digital book.
ISBN	International Standard Book Number of the book.
keywords	Words that can be found in the title or description of a book.
keywords-begin	Search for all words that begin with this value.
language	Language, such as Spanish, that the book is written in.
pubdate	Book's publication date.
publisher	Name of the book's publisher.
subject	Find books with the specified word in their subject description.
subject-begins	Find books whose subject description begins with this word or character.
subject-words-begin	Find books whose subject description begins with these words
title	Title of the book.
title-begins	Title of the book begins with this value exactly. This value does not mean "begins with", it means "begins". For the example, if the value is "I", the book, "I Lied", would be returned; the book, "Isabel", would not be returned.
title-words-begin	Title of the book begins with this word.

Power Search Operators

The following table describes the Power search queries operators that you use to build queries.

Header 1	Header 2
not	Excludes the following parameter from the results, for example, subject:history and not military, excludes military history in the results.
and	Specifies that both values must be true to be selected. For example, subject:history and (Spanish and Mexican), requires that the books selected contain both Spanish and Mexican history.
or	Exclusive or which means one of either item but not both. For example subject:history and (Spanish or Mexican), means the subject matter can be about the history of Spain or Mexico, but not both.
colon (:)	Used as an equals sign, for example, subject:history, searches for books whose subject matter is history.

Header 1	Header 2
parenthesis	Groups terms to clarify operations, for example, subject: history and (spain or mexico) Without the parentheses, you would search for books about Spanish history or Mexico. With parentheses, you search for books about Spanish history or Mexican history.
asterisk (*)	Stands for zero or more alphanumeric characters, for example, keywords:high%20tech* Some of the keywords used in a search would be "high tech", "high technology", and "high technologies". Use the asterisk to generalize your search.
Quotation marks (" ")	Specifies an exact match of the word(s) within the quotes, for example, keywords: "Saving Miss Oliver's"

Example Power Searches

The following five Power Search examples demonstrate many of the key words and operators.

```
author: ambrose and binding: (abridged or large print) and pubdate: after 11-1996
subject: history and (Spain or Mexico) and not military and language: Spanish
(subject: marketing and author: kotler) or (publisher: harper and subject: "high technology")
keywords: "high tech*" and not fiction and pubdate: during 1999
isbn: 0446394319 or 0306806819 or 1567993850
```

Power Search Tips

Topics

- [Exact Matches \(p. 84\)](#)
- [Discarded Search Words \(p. 85\)](#)

The following tips help you avoid common problems when using power search keywords.

Exact Matches

title-begins require exact matches. So, the following examples return completely different results.

```
title-begins:I
title-begins:I*
```

The first example would return the book, "I Lied", but it would not return the book, "Isabel". The second example would return both books. This behavior also applies to:

- author
- author-begins
- author-exact
- title

- title-words-begin

Discarded Search Words

The search engine automatically discards small words, such as "A" and "The". So, for example, if you created queries using either of the following key:value pairs. `title-begins:A` `title-begins:The` you would get zero results because the search engine discards those words when searching through titles. This behavior is also true for:

- author
- author-begins
- author-exact
- title
- title-words-begin

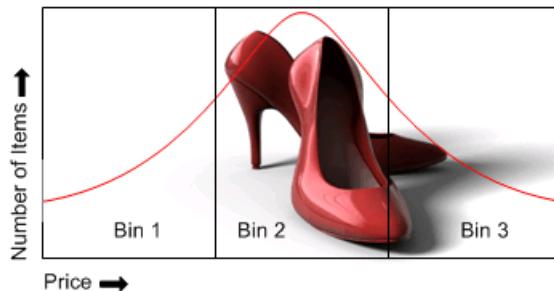
Use Search Bins to Find Items

Topics

- [NarrowBy Attribute \(p. 86\)](#)
- [Search Bin Example \(p. 86\)](#)
- [NarrowBy Values \(p. 87\)](#)
- [Results Refined by Repeating ItemSearch Requests \(p. 89\)](#)
- [ItemSearch Request Tips \(p. 91\)](#)

Up to this point, you have seen how you can use operation input parameters and response groups to filter out unwanted responses. The `SearchBins` response group provides a different means of refining results. It enables you to filter results based on values returned in a response.

The `SearchBins` response group categorizes the items returned by `ItemSearch` into groups, called bins. The grouping is based on some criteria, depending on the search index. For example, a set of bins can be based on a set of price ranges for an item. In the case of women's shoes, for example, `SearchBins` might return a bin that contains ASINs for shoes that cost between \$0 and \$50, a second bin for shoes that cost \$50 to \$100, and a third bin for shoes that cost more than \$100.



The advantage of using search bins is that the response group divides the items into bins without you having to return or parse item attributes. You can then submit a second `ItemSearch` request and return only the items in one bin.

You cannot create bins nor can you specify the criteria used to divide the items into groups. The `SearchBins` response group does that automatically.

Some search indices support more than one kind of bin. For example, apparel items can be divided in to bins according to price range and brand. In this case, the response would return multiple sets of bins, called SearchBinSets, in which the items would be divided according to different criteria.

The criteria used to divide the returned items into bins is called the NarrowBy value.

Note

Search bins are only available in the US locale.

Related Topics

- [SearchBins Response Group \(p. 286\)](#)

NarrowBy Attribute

The basis on which returned items are split into bins is specified by the `NarrowBy` attribute in the `SearchBinSet` element. In the following example, the `NarrowBy` attribute shows that the bins are based on price range.

```
<SearchBinSet NarrowBy="PriceRange">
```

For another product category, the `NarrowBy` attribute might be different, for example:

```
<SearchBinSet NarrowBy="BrandName">
```

You cannot specify `NarrowBy` values nor can you specify the values they encompass. When the `SearchBins` response group is included in a request, `ItemSearch` automatically divides the `ItemSearch` results into bins.

Search Bin Example

The following request uses the `SearchBins` response group to return search bins.

```
Service=AWSECommerceService&
AWSAccessKeyId=[Access Key ID]&
AssociateTag=[Associate Tag]&
Operation=ItemSearch&
SearchIndex=Baby&
Keywords=pants&
Availability=Available&
Condition=All&
ResponseGroup=SearchBins
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The following XML is a snippet from the response.

```
<SearchBinSets>
  <SearchBinSet NarrowBy="PriceRange">
    <Bin>
      <BinName>$0-$24</BinName>
      <BinItemCount>1645</BinItemCount>
      <BinParameter>
        <Name>MinimumPrice</Name>
```

```
<Value>0</Value>
</BinParameter>
<BinParameter>
<Name>MaximumPrice</Name>
<Value>2499</Value>
</BinParameter>
</Bin>
<Bin>
<BinName>$25$49</BinName>
<BinItemCount>647</BinItemCount>
<BinParameter>
<Name>MinimumPrice</Name>
<Value>2500</Value>
</BinParameter>
<BinParameter>
<Name>MaximumPrice</Name>
<Value>4999</Value>
</BinParameter>
</Bin>
<Bin>
<BinName>$50$99</BinName>
<BinItemCount>173</BinItemCount>
<BinParameter>
<Name>MinimumPrice</Name>
<Value>5000</Value>
</BinParameter>
<BinParameter>
<Name>MaximumPrice</Name>
<Value>9999</Value>
</BinParameter>
</Bin>
```

This response snippet shows the first three bins in the response. The NarrowBy value shows that the items were divided up based on price range. The BinName element names the bin. The names are descriptive of the price ranges that each bin represents. For example, the BinName, \$50\$99, contains items that cost between \$50 and \$99.99, which you can see by the values returned for MinimumPrice and MaximumPrice in that bin. The BinItemCount element shows how many items are in each bin, for example, there are 173 items in the last bin.

The BinParameter/Value elements show the values used to create the bins. In this example, the parameters are the minimum and maximum prices of the items in that bin. For example, in the last bin, the minimum price of an item in that bin is \$50.00 and the maximum value is \$99.99.

The BinParameter/Name value, such as MaximumPrice, is an ItemSearch parameter name. This means that you can use the <Value> as the value for the parameter named by <Name> in a subsequent ItemSearch request. In this example, MinimumPrice is the ItemSearch parameter and, in the last bin, the value is 5000. By submitting a second request using ItemSearch's parameters, MinimumPrice and MaximumPrice , you could return the item attributes for only the items in that bin.

As you can see from this example, the SearchBins response group enables you to narrow your search without you having to parse through item attributes.

NarrowBy Values

The following table describes the NarrowBy values.

NarrowBy Values	Description
BrandName	Brands, such as Levi's, Reebok, and Nike, that create the item. Use the name of a brand to filter out similar items made by other companies.
PriceRange	Minimum and maximum prices for a bin of items. Use the minimum and maximum price values in each bin to filter out items outside of the price range you want.
SpecialSize	Uncommon sizes an item comes in. Examples are "Plus Size & Tall", "Misses", "Maternity", "Husky", "Petites", and "Big & Tall".
Subject	BrowseNode IDs of all topics related to items returned by ItemSearch. For example, searching for books about dogs returns, in the Subject bins, BrowseNodes for "Home & Garden", "Animal Care & Pets", "Dogs", and "Educational".

All search indices return the Subject bin. Most return all of the bins.

NarrowBy Values Per Search Index

The following table shows which NarrowBy values are returned by each search index:

Search Index	NarrowBy
Apparel	Subject, BrandName, PriceRange, SpecialSize
Baby	Subject, BrandName, PriceRange, SpecialSize
Beauty	Subject, BrandName, PriceRange, SpecialSize
Blended	Not supported
Books	Subject
Classical	Subject
DVD	Subject
Electronics	Subject, BrandName, PriceRange, SpecialSize
Garden	Subject, BrandName, PriceRange, SpecialSize
GourmetFood	Subject, BrandName, PriceRange, SpecialSize
HealthPersonalCare	Subject, BrandName, PriceRange, SpecialSize
Jewelry	Subject, BrandName, PriceRange, SpecialSize
Kitchen	Subject, BrandName, PriceRange, SpecialSize
Magazines	Subject
Marketplace	Subject
Miscellaneous	Subject, BrandName, PriceRange, SpecialSize
Music	Subject
MusicalInstruments	Subject, BrandName, PriceRange, SpecialSize

Search Index	NarrowBy
OfficeProducts	Subject, BrandName, PriceRange, SpecialSize
PCHardware	Subject, BrandName, PriceRange, SpecialSize
Photo	Subject, BrandName, PriceRange, SpecialSize
Software	Subject, BrandName, PriceRange, SpecialSize
SportingGoods	Subject, BrandName, PriceRange, SpecialSize
Tools	Subject, BrandName, PriceRange, SpecialSize
Toys	Subject, BrandName, PriceRange, SpecialSize
VHS	Subject
Video	Subject
VideoGames	Subject, BrandName, PriceRange, SpecialSize
Wireless	Subject
WirelessAccessories	Subject

Results Refined by Repeating ItemSearch Requests

One value of using search bins is that you can divide items into groups according to criteria without having to parse item attributes. Based on the search bins returned, you can then submit a second request using the `ItemSearch` parameter value that helps target your results, but how?

The names of bins and the parameters that describe the bins vary according to the bin. The following sample shows a bin based on price.

```
<SearchBinSetNarrowBy="PriceRange">
<Bin>
  <BinName>$0$24</BinName>
  <BinItemCount>1645</BinItemCount>
  <BinParameter>
    <Name>MinimumPrice</Name>
    <Value>9</Value>
  </BinParameter>
  <BinParameter>
    <Name>MaximumPrice</Name>
    <Value>2499</Value>
  </BinParameter>
</Bin>
```

The response shows the minimum and maximum price for items in the bin, \$0\$24, and the number of items in it, 1645.

Other NarrowBy values name bins differently.

As you can see from these examples, BinParameter names are the same as `ItemSearch` input parameter names. This correspondence means that you can create a second `ItemSearch` request using the search bin results as values for `ItemSearch` parameter values. For example, `MinimumPrice` and `MaximumPrice` are returned in search bins based on `PriceRange`. You could take the values of the search bin and put

them directly into `ItemSearch` parameters. Using the preceding `PriceRange` example, you could write the following `ItemSearch` request to retrieve items only in the first search bin.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=ItemSearch&  
SearchIndex=Baby&  
Keywords=pants&  
Availability=Available&  
Condition=All&  
MinimumPrice=0& MaximumPrice=2499&  
ResponseGroup=SearchBins  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

`ItemSearch` divides the results of this request into another set of search bins because the `SearchBins` response group was used again. This means that the price range of the first search bin in the first response is split into multiple search bins in the response to the second request. The second response enables you to present more granularity in price ranges. For example, from the first response, you could return all items that cost between \$0 and \$24.99. In the second response, you are able to provide a much smaller price interval, for example, \$10 to \$14.99.

To refine search results using `SearchBin`

1. Include the `SearchBins` response group in an `ItemSearch` request.

The response includes the `NarrowBy` value and results grouped by bins.

2. Use related `ItemSearch` parameters to send a second request to return results related to only those items within the specified bin.

For example, if the `NarrowBy` value is `PriceRange`, the related `ItemSearch` parameters are `MinimumPrice` and `MaximumPrice`.

3. If you want to refine the results further, use the `SearchBins` response group in a second request and repeat the procedure.

The process of using search bin results for `ItemSearch` parameter values can be iterative. You can, for example, submit a third request using the `SearchBins` response group to divide one search bin into more search bins. This process can be repeated until the level of granularity you desire is reached. At that point, you can send a last request using other response groups of your choosing.

Alternatively, you could refine the search results in a different way. Some search indices return more than one set of search bins. In those cases, you can use the values from more than one set of search bins in an `ItemSearch` request. Using the preceding example, if the response also included a search bin based on `BrandName`, which is the `NarrowBy` value, you could use brand and price range values in an `ItemSearch` request.

```
Brand=Levi's&  
MinimumPrice=0&  
MaximumPrice=2499&
```

The response would then only include shirts by Levi's that cost under \$25. You could continue to drill down by adding additional parameters to the request.

ItemSearch Request Tips

Here are some tips to help you create accurate [ItemSearch \(p. 185\)](#) requests.

- If you want to find only items sold by Amazon, set the MerchantId parameter to "Amazon". (Note: Amazon sells only new items).
- The Keywords parameter searches for word matches in an item's title and description. If you know a word is part of the title of an item, use the Title parameter because, in this case, it often returns fewer but more accurate results than the Keywords parameter.
- To use Boolean values, such as AND, NOT, or OR, in an ItemSearch request, use the Power parameter. You can create relatively sophisticated search criteria using this parameter. For more information, see [ItemSearch \(p. 185\)](#).

Find Items with Browse Nodes

Topics

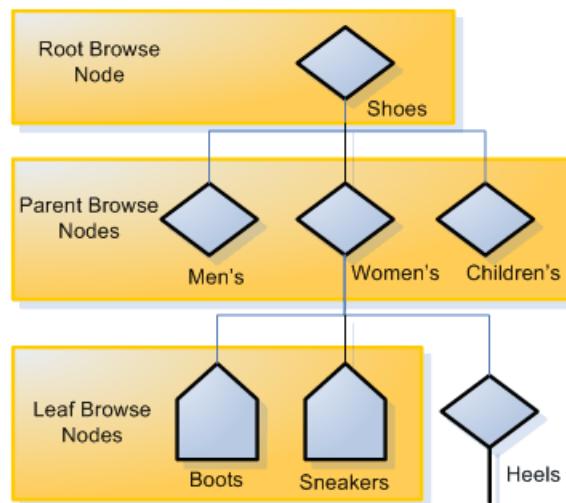
- [Traverse the Browse Node Hierarchy \(p. 92\)](#)
- [Find a Node to Start Your Search \(p. 93\)](#)
- [Understand BrowseNode Results When Drilling Down \(p. 94\)](#)

Browse nodes form an organizational hierarchy of items for sale. The hierarchy of nodes is designed to make it easy and fast to browse for items.

Each browse node has an ID (a positive integer) and a name. All items associated with a node are related to the name of the node. For example, a node name "Books about Ancient Greece" will include books about ancient Greece.

This hierarchy of nodes is dynamic, as are the items associated with each node; items can be added and removed from browse nodes at any time. Nodes can be added or removed in the node hierarchy.

The value of the hierarchy is that it groups similar items and relates items in an intuitive way. As you move down the hierarchy, you move from parent to child nodes, where child nodes are subsets of the parent node's product category, as shown in the following figure.



As you move down the hierarchy, you refine the number of items that can be returned. In this example, the Shoes category has as a subset, "Women's Shoes". It has two child nodes, "Boots" and "Sneakers".

An item can be associated with more than one browse node, and that a browse node can be associated with more than one browse node.

[ItemSearch \(p. 185\)](#) returns all of the items associated with a node and all of the nodes below it. You can see that a search at the level of "Women's Shoes" will return all items related to "Women's Boots" and "Women's Sneakers". If the customer is only interested in women's boots, a more targeted search will specify the "Boots" browse node using ItemSearch's BrowseNode parameter. For example, if the browse node ID of "Women's Boots" is "123456", a targeted request will look like the following:

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=ItemSearch&  
Keywords=tall&  
SearchIndex=Apparel&  
BrowseNode=123456  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Traverse the Browse Node Hierarchy

You move down the hierarchy to refine your search. You might also move up the hierarchy to find a parent browse node or even the root category of a product.

For example, if you have an item like a carving knife, you might find its node and go up the hierarchy to find the root product category of knives. In another request, use the TopSellers response group to return the top sellers in the product category.

[BrowseNodeLookup \(p. 194\)](#) operation is the most direct way of traversing the browse node hierarchy. When you supply it a browse node ID, it returns the name of the browse node as well as its direct descendants and a lineage of ancestors. Look at the response to the following request.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=BrowseNodeLookup&  
BrowseNodeId=11232&  
ResponseGroup=BrowseNodeInfo  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

A snippet of its response follows.

```
<Item>  
<ASIN>0131856340</ASIN>  
<BrowseNodes>  
  <BrowseNode>  
    <BrowseNodeId>11232</BrowseNodeId>  
    <Name> Social Sciences</Name>  
    <Ancestors>  
      <BrowseNode>  
        <BrowseNodeId>53</BrowseNodeId>  
        <Name>Nonfiction</Name>  
      <Ancestors>
```

```
<BrowseNode>
  <BrowseNodeId>1000</BrowseNodeId>
  <Name>Subjects</Name>
  <Ancestors>
    <BrowseNode>
      <BrowseNodeId>283155</BrowseNodeId>
      <Name>Books</Name>
    </BrowseNode>
  </Ancestors>
  <BrowseNode>
    <Ancestors>
      <BrowseNode>
        <BrowseNodeId>11233</BrowseNodeId>
        <Name>Anthropology</Name>
      </BrowseNode>
      <BrowseNode>
        <BrowseNodeId>11242</BrowseNodeId>
        <Name>Archaeology</Name>
      </BrowseNode>
      <BrowseNode>
        <BrowseNodeId>3048861</BrowseNodeId>
        <Name>Children's Studies</Name>
      </BrowseNode>
    </Ancestors>
  <Children>
    <BrowseNode>
      <BrowseNodeId>11233</BrowseNodeId>
      <Name>Anthropology</Name>
    </BrowseNode>
    <BrowseNode>
      <BrowseNodeId>11242</BrowseNodeId>
      <Name>Archaeology</Name>
    </BrowseNode>
    <BrowseNode>
      <BrowseNodeId>3048861</BrowseNodeId>
      <Name>Children's Studies</Name>
    </BrowseNode>
  </Children>
</BrowseNodes>
```

To move down the hierarchy, choose the browse node ID that is relevant and repeat the `BrowseNodeLookup` operation until you find to the appropriate browse node.

Moving up the hierarchy is similar. Although the [BrowseNodeLookup \(p. 194\)](#) operation, by default, returns the complete ancestral lineage of the subject browse node, there is a limit. The request only returns one parent browse node for each node. If a node has more than one parent, the response will only show one of the parent nodes. The parent browse node returned is arbitrary. For that reason, you might want to move up the hierarchy one node level at a time.

Related Topics

- [BrowseNodes \(p. 236\)](#)

Find a Node to Start Your Search

In the US locale, there are over 120,000 nodes and they are constantly changing. Fortunately, there are multiple ways to find the browse node where you can start your search:

- Many high level browse nodes are listed in [Locale Reference \(p. 328\)](#) for each locale.
- The `BrowseNodes` response group returns browse nodes. You can use this response group with the following operations: [ItemSearch \(p. 185\)](#), and [SimilarityLookup \(p. 203\)](#).
- On Amazon.com, search for an item that is similar to the one you want and then copy the browse node from the URL.

For example, this request searches for item listings for a horse bridle. None of the top product categories (search indices) relate directly to horses. So, use `ItemSearch` and the `BrowseNodes` response groups to find a browse node that is associated with horse bridles.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=ItemSearch&  
Keywords=horse,bridle&  
SearchIndex=PetSupplies,SportingGoods  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

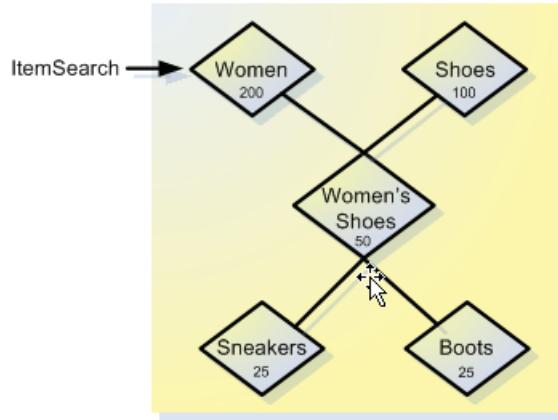
This request looks in the PetSupplies and SportingGoods search indices for anything related to horses or bridles.

Related Topics

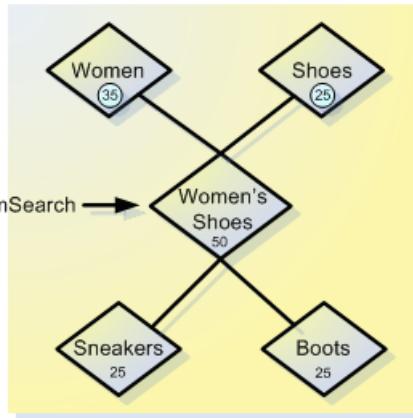
- [BrowseNodes \(p. 236\)](#)

Understand BrowseNode Results When Drilling Down

`BrowseNodes` are related in a hierarchy where one `BrowseNode` can have zero or more ancestor and child `BrowseNodes`, as shown in the following figure.



- This diagram shows five `BrowseNodes` and their hierarchy. Node A might be, for example, "Women". Node B might be "Shoes". Node C, a child of `BrowseNodes` A and B, might be "Women's Shoes".
- The numbers represent the number of items in each `BrowseNode`.
- "Women's Shoes" is a subset of the two browse nodes, "Women" and "Shoes". It's possible that all 50 items in "Women's Shoes" are also in "Women".
- When `ItemSearch` searches "Women", the operation returns that the node has 200 total items.
- When `ItemSearch` is repeated with a `BinParameter Name` value that was returned in the first `ItemSearch` request, it is the same as running `ItemSearch` on a child node of "Women", which in this case is "Women's Shoes".
- By narrowing the search to only the values found in "Women's Shoes", `ItemSearch` returns only items in "Women" and "Women's Shoes". The number of common items is 35.



As a result, `ItemSearch` returns the items from "Women", which has 35 items. The change in the item count in "Women" might be confusing, but understand that `ItemSearch` returns only the intersection of "Women" and "Women's Shoes", which explains why the number of items in "Women" changes. Note that the item counts in other `BrowseNodes` can also change.

Because the number of items in `BrowseNodes` can change with each `ItemSearch` operation, the `BrowseNodes` with the greatest number of items can change dramatically. In the previous example, the number of items in "Women" changed from 200 to 35 after successive `ItemSearch` requests. Because `ItemSearch` returns only the top ten `BrowseNodes` that have the most items, the identity of the top ten `BrowseNodes` can change with each `ItemSearch`.

Request and Response Examples

The following examples show how to search for items with the Product Advertising API. By default, the Product Advertising API returns up to 10 items per page.

Topics

- [Discover Accessories \(p. 96\)](#)
- [Discover Available Items Only \(p. 97\)](#)
- [Discover Newer Versions \(p. 98\)](#)
- [Discover Similar Items \(p. 101\)](#)
- [Discover Top Lists \(p. 102\)](#)
- [Discover Variations \(p. 103\)](#)
- [Find Related Items \(p. 105\)](#)
- [Find Similar Items \(p. 107\)](#)
- [Lookup by ISBN \(p. 111\)](#)
- [Lookup by UPC \(p. 112\)](#)
- [Retrieve Customer Reviews \(p. 113\)](#)
- [Retrieve Images \(p. 114\)](#)
- [Retrieve Price \(p. 116\)](#)
- [Search by Actor \(p. 118\)](#)
- [Search by Author \(p. 119\)](#)
- [Search by Keyword \(p. 120\)](#)
- [Search by Manufacturer \(p. 122\)](#)
- [Search by Theme \(p. 123\)](#)
- [Search by Theme in a Category \(p. 124\)](#)
- [Search by Title \(p. 125\)](#)

Discover Accessories

The following request uses the `ItemLookup` operation with the `Accessories` response group. The request finds accessories for ASIN `B0036WT3RA` (Samsung TV).

Example Request

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService  
&Operation=ItemLookup  
&ResponseGroup=Accessories  
&IdType=ASIN  
&ItemId=B0036WT3RA  
&AWSAccessKeyId=[Your_AWSAccessKeyID]  
&AssociateTag=[Your_AssociateTag]  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request_Signature]
```

Example Response

The following response returns five accessories.

```
<Item>  
  <ASIN>B0036WT3RA</ASIN>  
  <Accessories>  
    <Accessory>  
      <ASIN>B002TLTE6Y</ASIN>  
      <Title>AmazonBasics Universal Tilt TV Wall Mount for 34- to 65-Inch Displays [Amazon Frustration-Free Packaging]</Title>  
    </Accessory>  
    <Accessory>  
      <ASIN>B002TLTE7I</ASIN>  
      <Title>AmazonBasics Universal Fixed TV Wall Mount for 26- to 50-inch Displays [Amazon Frustration-Free Packaging]</Title>  
    </Accessory>  
    <Accessory>  
      <ASIN>B003ES5ZRS</ASIN>  
      <Title>AmazonBasics High Speed Mini-HDMI to HDMI Cable with Ethernet (9.8 Feet / 3 Meters)</Title>  
    </Accessory>  
    <Accessory>  
      <ASIN>B003ES5ZVO</ASIN>  
      <Title>AmazonBasics High Speed HDMI Cable with Ethernet - Braided (9.8 Feet / 3.0 Meters)</Title>  
    </Accessory>  
    <Accessory>  
      <ASIN>B003ES5ZUU</ASIN>  
      <Title>AmazonBasics High Speed HDMI Cable with Ethernet - Braided (6.5 Feet / 2 Meters)</Title>  
    </Accessory>  
  </Accessories>  
</Item>
```

Discover Available Items Only

The following request uses the `ItemSearch` operation to find engagement ring sets in the Jewelry search index. To return available items only, set the `Availability` parameter to `Available`.

Example Request

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService  
&Operation=ItemSearch  
&ResponseGroup=Small  
&Keywords=Engagement%20ring%20set  
&Availability=Available  
&SearchIndex=Jewelry  
&AssociateTag=[YourAssociateTag]  
&AWSAccessKeyId=[YourAWSAccessKeyId]  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[RequestSignature]
```

Example Response

The following response returns two items.

```
<MoreSearchResultsUrl>
    http://www.amazon.com/gp/redirect.html?AWSAccessKeyId=[Your_AWSAccessKey
ID]&ie=UTF8&location=http%3A%2F%2Fwww.amazon.com%2Fgp%2Fsearch%2F%3Fkeywords%3Den
gagement%2520ring%2520sets%26ie%3DUTF8%26url%3Dsearch-alias%253Djew
elry&tag=[Your_AssociateTag]&creative=386001&camp=2025&linkCode=xm2
</MoreSearchResultsUrl>
<Item>
    <ASIN>B000MFGN6I</ASIN>
    <DetailPageURL>
        http://www.amazon.com/Sterling-Silver-Zirconia-Earrings-Diamet
er/dp/B000MFGN6I%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_Associ
ateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS
IN%3DB000MFGN6I
    </DetailPageURL>
    <ItemAttributes>
        <Manufacturer>Amazon.com Collection</Manufacturer>
        <ProductGroup>Jewelry</ProductGroup>
        <Title>Sterling Silver Cubic Zirconia Hoop Earrings (0.6" Diameter)</Title>
    </ItemAttributes>
</Item>
<Item>
    <ASIN>B000SMNL2E</ASIN>
    <DetailPageURL>
        http://www.amazon.com/Sterling-Silver-Marcasite-Colored-
Pendant/dp/B000SMNL2E%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_As
sociateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS
IN%3DB000SMNL2E
    </DetailPageURL>
    <ItemAttributes>
        <Manufacturer>Amazon.com Collection</Manufacturer>
        <ProductGroup>Jewelry</ProductGroup>
        <Title>Sterling Silver Marcasite and Garnet Colored Glass Heart Pendant,
18"</Title>
    </ItemAttributes>
</Item>
```

Discover Newer Versions

The following request uses the `ItemLookup` operation with the `RelationshipType` parameter. Use the `NewerVersion` value to find the latest version of the item in the request. The requested item is ASIN `B00HQDBLDO` (Nikon Coolpix L330 Digital Camera).

Note

The `NewerVersion` value is supported in all locales, except Brazil (BR) and Mexico (MX).

Example Request

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService
&Operation=ItemLookup
&ItemId=B00HQDBLDO
&IdType=ASIN
&ResponseGroup=Images,ItemAttributes,Offers,RelatedItems
&RelationshipType=NewerVersion
&AssociateTag=YourAssociateTag
&SubscriptionId=YourAWSAccessKeyId
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=RequestSignature
```

The following response returns the latest version, which is ASIN B00THKEKEQ (Nikon Coolpix L340 Digital Camera).

Example Response

```
<RelatedItems>
  <Relationship>Children</Relationship>
  <RelationshipType>NewerVersion</RelationshipType>
  <RelatedItemCount>1</RelatedItemCount>
  <RelatedItemPageCount>1</RelatedItemPageCount>
  <RelatedItemPage>1</RelatedItemPage>
  <RelatedItem>
    <Item>
      <ASIN>B00THKEKEQ</ASIN>
      <ItemAttributes>
        <Binding>Camera</Binding>
        <Brand>DavisMAX</Brand>
        <CatalogNumberList>
          <CatalogNumberListElement>##-NIKON--VNA780E1</CatalogNumberListElement>

        </CatalogNumberList>
        <Color>Black</Color>
        <EAN>0018208943104</EAN>
        <EANList>
          <EANListElement>0018208943104</EANListElement>
        </EANList>
        <Feature>28x optical zoom, 56x Dynamic Fine Zoom</Feature>
        <Feature>20.2-MP CCD sensor for bright, sharp photos and HD
videos</Feature>
        <Feature>Larger ergonomic design for easier shooting</Feature>
        <Feature>HD 720p videos with sound bring your memories to life</Fea
ture>
        <IsEligibleForTradeIn>1</IsEligibleForTradeIn>
        <ItemDimensions>
          <Height Units = "hundredths-inches">300</Height>
          <Length Units = "hundredths-inches">437</Length>
          <Weight Units = "hundredths-pounds">95</Weight>
          <Width Units = "hundredths-inches">328</Width>
        </ItemDimensions>
        <Label>DavisMax</Label>
        <Manufacturer>DavisMax</Manufacturer>
        <Model>18208943104</Model>
        <MPN>18208943104</MPN>
        <PackageDimensions>
          <Height Units = "hundredths-inches">450</Height>
          <Length Units = "hundredths-inches">610</Length>
          <Weight Units = "hundredths-pounds">165</Weight>
          <Width Units = "hundredths-inches">540</Width>
        </PackageDimensions>
        <PackageQuantity>1</PackageQuantity>
        <PartNumber>18208943104</PartNumber>
        <ProductGroup>Photography</ProductGroup>
        <ProductTypeName>CAMERA_DIGITAL</ProductTypeName>
        <Publisher>DavisMax</Publisher>
        <Studio>DavisMax</Studio>
        <Title>Nikon COOLPIX L340 Digital Camera (Black) [Import Model]</Title>
      <TradeInValue>
        <Amount>6633</Amount>
        <CurrencyCode>USD</CurrencyCode>
        <FormattedPrice>$66.33</FormattedPrice>
      </TradeInValue>
    </Item>
  </RelatedItem>
</RelatedItems>
```

```
</TradeInValue>
<UPC>018208943104</UPC>
<UPCList>
  <UPCLlistElement>018208943104</UPCLlistElement>
</UPCList>
</ItemAttributes>
</Item>
</RelatedItem>
</RelatedItems>
```

Discover Similar Items

The following request uses the `ItemLookup` operation with the `Similarities` response group. The request finds items similar to UPC 047875840041 (Call of Duty: Black Ops video game).

Example Request

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService
&Operation=ItemLookup
&ResponseGroup=Similarities
&SearchIndex=All
&IdType=UPC
&ItemId=047875840041
&AWSAccessKeyId=[Your_AWSAccessKeyId]
&AssociateTag=[Your_AssociateTag]
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request_Signature]
```

Example Response

The following response returns five similar items.

```
<Item>
  <ASIN>B003JVCA9Q</ASIN>
  <SimilarProducts>
    <SimilarProduct>
      <ASIN>B0015AARJI</ASIN>
      <Title>PlayStation 3 Dualshock 3 Wireless Controller</Title>
    </SimilarProduct>
    <SimilarProduct>
      <ASIN>B002I0K6DG</ASIN>
      <Title>Killzone 3</Title>
    </SimilarProduct>
    <SimilarProduct>
      <ASIN>B000ZK9QCS</ASIN>
      <Title>God of War III</Title>
    </SimilarProduct>
    <SimilarProduct>
      <ASIN>B003L8DXOI</ASIN>
      <Title>Assassin's Creed: Brotherhood</Title>
    </SimilarProduct>
    <SimilarProduct>
      <ASIN>B000HKP88C</ASIN>
      <Title>Grand Theft Auto IV</Title>
    </SimilarProduct>
  </SimilarProducts>
</Item>
```

Discover Top Lists

The following request uses the `BrowseNodeLookup` operation with the `NewReleases` and `TopSellers` response groups. The request finds newly released items and best selling items in the browse node 1063498 (kitchen).

Example Request

```
http://webservices.amazon.com/onca/xml?
  Service=AWSECommerceService
  &Operation=BrowseNodeLookup
  &ResponseGroup=NewReleases,TopSellers
  &BrowseNodeId=1063498
  &AWSAccessKeyId=[Your_AWSAccessKeyId]
  &AssociateTag=[Your_AssociateTag]
  &Timestamp=[YYYY-MM-DDThh:mm:ssZ]
  &Signature=[Request_Signature]
```

Example Response

The following response returns four items.

```
<BrowseNode>
  <BrowseNodeId>1063498</BrowseNodeId>
  <TopSellers>
    <TopSeller>
      <ASIN>B00023RSUA</ASIN>
      <Title>Black & Decker CCC3000 18-Volt Cordless Electric Lawncare Cen
ter</Title>
    </TopSeller>
    <TopSeller>
      <ASIN>B001KXZ808</ASIN>
      <Title>EatSmart Precision Digital Bathroom Scale w/ Extra Large Backlit
3.5" Display and "Step-On" Technology</Title>
    </TopSeller>
  </TopSellers>
  <NewReleases>
    <NewRelease>
      <ASIN>B004NXUAXW</ASIN>
      <Title>Ozeri Touch Digital Bathroom Scale -- Measures Weight, Body Fat,
Hydration, Muscle and Bone Mass with Auto Recognition Technology for 8 Personal
Profiles</Title>
    </NewRelease>
    <NewRelease>
      <ASIN>B000SQK3QW</ASIN>
      <Title>Costless Pet Treats Duck Breast Fillets, 32-Ounce</Title>
    </NewRelease>
  </NewReleases>
</BrowseNode>
```

Discover Variations

The following request uses the `ItemLookup` operation with the `VariationMatrix` response group. The request finds variations of ASIN `B003FVODV8` (Nike Air Flightposite men's shoes).

Example Request

```
http://webservices.amazon.com/onca/xml?
  Service=AWSECommerceService
  &Operation=ItemLookup
  &ResponseGroup=VariationMatrix
  &IdType=ASIN
  &ItemId=B003FVODV8
  &AssociateTag=[YourAssociateTag]
  &AWSAccessKeyId=[YourAWSAccessKeyId]
  &Timestamp=[YYYY-MM-DDThh:mm:ssZ]
  &Signature=[RequestSignature]
```

Example Response

The following response returns three item variations.

```
<Item>
  <ASIN>B003FVODV8</ASIN>
  <Variations>
    <VariationDimensions>
      <VariationDimension>Size</VariationDimension>
      <VariationDimension>Color</VariationDimension>
    </VariationDimensions>
    <Item>
      <ASIN>B0033P8NU8</ASIN>
      <ItemAttributes>
        <Color>Black / Black-Black</Color>
        <Size>10 D(M) US</Size>
      </ItemAttributes>
      <VariationAttributes>
        <VariationAttribute>
          <Name>Size</Name>
          <Value>10 D(M) US</Value>
        </VariationAttribute>
        <VariationAttribute>
          <Name>Color</Name>
          <Value>Black / Black-Black</Value>
        </VariationAttribute>
      </VariationAttributes>
    </Item>
    <Item>
      <ASIN>B0033P4SF2</ASIN>
      <ItemAttributes>
        <Color>Black / Black-Black</Color>
        <Size>11.5 D(M) US</Size>
      </ItemAttributes>
      <VariationAttributes>
        <VariationAttribute>
          <Name>Size</Name>
          <Value>11.5 D(M) US</Value>
        </VariationAttribute>
        <VariationAttribute>
          <Name>Color</Name>
          <Value>Black / Black-Black</Value>
        </VariationAttribute>
      </VariationAttributes>
    </Item>
    <Item>
      <ASIN>B0039ONLL4</ASIN>
      <ItemAttributes>
        <Color>Metallic Zinc / Metallic Zinc-Black</Color>
        <Size>11.5 D(M) US</Size>
      </ItemAttributes>
      <VariationAttributes>
        <VariationAttribute>
          <Name>Size</Name>
          <Value>11.5 D(M) US</Value>
        </VariationAttribute>
        <VariationAttribute>
          <Name>Color</Name>
```

```
<Value>Metallic Zinc / Metallic Zinc-Black</Value>
</VariationAttribute>
</VariationAttributes>
</Item>
</Variations>
</Item>
```

Find Related Items

The following request uses the `ItemLookup` operation with the `RelatedItems` response group to find music tracks for ASIN `B00136LUWW` (*Bob Dylan's Greatest Hits*).

The `RelationshipType` parameter specifies how items are related. This parameter is required with the `RelatedItems` response group.

When the `RelationshipType` parameter is set to `Tracks`, the API finds the relationship between the MP3 Album (parent) and its Tracks (children). You can also find the reverse relationship; specify a track to find the parent album.

Example Request

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService
&Operation=ItemLookup
&ResponseGroup=RelatedItems,Small
&IdType=ASIN
&ItemId=B00136LUWW
&RelationshipType=Tracks
&AssociateTag=[YourAssociateTag]
&AWSAccessKeyId=[YourAWSAccessKeyId]
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[RequestSignature]
```

Example Response

The following response returns the children items (tracks) for the parent item (album).

```
<Item>
  <ASIN>B00136LUWW</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Bob-Dylans-Greatest-Hits/dp/B00136LUWW%3FAWSAccessKey
    Id%3D[Your_AccessKeyId]%26tag%3D[Your_AssociateTag]%26link
    Code%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3DB00136LUWW
  </DetailPageURL>
  <ItemAttributes>
    <Creator Role="Primary Contributor">Bob Dylan</Creator>
    <Manufacturer>Columbia</Manufacturer>
    <ProductGroup>Digital Music Album</ProductGroup>
    <Title>Bob Dylan's Greatest Hits</Title>
  </ItemAttributes>
  <RelatedItems>
    <Relationship>Children</Relationship>
    <RelationshipType>Tracks</RelationshipType>
    <RelatedItemCount>10</RelatedItemCount>
    <RelatedItemPageCount>1</RelatedItemPageCount>
    <RelatedItemPage>1</RelatedItemPage>
    <RelatedItem>
      <Item>
        <ASIN>B00136NUSY</ASIN>
        <ItemAttributes>
          <Creator Role="Primary Contributor">Bob Dylan</Creator>
          <Manufacturer>Columbia</Manufacturer>
          <ProductGroup>Digital Music Track</ProductGroup>
          <Title>Just Like A Woman</Title>
        </ItemAttributes>
      </Item>
    </RelatedItem>
    <RelatedItem>
      <Item>
        <ASIN>B00136LTN2</ASIN>
        <ItemAttributes>
          <Creator Role="Primary Contributor">Bob Dylan</Creator>
          <Manufacturer>Columbia</Manufacturer>
          <ProductGroup>Digital Music Track</ProductGroup>
          <Title>Rainy Day Women #12 & 35</Title>
        </ItemAttributes>
      </Item>
    </RelatedItem>
    <RelatedItem>
      <Item>
        <ASIN>B00136JOO8</ASIN>
        <ItemAttributes>
          <Creator Role="Primary Contributor">Bob Dylan</Creator>
          <Manufacturer>Columbia</Manufacturer>
          <ProductGroup>Digital Music Track</ProductGroup>
          <Title>Blowin' In The Wind</Title>
        </ItemAttributes>
      </Item>
    </RelatedItem>
  </RelatedItems>
</Item>
```

Find Similar Items

The following request uses the `Similarity` operation with the `Small` response group. The request finds items similar to ASIN `B000W7JWUA` (Settlers of Catan board game).

Example Request

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService  
&Operation=SimilarityLookup  
&ResponseGroup=Small  
&IdType=ASIN  
&ItemId=B000W7JWUA  
&AssociateTag=[YourAssociateTag]  
&AWSAccessKeyId=[YourAWSAccessKeyId]  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[RequestSignature]
```

Example Response

The following response returns three similar items.

```
<Item>
  <ASIN>B000W7JWYG</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Settlers-Catan-Player-Extension/dp/B000W7JWYG%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_AssociateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3DB000W7JWYG
  </DetailPageURL>
  <ItemAttributes>
    <Manufacturer>Mayfair Games</Manufacturer>
    <ProductGroup>Toy</ProductGroup>
    <Title>The Settlers of Catan 5 and 6 Player Extension</Title>
  </ItemAttributes>
</Item>
<Item>
  <ASIN>B000W7G78A</ASIN>
  <DetailPageURL>
    http://www.amazon.com/MayFair-Games-MFG3063-Catan-Sea
    farers/dp/B000W7G78A%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_AssociateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS
    IN%3DB000W7G78A
  </DetailPageURL>
  <ItemAttributes>
    <Manufacturer>Mayfair Games</Manufacturer>
    <ProductGroup>Toy</ProductGroup>
    <Title>Catan: Seafarers Game Expansion</Title>
  </ItemAttributes>
</Item>
<Item>
  <ASIN>B000W7I7EW</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Catan-Cities-Knights-Game-Expansion/dp/B000W7I7EW%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_AssociateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS
    IN%3DB000W7I7EW
  </DetailPageURL>
  <ItemAttributes>
    <Manufacturer>Mayfair Games</Manufacturer>
    <ProductGroup>Toy</ProductGroup>
    <Title>Catan: Cities and Knights Game Expansion</Title>
  </ItemAttributes>
</Item>
```

Example Request

The following request uses the `Similarity` operation to find items similar to multiple ASINs in the request.

When the `SimilarityType` parameter is set to `Random`, items are picked randomly from all similar items. Repeating the operation can return different results.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService  
&Operation=SimilarityLookup  
&ResponseGroup=Small  
&ItemId=B000W7JWYG,B000W7G78A,B000W7I7EW  
&SimilarityType=Random  
&AssociateTag=[YourAssociateTag]  
&AWSAccessKeyId=[YourAWSAccessKeyId]  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[RequestSignature]
```

Example Response

The following response returns five similar items.

```
<Item>
    <ASIN>B00F1BWZBA</ASIN>
    <DetailPageURL>http://www.amazon.com/Catan-Scenarios-Helpers-
of/dp/B00F1BWZBA%3FSubscriptionId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_Associ-
ateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS
IN%3DB00F1BWZBA</DetailPageURL>
    <ItemAttributes>
        <Manufacturer>Flat River Group</Manufacturer>
        <ProductGroup>Toy</ProductGroup>
        <Title>Catan Scenarios Helpers of Catan</Title>
    </ItemAttributes>
    <Item>
        <ASIN>B003NX73F6</ASIN>
        <DetailPageURL>http://www.amazon.com/Catan-Histories-Settlers-
America-Trails/dp/B003NX73F6%3FSubscriptionId%3D[Your_AWSAccessKeyId]%
26tag%3D[Your_AssociateTag]&%26linkCode%3Dxm2%26camp%3D2025%26creat
ive%3D165953%26creativeASIN%3DB003NX73F6</DetailPageURL>
        <ItemAttributes>
            <Manufacturer>Mayfair Games</Manufacturer>
            <ProductGroup>Toy</ProductGroup>
            <Title>Catan Histories: Settlers of America Trails to
Rails</Title>
        </ItemAttributes>
    </Item>
    <Item>
        <ASIN>B009B5GYP0</ASIN>
        <DetailPageURL>http://www.amazon.com/Mayfair-Games-MFG-73002-
Catan/dp/B009B5GYP0%3FSubscriptionId%3D[Your_AWSAccessKeyId]%
26tag%3D[Your_AssociateTag]&%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS
IN%3DB009B5GYP0</DetailPageURL>
        <ItemAttributes>
            <Manufacturer>Flat River Group</Manufacturer>
            <ProductGroup>Toy</ProductGroup>
            <Title>Catan: Family Edition</Title>
        </ItemAttributes>
    </Item>
    <Item>
        <ASIN>B000W7JX9U</ASIN>
        <DetailPageURL>http://www.amazon.com/Catan-Seafarers-Player-Exten-
sion-Edition/dp/B000W7JX9U%3FSubscriptionId%3D[Your_AWSAccessKeyId]%
26tag%3D[Your_AssociateTag]&%26linkCode%3Dxm2%26camp%3D2025%26creat
ive%3D165953%26creativeASIN%3DB000W7JX9U</DetailPageURL>
        <ItemAttributes>
            <Manufacturer>Mayfair Games</Manufacturer>
            <ProductGroup>Toy</ProductGroup>
            <Title>Catan: Seafarers 5&6 Player Extension 4th Edition</Title>
        </ItemAttributes>
    </Item>
    <Item>
        <ASIN>B000W7G78A</ASIN>
        <DetailPageURL>http://www.amazon.com/Catan-Seafarers-Game-Expansion-
Edition/dp/B000W7G78A%3FSubscriptionId%3D[Your_AWSAccessKeyId]%
26tag%3D[Your_AssociateTag]&%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS
IN%3DB000W7G78A</DetailPageURL>
    </Item>

```

```
IN%3DB000W7G78A</DetailPageURL>
<ItemAttributes>
    <Manufacturer>Mayfair Games</Manufacturer>
    <ProductGroup>Toy</ProductGroup>
    <Title>Catan: Seafarers Game Expansion 4th Edition</Title>
</ItemAttributes>
</Item>
```

Lookup by ISBN

The following request uses the `ItemLookup` operation to find an item by ISBN. The `IdType` parameter is set to `ISBN`, and the `ItemId` value is `076243631X`.

Example Request

```
http://webservices.amazon.com/onca/xml?
    Service=AWSECommerceService
    &Operation=ItemLookup
    &ResponseGroup=Large
    &SearchIndex=All
    &IdType=ISBN
    &ItemId=076243631X
    &AWSAccessKeyId=[Your_AWSAccessKeyId]
    &AssociateTag=[Your_AssociateTag]
    &Timestamp=[YYYY-MM-DDThh:mm:ssZ]
    &Signature=[Request_Signature]
```

Example Response

The following response returns an item that matches the ISBN (*The Mammoth Book of Tattoos*).

```
<Item>
  <ASIN>076243631X</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Mammoth-Book-Tattoos-Lal-
    Hardy/dp/076243631X%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_As-
    sociateTag]%26linkCode%3Dxm%26camp%3D2025%26creative%3D165953%26creativeAS-
    IN%3D076243631X
  </DetailPageURL>
  <SalesRank>7483</SalesRank>
  <ItemAttributes>
    <Author>Lal Hardy</Author>
    <Binding>Paperback</Binding>
    <DeweyDecimalNumber>391.65</DeweyDecimalNumber>
    <EAN>9780762436316</EAN>
    <Feature>ISBN13: 9780762436316</Feature>
    <Feature>Condition: New</Feature>
    <ISBN>076243631X</ISBN>
    <Manufacturer>Running Press</Manufacturer>
    <NumberOfItems>1</NumberOfItems>
    <NumberOfPages>512</NumberOfPages>
    <ProductGroup>Book</ProductGroup>
    <PublicationDate>2009-03-10</PublicationDate>
    <Publisher>Running Press</Publisher>
    <Studio>Running Press</Studio>
    <Title>The Mammoth Book of Tattoos</Title>
  </ItemAttributes>
</Item>
```

Lookup by UPC

This following request uses the `ItemLookup` operation to find an item by UPC. The `ItemId` parameter is set to UPC, and the `ItemId` value is 635753490879.

Example Request

```
http://webservices.amazon.com/onca/xml?
  Service=AWSECommerceService
  &Operation=ItemLookup
  &ResponseGroup=Large
  &SearchIndex=All
  &IdType=UPC
  &ItemId=635753490879
  &AWSAccessKeyId=[Your_AWSAccessKeyId]
  &AssociateTag=[Your_AssociateTag]
  &Timestamp=[YYYY-MM-DDThh:mm:ssZ]
  &Signature=[Request_Signature]
```

Example Response

The following response returns an item that matches the UPC (Samsung Galaxy tablet).

```
<Item>
  <ASIN>B004U9USEA</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Samsung-GT-P1010CWAXAR-Galaxy-Tab-Wi-
    Fi/dp/B004U9USEA%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_Associ-
    ateTag]%26linkCode%3Dxm%26camp%3D2025%26creative%3D165953%26creativeAS
    IN%3DB004U9USEA
  </DetailPageURL>
  <SalesRank>6</SalesRank>
  <ItemAttributes>
    <Binding>Personal Computers</Binding>
    <Brand>Samsung</Brand>
    <Color>Black/White</Color>
    <CPUManufacturer>Intel</CPUManufacturer>
    <CPUSpeed Units="GHz">2.1</CPUSpeed>
    <CPUType>Intel Pentium 4</CPUType>
    <DisplaySize Units="inches">10.1</DisplaySize>
    <EAN>0635753490879</EAN>
    <Feature>Android Froyo (2.2); CPU: 1GHz A8 Cortex Processor;</Feature>
    <Feature>Battery: Li-Polymer, 4000mAh</Feature>
    <Feature>Memory: 16GB Internal Memory; Up to 32GB Expandable Memory (mi-
    croSD)</Feature>
    <Feature>WiFi: 802.11 a/b/g/n; USB 2.0; Bluetooth 2.1 Connectivity</Feature>

    <Feature>Audio: MP3, OGG, AAC, AMR-NB, AMR-WB, WMA, FLAC, WAV, MID, AC3,
    RTTTL/RTX, OTA, i-Melody, SP-MIDI</Feature>
    <Feature>Display: 7.0" WSVGA Display; 600 x 1024 Pixel Display Resolution;
    3.54" x 6.05" Display Size; 16M TFT Display Technology</Feature>
    <Feature>Video: Codec: Mpeg4, H.264, H.263, Divx/XviD Format: 3gp(mp4),
    wmv(asf), avi(divx), mkv, flv</Feature>
    <HardDiskSize Units="GB">1</HardDiskSize>
    <ListPrice>
      <Amount>36999</Amount>
      <CurrencyCode>USD</CurrencyCode>
      <FormattedPrice>$369.99</FormattedPrice>
    </ListPrice>
    <Manufacturer>Samsung IT</Manufacturer>
    <Model>GT-P1010CWAXAR</Model>
    <MPN>GT-P1010CWAXAR</MPN>
    <Title>Samsung Galaxy Tab (Wi-Fi)</Title>
    <UPC>635753490879</UPC>
  </ItemAttributes>
</Item>
```

Retrieve Customer Reviews

The following request uses the `ItemLookup` operation with the `Reviews` response group. The request retrieves customer reviews for ASIN `B004HFS6Z0` (Kindle Wi-Fi).

Example Request

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService  
&Operation=ItemLookup  
&ResponseGroup=Reviews  
&IdType=ASIN  
&ItemId=B004HFS6Z0  
&AssociateTag=[YourAssociateTag]  
&AWSAccessKeyId=[YourAWSAccessKeyId]  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[RequestSignature]
```

Example Response

The following response returns the URL that displays customer reviews.

```
<Item>  
  <ASIN>B004HFS6Z0</ASIN>  
  <CustomerReviews>  
    <IFrameURL>  
      http://www.amazon.com/reviews/iframe?akid=[YourAWSAccessKeyId]&alink  
Code=xm2&asin=B004HFS6Z0&atag=[YourAssociateTag]&exp=2011-06-  
01T22%3A32%3A53Z&v=2&sig=pxn6bbln%2B%2FVTPJdj8oCcXvjTHmo3spkUMjbQMPbhCKI%3D  
    </IFrameURL>  
    <HasReviews>true</HasReviews>  
  </CustomerReviews>  
</Item>
```

Retrieve Images

The following request uses the ItemLookup operation with the Images response group. The request retrieves images for ASIN B004H06I4M (*Rio, the movie*).

Example Request

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService  
&Operation=ItemLookup  
&ResponseGroup=Images  
&IdType=ASIN&  
&ItemId=B004H06I4M  
&AWSAccessKeyId=[YourAWSAccessKeyId]  
&AssociateTag=[YourAssociateTag]  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[RequestSignature]
```

Example Response

The following response returns image details for the item.

```
<Item>
<ASIN>B004HO6I4M</ASIN>
<SmallImage>
<URL>
    http://ecx.images-amazon.com/images/I/519SgX2wwDL._SL75_.jpg
</URL>
<Height Units="pixels">75</Height>
<Width Units="pixels">56</Width>
</SmallImage>
<MediumImage>
<URL>
    http://ecx.images-amazon.com/images/I/519SgX2wwDL._SL160_.jpg
</URL>
<Height Units="pixels">160</Height>
<Width Units="pixels">120</Width>
</MediumImage>
<LargeImage>
<URL>
    http://ecx.images-amazon.com/images/I/519SgX2wwDL._SL500_.jpg
</URL>
<Height Units="pixels">500</Height>
<Width Units="pixels">375</Width>
</LargeImage>
<ImageSets>
<ImageSet Category="primary">
<SwatchImage>
<URL>
    http://ecx.images-amazon.com/images/I/519SgX2wwDL._SL30_.jpg
</URL>
<Height Units="pixels">30</Height>
<Width Units="pixels">22</Width>
</SwatchImage>
<SmallImage>
<URL>
    http://ecx.images-amazon.com/images/I/519SgX2wwDL._SL75_.jpg
</URL>
<Height Units="pixels">75</Height>
<Width Units="pixels">56</Width>
</SmallImage>
<ThumbnailImage>
<URL>
    http://ecx.images-amazon.com/images/I/519SgX2wwDL._SL75_.jpg
</URL>
<Height Units="pixels">75</Height>
<Width Units="pixels">56</Width>
</ThumbnailImage>
<TinyImage>
<URL>
    http://ecx.images-amazon.com/images/I/519SgX2wwDL._SL110_.jpg
</URL>
<Height Units="pixels">110</Height>
<Width Units="pixels">82</Width>
</TinyImage>
<MediumImage>
```

```
<URL>
    http://ecx.images-amazon.com/images/I/519SgX2wwDL._SL160_.jpg
</URL>
<Height Units="pixels">160</Height>
<Width Units="pixels">120</Width>
</MediumImage>
<LargeImage>
    <URL>
        http://ecx.images-amazon.com/images/I/519SgX2wwDL._SL500_.jpg
    </URL>
    <Height Units="pixels">500</Height>
    <Width Units="pixels">375</Width>
</LargeImage>
</ImageSet>
</ImageSets>
</Item>
```

Images are returned in two ways:

- The images under `Item` are specified by the `SmallImage`, `MediumImage`, and `LargeImage` child elements.
- The image sizes under `ImageSets` are specified by the `_SLXXX` suffix in the URL, where `XXX` is the number of pixels on the longest side of the image. For example, the `LargeImage` has 500 pixels on its longest side, so it has the suffix `_SL500_.jpg`. When you reference images, include the `_SLXXX` suffix.

The `ImageSets` element attribute `Category` is set to `Primary`. Primary images are those shown in the `Item` section.

Retrieve Price

The following request uses the `ItemLookup` operation with the `Offers` response group. The request finds the price for ASIN `B00KOKTZLQ` (ASICS Men's GEL-Venture 4 running shoes).

Example Request

```
http://webservices.amazon.com/onca/xml?
    Service=AWSECommerceService
    &Operation=ItemLookup
    &ResponseGroup=Offers
    &IdType=ASIN
    &ItemId=B00KOKTZLQ
    &AssociateTag=[YourAssociateTag]
    &AWSAccessKeyId=[YourAWSAccessKeyId]
    &Timestamp=[YYYY-MM-DDThh:mm:ssZ]
    &Signature=[RequestSignature]
```

Example Response

The following response returns the price information for the item.

```
<Item>
  <ASIN>B00KOKTZLQ</ASIN>
  <OfferSummary>
    <LowestNewPrice>
      <Amount>3998</Amount>
      <CurrencyCode>USD</CurrencyCode>
      <FormattedPrice>$39.98</FormattedPrice>
    </LowestNewPrice>
    <TotalNew>4</TotalNew>
    <TotalUsed>0</TotalUsed>
    <TotalCollectible>0</TotalCollectible>
    <TotalRefurbished>0</TotalRefurbished>
  </OfferSummary>
  <Offers>
    <TotalOffers>1</TotalOffers>
    <TotalOfferPages>1</TotalOfferPages>
    <MoreOffersUrl>
      http://www.amazon.com/gp/offer-listing/B00KOKTZLQ/?&AWSAccessKey
      Id=[Your_AWSAccessKeyId]&ie=UTF8&tag=[Your_AssociateTag]26link
      Code%3Dxm%26camp%3D2025%26creative%3D386001%26creativeASIN%3DB00KOKTZLQ
    </MoreOffersUrl>
    <Offer>
      <OfferAttributes>
        <Condition>New</Condition>
      </OfferAttributes>
      <OfferListing>
        <OfferListingId>
          LYcccWDl3oNWXKKngMuydLjjkX
        </OfferListingId>
        <Price>
          <Amount>6000</Amount>
          <CurrencyCode>USD</CurrencyCode>
          <FormattedPrice>$60.00</FormattedPrice>
        </Price>
        <SalePrice>
          <Amount>4495</Amount>
          <CurrencyCode>USD</CurrencyCode>
          <FormattedPrice>$44.95</FormattedPrice>
        </SalePrice>
        <AmountSaved>
          <Amount>1505</Amount>
          <CurrencyCode>USD</CurrencyCode>
          <FormattedPrice>$15.05</FormattedPrice>
        </AmountSaved>
        <PercentageSaved>25</PercentageSaved>
        <Availability>Usually ships in 1-2 business days</Availability>
        <AvailabilityAttributes>
          <AvailabilityType>now</AvailabilityType>
          <MinimumHours>24</MinimumHours>
          <MaximumHours>48</MaximumHours>
        </AvailabilityAttributes>
        <IsEligibleForSuperSaverShipping>0</IsEligibleForSuperSaverShipping>
      </OfferListing>
    </Offer>
  </Offers>
</Item>
```

```
<IsEligibleForPrime>0</IsEligibleForPrime>
</OfferListing>
</Offer>
</Offers>
</Item>
```

Search by Actor

The following request uses the `ItemSearch` operation to find items in the DVD search index. The `Actor` parameter has the value, `Tom Hanks`.

Example Request

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService
&Operation=ItemSearch
&ResponseGroup=Small
&SearchIndex=DVD
&Actor=Tom Hanks
&AWSAccessKeyId=[Your_AWSAccessKeyId]
&AssociateTag=[Your_AssociateTag]
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request_Signature]
```

Example Response

The following response returns two items.

```
<Item>
  <ASIN>B00275EHJG</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Toy-Story-3-Tom-Hanks/dp/B00275EHJG%3FAWSAccessKey
    Id%3D[Your_AWSAccessKeyId]%26tag%3D[Your_AssociateTag]%26link
    Code%3Dxm%26camp%3D2025%26creative%3D165953%26creativeASIN%3DB00275EHJG
  </DetailPageURL>
  <ItemAttributes>
    <Actor>Tom Hanks</Actor>
    <Actor>Tim Allen</Actor>
    <Director>Lee Unkrich</Director>
    <Manufacturer>Disney*Pixar</Manufacturer>
    <ProductGroup>DVD</ProductGroup>
    <Title>Toy Story 3</Title>
  </ItemAttributes>
</Item>
<Item>
  <ASIN>B0030IIZ4M</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Toy-Story-Tim-Allen/dp/B0030IIZ4M%3FAWSAccessKey
    Id%3D[Your_AWSAccessKeyId]%26tag%3D[Your_AssociateTag]%26link
    Code%3Dxm%26camp%3D2025%26creative%3D165953%26creativeASIN%3DB0030IIZ4M
  </DetailPageURL>
  <ItemAttributes>
    <Actor>Tim Allen</Actor>
    <Actor>Tom Hanks</Actor>
    <Actor>Annie Potts</Actor>
    <Actor>John Ratzenberger</Actor>
    <Actor>Don Rickles</Actor>
    <Manufacturer>Disney*Pixar</Manufacturer>
    <ProductGroup>DVD</ProductGroup>
    <Title>Toy Story</Title>
  </ItemAttributes>
</Item>
```

Search by Author

The following request uses the `ItemSearch` operation to find items in the Books search index. The `Author` parameter has the value, J.K. Rowling.

Example Request

```
http://webservices.amazon.com/onca/xml?
  Service=AWSECommerceService
  &Operation=ItemSearch
  &ResponseGroup=Small
  &SearchIndex=Books
  &Author=J.K.Rowling
  &AWSAccessKeyId=[Your_AWSAccessKeyId]
  &AssociateTag=[Your_AssociateTag]
  &Timestamp=[YYYY-MM-DDThh:mm:ssZ]
  &Signature=[Request_Signature]
```

Example Response

The following response returns two items.

```
<Item>
  <ASIN>0545139708</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Harry-Potter-Deathly-Hallows-
    Book/dp/0545139708%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_Aso-
    ciateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS
    IN%3D0545139708
  </DetailPageURL>
  <ItemAttributes>
    <Author>J.K. Rowling</Author>
    <Manufacturer>Arthur A. Levine Books</Manufacturer>
    <ProductGroup>Book</ProductGroup>
    <Title>Harry Potter and the Deathly Hallows (Book 7)</Title>
  </ItemAttributes>
</Item>
<Item>
  <ASIN>0545162076</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Harry-Potter-Paperback-Box-
    Books/dp/0545162076%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_Aso-
    ciateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS
    IN%3D0545162076
  </DetailPageURL>
  <ItemAttributes>
    <Author>J.K. Rowling</Author>
    <Manufacturer>Arthur A. Levine Books</Manufacturer>
    <ProductGroup>Book</ProductGroup>
    <Title>Harry Potter Paperback Box Set (Books 1-7)</Title>
  </ItemAttributes>
</Item>
```

Search by Keyword

The following request uses the `ItemSearch` operation to find items by keyword. The `Keywords` parameter has the value, `harry potter`. To search the entire Amazon catalog, use the `All` search index.

Example Request

```
http://webservices.amazon.com/onca/xml?
  Service=AWSECommerceService
  &Operation=ItemSearch
  &ResponseGroup=Small
  &SearchIndex=All
  &Keywords=harry_potter
  &AWSAccessKeyId=[Your_AWSAccessKeyId]
  &AssociateTag=[Your_AssociateTag]
  &Timestamp=[YYYY-MM-DDThh:mm:ssZ]
  &Signature=[Request_Signature]
```

Example Response

The following response returns three items.

```
<MoreSearchResultsUrl>
    http://www.amazon.com/gp/redirect.html?camp=2025&creative=386001&location=ht
    tp%3A%2F%2Fwww.amazon.com%2Fgp%2Fsearch%3Fkeywords%3Dharry%2Bpot
    ter%26url%3Dsearch-alias%253Daws-amazon-aps&linkCode=xm2&tag=[Your_Associat
    eTag]&AWSAccessKeyId=[Your_AWSAccessKeyId]
</MoreSearchResultsUrl>
<Item>
    <ASIN>B005OCFGTO</ASIN>
    <DetailPageURL>
        http://www.amazon.com/Harry-Potter-Complete-8-Film-Collection/dp/B005OCFG
        TO%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_AssociateTag]%26link
        Code%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3DB005OCFGTO
    </DetailPageURL>
    <ItemAttributes>
        <Actor>Daniel Radcliffe</Actor>
        <Actor>Rupert Grint</Actor>
        <Actor>Emma Watson</Actor>
        <Actor>Robbie Coltrane</Actor>
        <Actor>Maggie Smith</Actor>
        <Director>Chris Columbus</Director>
        <Director>Alfonso Cuaron</Director>
        <Director>Mike Newell</Director>
        <Director>David Yates</Director>
        <Manufacturer>Warner Bros.</Manufacturer>
        <ProductGroup>DVD</ProductGroup>
        <Title>Harry Potter: The Complete 8-Film Collection</Title>
    </ItemAttributes>
</Item>
<Item>
    <ASIN>059035342X</ASIN>
    <DetailPageURL>
        http://www.amazon.com/Harry-Potter-Sorcerers-Stone-
        Book/dp/059035342X%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_Aso
        ciateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS
        IN%3D059035342X
    </DetailPageURL>
    <ItemAttributes>
        <Author>J.K. Rowling</Author>
        <Creator Role="Illustrator">Mary GrandPré</Creator>
        <Manufacturer>Scholastic Paperbacks</Manufacturer>
        <ProductGroup>Book</ProductGroup>
        <Title>Harry Potter and the Sorcerer's Stone (Book 1)</Title>
    </ItemAttributes>
</Item>
<Item>
    <ASIN>B002DQLGHU</ASIN>
    <DetailPageURL>
        http://www.amazon.com/Harry-Potter-Sorcerers-Stone/dp/B002DQLGHU%3FAWSAc
        cessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_AssociateTag]%26link
        Code%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3DB002DQLGHU
    </DetailPageURL>
    <ItemAttributes>
        <Actor>Daniel Radcliffe</Actor>
        <Actor>Rupert Grint</Actor>
```

```
<Actor>Emma Watson</Actor>
<Actor>John Cleese</Actor>
<Actor>Robbie Coltrane</Actor>
<Creator Role="Producer">David Heyman</Creator>
<Creator Role="Writer">Steve Kloves</Creator>
<Creator Role="Writer">J.k. Rowling</Creator>
<Director>Chris Columbus</Director>
<ProductGroup>Movie</ProductGroup>
<Title>Harry Potter and the Sorcerer's Stone</Title>
</ItemAttributes>
</Item>
```

Search by Manufacturer

The following request uses the `ItemSearch` operation to find electronics by manufacturer in the `Electronics` search index. The `Manufacturer` parameter has the value, `Sony`.

Example Request

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService
&Operation=ItemSearch
&ResponseGroup=Small
&SearchIndex=Electronics
&Manufacturer=Sony
&AWSAccessKeyId=[Your_AWSAccessKeyId]
&AssociateTag=[Your_AssociateTag]
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request_Signature]
```

Example Response

The following response returns two items.

```
<Item>
  <ASIN>B0015AARJI</ASIN>
  <DetailPageURL>
    http://www.amazon.com/PlayStation-3-Dualshock-Wireless-Control
    ler/dp/B0015AARJI%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_Aso
    ciateTag]%26linkCode%3Dxm%26camp%3D2025%26creative%3D165953%26creativeAS
    IN%3DB0015AARJI
  </DetailPageURL>
  <ItemAttributes>
    <Manufacturer>Sony Computer Entertainment</Manufacturer>
    <ProductGroup>Video Games</ProductGroup>
    <Title>PlayStation 3 Dualshock 3 Wireless Controller</Title>
  </ItemAttributes>
</Item>
<Item>
  <ASIN>B003VUO6H4</ASIN>
  <DetailPageURL>
    http://www.amazon.com/PlayStation-3-160-GB/dp/B003VUO6H4%3FAWSAccessKey
    Id%3D[Your_AWSAccessKeyId]%26tag%3D[Your_AssociateTag]%26link
    Code%3Dxm%26camp%3D2025%26creative%3D165953%26creativeASIN%3DB003VUO6H4
  </DetailPageURL>
  <ItemAttributes>
    <Manufacturer>Sony</Manufacturer>
    <ProductGroup>Video Games</ProductGroup>
    <Title>PlayStation 3 160 GB</Title>
  </ItemAttributes>
</Item>
```

Search by Theme

The following request uses the ItemSearch operation to find items related to a theme. The Keywords parameter has the value, Travel.

Example Request

```
http://webservices.amazon.com/onca/xml?
  Service=AWSECommerceService
  &Operation=ItemSearch
  &ResponseGroup=Small
  &SearchIndex=All
  &Keywords=Travel
  &AWSAccessKeyId=[Your_AWSAccessKeyId]
  &AssociateTag=[Your_AssociateTag]
  &Timestamp=[YYYY-MM-DDThh:mm:ssZ]
  &Signature=[Request_Signature]
```

Example Response

The following response returns two items.

```
<Item>
  <ASIN>1426201257</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Journeys-Lifetime-Worlds-Greatest-
    Trips/dp/1426201257%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_As-
    sociateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS-
    IN%3D1426201257
  </DetailPageURL>
  <ItemAttributes>
    <Author>National Geographic</Author>
    <Manufacturer>National Geographic</Manufacturer>
    <ProductGroup>Book</ProductGroup>
    <Title>Journeys of a Lifetime: 500 of the World's Greatest Trips</Title>
  </ItemAttributes>
</Item>
<Item>
  <ASIN>B003EM7MP2</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Travel-Accessories-Samsonite-Space-Medi-
    um/dp/B003EM7MP2%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_Associ-
    ateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS-
    IN%3DB003EM7MP2
  </DetailPageURL>
  <ItemAttributes>
    <ProductGroup>Apparel</ProductGroup>
    <Title>Travel Accessories Samsonite Space Saver Bags Medium Packs Set of 4
    Clear</Title>
  </ItemAttributes>
</Item>
```

Search by Theme in a Category

The following request uses the ItemSearch operation to find items related to the theme, Travel. To specify the category, the request uses the Books search index.

Example Request

```
http://webservices.amazon.com/onca/xml?
  Service=AWSECommerceService
  &Operation=ItemSearch
  &ResponseGroup=Small
  &SearchIndex=Books
  &Keywords=Travel
  &AWSAccessKeyId=[Your_AWSAccessKeyId]
  &AssociateTag=[Your_AssociateTag]
  &Timestamp=[YYYY-MM-DDThh:mm:ssZ]
  &Signature=[Request_Signature]
```

Example Response

The following response returns two items.

```
<MoreSearchResultsUrl>
  http://www.amazon.com/gp/redirect.html?&AWSAccessKeyId=[Your_AWSAccessKeyId]
  &ie=UTF8&location=ht
  tp%3A%2F%2Fwww.amazon.com%2Fgp%2Fsearch%2Fkeywords%3DTravel%26ie%3DUTF8%26url%3Dsearch-
  alias%253Dstripbooks&tag=[Your_AssociateTag]&creative=386001&camp=2025&link
  Code=xm2
</MoreSearchResultsUrl>
<Item>
  <ASIN>1426201257</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Journeys-Lifetime-Worlds-Greatest-
    Trips/dp/1426201257%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_As
    sociateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS
    IN%3D1426201257
  </DetailPageURL>
  <ItemAttributes>
    <Author>National Geographic</Author>
    <Manufacturer>National Geographic</Manufacturer>
    <ProductGroup>Book</ProductGroup>
    <Title>Journeys of a Lifetime: 500 of the World's Greatest Trips</Title>
  </ItemAttributes>
</Item>
<Item>
  <ASIN>030740692X</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Once-Lifetime-Trips-Extraordinary-Experi
    ences/dp/030740692X%3FAWSAccessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[Your_As
    sociateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS
    IN%3D030740692X
  </DetailPageURL>
  <ItemAttributes>
    <Author>Chris Santella</Author>
    <Manufacturer>Clarkson Potter</Manufacturer>
    <ProductGroup>Book</ProductGroup>
    <Title>Once in a Lifetime Trips: The World's 50 Most Extraordinary and
    Memorable Travel Experiences</Title>
  </ItemAttributes>
</Item>
```

Search by Title

The following request uses the `ItemSearch` operation to find items in the `Music` search index. The `Title` parameter has the value, `Blue`.

Example Request

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService  
&Operation=ItemSearch  
&ResponseGroup=Small  
&SearchIndex=Music  
&Title=Blue  
&AWSAccessKeyId=[Your_AWSAccessKeyId]  
&AssociateTag=[Your_AssociateTag]  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request_Signature]
```

Example Response

The following response returns two items.

```
<MoreSearchResultsUrl>
  http://www.amazon.com/gp/redirect.html?
  AWSAccessKeyId=[Your_AWSAccessKeyId]
  &ie=UTF8
  &location=ht
  tp%3A%2F%2Fwww.amazon.com%2Fgp%2Fsearch%2F%3Fkeywords%3DBlue%26ie%3DUTF8%26url%3Dsearch-
  alias%253Dpopular
  &tag=[YourAssociateTag]
  &creative=386001
  &camp=2025
  &linkCode=xm2
</MoreSearchResultsUrl>
<Item>
  <ASIN>B004LL1HM4</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Helplessness-Blues-Fleet-
    Foxes/dp/B004LL1HM4%3FAWSAccessKeyId%[Your_AWSAccessKeyId]%26tag%3D[Your_Aso
    ciateTag]%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeAS
    IN%3DB004LL1HM4
  </DetailPageURL>
  <ItemAttributes>
    <Artist>Fleet Foxes</Artist>
    <Manufacturer>Sub Pop</Manufacturer>
    <ProductGroup>Music</ProductGroup>
    <Title>Helplessness Blues</Title>
  </ItemAttributes>
</Item>
<Item>
  <ASIN>B004AHNIGM</ASIN>
  <DetailPageURL>
    http://www.amazon.com/Low-Country-Blues-Gregg-Allman/dp/B004AHNIGM%3FAWSAC
    cessKeyId%3D[Your_AWSAccessKeyId]%26tag%3D[YourAssociateTag]%26link
    Code%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3DB004AHNIGM
  </DetailPageURL>
  <ItemAttributes>
    <Artist>Gregg Allman</Artist>
    <Manufacturer>Rounder</Manufacturer>
    <ProductGroup>Music</ProductGroup>
    <Title>Low Country Blues</Title>
  </ItemAttributes>
</Item>
```

Motivate Customers to Buy

Topics

- [Promotions \(p. 128\)](#)
- [Reviews \(p. 131\)](#)
- [Suggest Similar Items to Buy \(p. 133\)](#)
- [Top Sellers \(p. 140\)](#)
- [New Releases \(p. 143\)](#)

- [Sort by Popularity, Price, or Condition \(p. 144\)](#)

Product Advertising API operations and response groups can help drive sales.

Promotions

Topics

- [Promotion Categories \(p. 128\)](#)
- [Benefit and Eligible Items \(p. 129\)](#)
- [Common Response Elements \(p. 130\)](#)

Display discounted prices to help drive sales. The `PromotionSummary` response group can determine if an item has a promotion.

`PromotionSummary` returns the promotion information for an item.

Product Promotions

Buy 4 eligible items in the 4-for-3 promotion offered by Amazon.com and get 1 of them free. [Here's how](#) (restrictions apply)

Promotions are available only for items with offers. Items that do not have offers are Collection and Variation parent ASINs. When you request promotion information, you must include a response group that also returns offer information.

The following response groups return offer information:

- `Large`
- `OfferFull`
- `Offers`

The following operations return promotion information:

- `ItemLookup`
- `ItemSearch`
- `SimilarityLookup`

Note

You cannot use the Product Advertising API to create promotions. You can use the API to return promotion information for an item.

Related Topics

- [PromotionSummary Response Group \(p. 277\)](#)

Promotion Categories

The Product Advertising API supports the following promotion categories.

Category	Description
ForEachQuantityXGetQuantityFreeX	For a specified number of items, you receive additional items for free. For example, buy six dozen eggs and get a dozen eggs free.
BuyAmountXGetSimpleShippingFreeX	For a specified dollar amount, you receive free shipping. For example, spend \$25 and your item is shipped free of charge.
BuyAmountXGetAmountOffX	For a specified dollar amount, you receive a discounted price. For example, spend \$25 and get a \$5 discount.
BuyQuantityXGetAmountOffX	For a specified number of items, you receive a discounted price. For example, buy three balls and get a \$5 discount.
BuyQuantityXGetPercentOffX	For a specified number of items, you receive a percentage discount. For example, buy three balls and get a 15% discount.

Promotion categories appear in the response element `Category` as part of the `PromotionSummary` response.

Benefit and Eligible Items

Items returned in the promotion response group are related to the promotion in one of the following ways:

Benefit

The item is part of the promotional reward. For example, in a buy-one-camera-and-get-a-camera-case-free promotion, the camera case is the benefit item.

Eligible

The item is what the customer must buy to qualify for the promotion. For example, in a buy-one-camera-and-get-a-camera-case-free promotion, the camera is the eligible item.

Both

It's possible for an item to be the benefit item and the eligible item. For example, in a buy-two-shirts-get-the-third-shirt-half-off promotion, the shirt is the benefit item and the eligible item.

The promotion response groups use the `IsInBenefitSet` and `IsInEligibilitySet` elements to specify if an item is the benefit item or the eligible item.

```
<IsInBenefitSet>true</IsInBenefitSet>
<IsInEligibilityRequirementSet>true</IsInEligibilityRequirementSet>
```

The value for these elements is a boolean. For example, if `IsInBenefitSet` is true, the item is one of the benefits of the promotion, and the response will contain the `BenefitDescription` element, which describes the benefit item. The following example is a response for `BenefitDescription`.

```
<BenefitDescription>Save $25.00 when you spend $125.00 or more on Kitchen & Housewares or Bed & Bath products offered by Amazon.com. Enter code AUGSAVER at checkout.</BenefitDescription>
```

If `IsInEligibilitySet` is true, the response will have the `EligibilityDescription` element, which describes the eligible item.

Common Response Elements

The following response elements are commonly used to display promotion information.

Response Element	Description
BenefitDescription	Describes the benefit, which is the item(s) the customer receives for the promotion. The element will not appear if the item is not part of the promotional benefit.
EligibilityRequirement-Description	Describes the item(s) the customer must buy to qualify for the promotion. The element will not appear if the item does not qualify for the promotional benefit.
TermsAndConditions	Specifies the terms and conditions of the promotion.

The following response snippet shows the values for BenefitDescription.

```
<BenefitDescription>Save $25.00 when you spend $125.00 or more on Kitchen & Housewares or Bed & Bath products offered by Amazon.com. Enter code AUGSAVER at checkout.</BenefitDescription>
<TermsAndConditions><STRONG>To receive the Best Value discount:</STRONG> <OL>
<LI>Add $125 of qualifying Kitchen & Housewares or Bed & Bath products to your Shopping Cart via the <STRONG>Add to Shopping Cart</STRONG> button on each respective product information page. ...
</TermsAndConditions>
```

Response Element	Description
ComponentType	Specifies what the promotion applies to, such as Shipping, ItemPrice, and Subtotal.
CouponCombination-Type	Specifies the type of promotional coupons that can be combined. <ul style="list-style-type: none"> • Unrestricted—Can be used with other promotions. • Preferential—Can be used with some promotions. • Exclusive—Cannot be used with other promotions.
FormattedPrice	The price formatted for display.
IsInBenefitSet	A boolean value. When true, the associated ASIN is what the customer receives as part of the promotion.
IsInEligibilityRe-quirementSet	A boolean value. When true, the associated ASIN is what the customer must buy to qualify for the promotion.

The following response snippet shows the values for these elements.

```
<ComponentType>ItemPrice</ComponentType>
<CouponCombinationType>Unrestricted</CouponCombinationType>
<IsInBenefitSet>true</IsInBenefitSet>
<IsInEligibilityRequirementSet>true</IsInEligibilityRequirementSet>
```

The response shows the promotion is a price discount. The promotion can be combined with other promotions. To qualify, the customer must buy the item. The same item is also given as part of the promotion.

For more information, see [PromotionSummary Response Group \(p. 277\)](#).

Reviews

Topics

- [Get Editorial Reviews \(p. 131\)](#)
- [Get Customer Reviews \(p. 132\)](#)

Returned reviews are read-only. You can retrieve, but cannot create, reviews.

Get Editorial Reviews

The `EditorialReviews` response group provides the item description. This information is less of a customer review and more of a product description. The content of the `EditorialReviews` response group can be displayed individually as part of an `ItemLookup`, `ItemSearch`, or `SimilarityLookup` request.

Product Description

From the Manufacturer

This award-winning ride-on rocket features retro styling and Radio Flyer dependability. The classic red steel body is safe and durable. Kids love the astronaut and space sounds, vibrating motor action, clicking nose cone, and light up panel and after burner. Comfortable no-slip seat and covered front wheels make this car safe and sturdy.

For books, Editorial Reviews is a separate listing on the product detail page.

Editorial Reviews

The Midwest Book Review

SAVING MISS OLIVER'S is an engaging novel and is very highly recommended to all general fiction readers.

Bookwire, April 07, 2006

---Rich characters, unexpected plot development, and underlying themes of hope, lofty ideals, educational excellence, and historical pride.

Stephen Waters, Former School Head

---couldn't put it down. It was like a wire was attached to my brain, my memories, my very soul.

The following snippet shows part of the equivalent `EditorialReview` response.

```
<EditorialReview>
  <Source>Bookwire, April 07, 2006</Source>
  <Content>—Rich characters, unexpected plot development, and underlying themes
  of hope, lofty ideals, educational excellence, and historical pride.</Content>
</EditorialReview>
```

Note

Copyrighted editorial reviews are not returned. Reviews returned may be different from the reviews you see on the Amazon.com marketplace. For more information, see [EditorialReview \(p. 245\)](#).

Get Customer Reviews

The `Reviews` response group returns an iframe URL that displays customer reviews for an item.

Important

The iframe URL expires 24 hours from the date of the request. The expiration parameter appears in the URL. To get the latest iframe URL, you need to submit a new request. For example, when a customer views an item, your code should trigger a request to return the latest customer reviews.

The reviews are returned in the Customer Reviews section of the product detail page.

Customer Reviews

5,260 Reviews

5 star:		(2,526)
4 star:		(682)
3 star:		(450)
2 star:		(509)
1 star:		(1,093)

Average Customer Review
 (5,260 customer reviews)

Most Helpful Customer Reviews

1,666 of 1,942 people found the following review helpful:

Heartbreak of Heathcliff Proportions, August 3, 2008
By [J. Martin "Librarian"](#) (Dallas, TX) - [See all my reviews](#)
REVIEWER
This review is from: Breaking Dawn (The Twilight Saga, Book 4) (Hardcover)
I've only recently entered the Twilight fold. Having initially read reviews of the series in library journals and having heard passionate testimonials from avid fans, I thought I would give it a try.
Inexorably, I fell absolutely and positively in love with the first three Twilight books. I read them (the first time, that is) in three days. Then, like a junkie, I feverishly searched the media for news on the movie, the books, and all things Stephanie Meyers.
Stephenie Meyer's books were my brand of heroin.

The following snippet shows part of the equivalent Customer Reviews response returned by an `ItemLookup` or `ItemSearch` request.

```
<ItemLookupResponse>
  <Items>
    <Item>
      <ASIN>0316067938</ASIN>
      <CustomerReviews>
        <IFrameURL>
          http://www.amazon.com/reviews/iframe?akid=[AWS Access Key ID]&as
          in=0316067938&exp=2013-08-01T17%3A54%3A07Z&linkCode=xm2&summary=0&tag=ws&trun
          cate=256&v=2&sig=[Signature]
        </IFrameURL>
      </CustomerReviews>
    </Item>
  </Items>
</ItemLookupResponse>
```

After you have the `IFrameURL`, you can embed it on a web page, as shown in the following example:

```
<iframe src="http://www.amazon.com/reviews/iframe?akid=[AWS Access Key ID]&as in=0316067938&exp=2011-08-01T17%3A54%3A07Z&linkCode=xm2&summary=0&tag=ws&trun cate=256&v=2&sig=[Signature]" />
```

Suggest Similar Items to Buy

Topics

- [SimilarityLookup \(p. 133\)](#)
- [Find Similar Item IDs \(p. 135\)](#)
- [Return Items Similar to Cart Items \(p. 135\)](#)
- [Similar Versions of the Same Item \(p. 136\)](#)
- [Related Items \(p. 137\)](#)

Similarity is based on items customers bought. (Customers who bought X also bought Y.) This algorithm is different from the one used for items viewed. Basing the algorithm on purchases rather than viewing history returns items that will likely interest customers.

You can use the following operation and response groups to return similar items:

- `SimilarityLookup` operation
- `Similarities` response group
- `CartSimilarities` response group

The `Similarities` response group returns item IDs that are similar to items in a response. To return additional information for these items, specify the similar item IDs in `ItemLookup` requests.

SimilarityLookup

Contents

- [Specify Multiple Items \(p. 134\)](#)
- [Filter for Similar Items Sold by Amazon \(p. 135\)](#)

When you specify an item ID in a `SimilarityLookup` request, the response returns similar items. You can use the `Similarities` response group to return similar results. However, the `SimilarityLookup` operation has the following advantages:

- The operation provides a number of request parameters, such as `Condition`, to filter the response.
- You can specify many response groups to customize the information in the response.
- You can specify more than one item in a request and find items that are similar to all of them, or similar to each item in the request.

The following request returns items similar to the specified ASIN.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=SimilarityLookup&  
ItemId=[ASIN]
```

```
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

The following response snippet shows one found item, Mark VII Men's Short Sleeve Knit Golf Shirt, is similar to the ASIN specified in the request.

```
<Item>  
<ASIN>B0009VX8VI</ASIN>  
<ItemAttributes>  
<ProductGroup>Apparel</ProductGroup>  
<Title>Mark VII Men's Short Sleeve Knit Golf Shirt</Title>  
</ItemAttributes>  
</Item>
```

Specify Multiple Items

The following request is a `SimilarityLookup` operation that specifies multiple items in a request.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=SimilarityLookup&  
ItemId=ASIN1,ASIN2,ASIN3  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

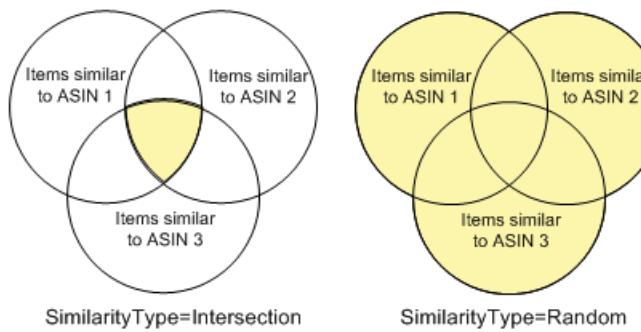
To specify how similar items are returned, set the value of the `SimilarityType` parameter. The following values are accepted:

Intersection

Items returned are similar to all items specified in the request. This is the default value.

Random

The items returned are similar to at least one of the items specified in the request.



The second value is called Random because the items returned are randomly chosen from all similar items found. The operation can return up to ten similar items. It's possible there are more than ten similar items, and identical `SimilarityLookup` requests can return different similar items.

It's also possible similar items are not returned. If similar items can't be found, a response error is returned.

```
<Error>
<Code>AWS.ECommerceService.NoSimilarities</Code>
<Message>There are no similar items for this ASIN: B000B776KY.
</Message>
</Error>
```

An empty result is more likely when the `SimilarityType` value is `Intersection`.

Filter for Similar Items Sold by Amazon

The `SimilarityLookup` operation can restrict items returned with the `MerchantId` parameter. Set the value to `Amazon` to specify similar items sold by Amazon. This parameter is optional in `SimilarityLookup` requests.

Find Similar Item IDs

The `Similarities` response group returns the title and ID of items that are similar to items returned in a response.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate Tag]&
Operation=ItemLookup&
ItemId= B000184IY0&
ResponseGroup=Similarities
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The following similar item was returned.

```
<SimilarProduct>
<ASIN>B00004GJVO</ASIN>
<Title>Minor Move</Title>
</SimilarProduct>
```

You can use the information in this response to display the titles of similar items. To display more information about similar items, use the `ASIN` returned in a new `ItemLookup` request and specify an appropriate response group, such as `ItemAttributes`.

Return Items Similar to Cart Items

The `CartSimilarities` response group returns items that are similar to items in the cart. Items are returned based on the following:

Similarity

Items are similar to the item purchased.

Items viewed

Customers who viewed the item in the cart also viewed the items returned by the response group.

Similar items

Similar items in other product categories.

Each similarity type has its own element tag: `<SimilarProduct>`, `<SimilarViewedProduct>`, and `<OtherCategoriesSimilarProduct>`.

The following request searches for items similar to the item in the cart, ASIN B000062TU1.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=CartCreate&  
Item.1.ASIN=B000062TU1&  
Item.1.Quantity=2&  
ResponseGroup=CartSimilarities  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

This request returns the following response snippet.

```
<SimilarProducts>  
  <SimilarProduct>  
    <ASIN>B00008DDXC</ASIN>  
    <Title>Harry Potter and the Chamber of Secrets (Widescreen Edition) (Harry  
Potter 2)</Title>  
  </SimilarProduct>  
  </SimilarProducts>  
  <SimilarViewedProducts>  
    <SimilarViewedProduct>  
      <ASIN>B000E6UZZK</ASIN>  
      <Title>Harry Potter Years 1-4 (Harry Potter and the Sorcerer's Stone  
/Chamber of Secrets / Prisoner of Azkaban / Goblet of Fire) (Widescreen Edition)  
      </Title>  
    </SimilarViewedProduct>  
  </SimilarViewedProducts>  
  <OtherCategoriesSimilarProducts>  
    <OtherCategoriesSimilarProduct>  
      <ASIN>0590353403</ASIN>  
      <Title>Harry Potter and the Sorcerer's Stone (Book 1)</Title>  
    </OtherCategoriesSimilarProduct>  
  </OtherCategoriesSimilarProducts>
```

This response shows the *Harry Potter and the Sorcerer's Stone* DVD in the shopping cart is similar to another Harry Potter DVD, *Harry Potter and the Chamber of Secrets*. Customers who viewed the DVD in the shopping cart also viewed the DVD, *Harry Potter Years*.

A related item to the DVD in the cart, but in a different product category, is the book version of the DVD: *Harry Potter and the Sorcerer's Stone*.

For more information, see [CartSimilarities \(p. 244\)](#).

Similar Versions of the Same Item

Some items are available in different media formats, such as hardback, paperback, audio CD, and DVD. To return all media formats for an item, use the `AlternateVersions` response group.

Note

The `AlternateVersions` response group works with items in the KindleStore, Music, MP3Downloads, Books, or ForeignBooks product categories only. For `ItemSearch`, the search index must be Books or, in non-US locales, ForeignBooks. For `ItemLookup`, the specified item must be a music title, a book, or a foreign book.

The following request searches for books and similar media formats that have the keyword, potter.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
Operation=ItemSearch&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
SearchIndex=Books&  
Keywords=potter&  
ResponseGroup=AlternateVersions&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

This response shows the book, *Harry Potter and the Half-Blood Prince*, is available on audio cassette and CD.

```
<AlternateVersions>  
  <AlternateVersion>  
    <ASIN>030728364X</ASIN>  
    <Title>Harry Potter and the Half-Blood Prince (Book 6)</Title>  
    <Binding>Audio Cassette</Binding>  
  </AlternateVersion>  
  <AlternateVersion>  
    <ASIN>0307283658</ASIN>  
    <Title>Harry Potter and the Half-Blood Prince (Book 6)</Title>  
    <Binding>Audio CD</Binding>  
  </AlternateVersion>  
</AlternateVersions>
```

Related Items

Contents

- [Relationship Types \(p. 137\)](#)
- [Hierarchy of Relationship Types \(p. 138\)](#)
- [Recommend Items to Others \(p. 140\)](#)

You can use the `RelatedItems` response group in an `ItemLookup` request to return a list of related items. You must use the `RelationshipType` parameter in the request to specify the relationship between the related items. If you have more than ten related items, use the `RelatedItemPage` parameter to return the next set of ten items. For example, a value of 2 returns the second set of ten related items.

Relationship Types

The following table shows all valid values for `RelationshipType`.

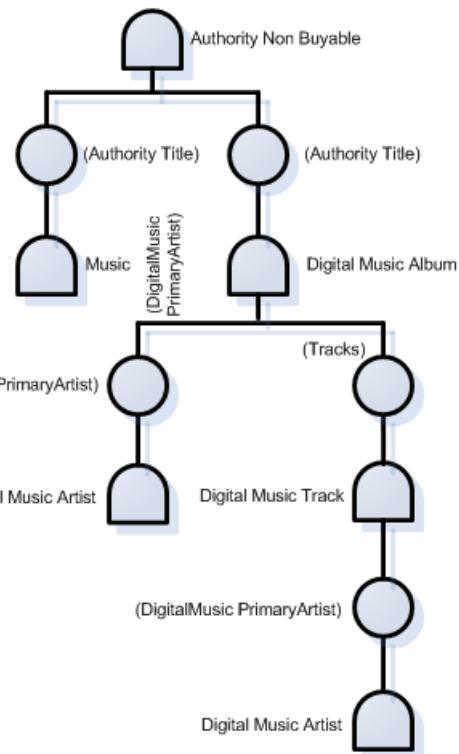
RelationshipType Value	Description
AuthorityTitle	Links a non-buyable ASIN TitleAuthority parent with its buyable children. A book might have a single TitleAuthority ASIN that relates to a list of child ASINs for different formats of the same book (hardback, paperback, audio book, Kindle). MP3 albums have the same AuthorityTitle parent as its physical CD counterpart.
DigitalMusicArranger	Non-buyable child of both MP3 albums and tracks.
DigitalMusicComposer	Non-buyable child of both MP3 albums and tracks.
DigitalMusicConductor	Non-buyable child of both MP3 albums and tracks.
DigitalMusicEnsemble	Non-buyable child of both MP3 albums and tracks.
DigitalMusicLyricist	Non-buyable child of both MP3 albums and tracks.
DigitalMusicPerformer	Non-buyable child of both MP3 albums and tracks.
DigitalMusicPrimaryArtist	Non-buyable child of both MP3 albums and tracks. This is the relationship that shows all MP3 downloads for a single artist on Amazon.com.
DigitalMusicProducer	Non-buyable child of both MP3 albums and tracks.
DigitalMusicRemixer	Non-buyable child of both MP3 albums and tracks.
DigitalMusicSongWriter	Non-buyable child of both MP3 albums and tracks.
Episode	Relates an Unbox Season (parent) to Episodes (children) from that season. This value can be used interchangeably with Tracks.
NewerVersion	Returns the latest version of an item.
Season	Relates an Unbox Series (parent) to its Seasons (children).
Tracks	Relates an MP3 Album (parent) to its Tracks (children). This value can be used interchangeably with Episode.

Hierarchy of Relationship Types

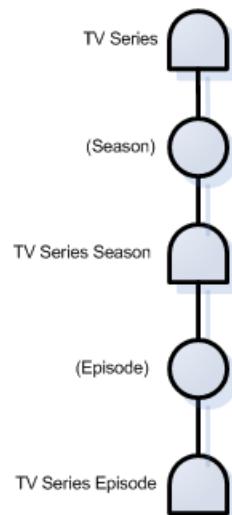
The RelationshipTypes values are arranged in a hierarchy. The following graphics show the hierarchies for MP3Downloads, UnboxVideo, and KindleStore items. The circles represent relationship types. The bell-shaped figures represent items.

This example shows the MP3Downloads hierarchy.

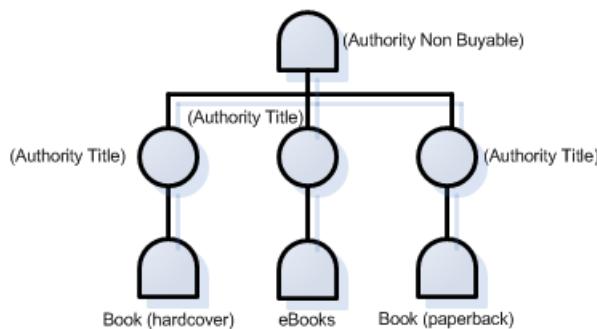
Product Advertising API Developer Guide
Suggest Similar Items to Buy



This example shows the UnboxVideo hierarchy.



This example shows the KindleStore hierarchy.



Authority Title is similar to a variation parent. It is an organizational tool (a container), not an item you can buy.

Recommend Items to Others

The Small response group and its parent response groups return a **Tell Your Friend** link. The link displays items you recently viewed and items Amazon recommends.

```
http://www.amazon.com/gp/pdp/taf/B000IHL51S%3FSubscriptionId%3DAKIAIOSFODNN7EX
AMPLE%26tag%3Dws%26linkCode%3Dxm2%26camp%3D2025%26creative%3D386001%26creativeAS
IN%3DB000IHL51S
```

Top Sellers

Topics

- [Find Top Sellers in a Browse Node \(p. 141\)](#)
- [Get Top Sellers from the Shopping Cart \(p. 142\)](#)

You can use the `BrowseNodeLookup` operation with the `TopSellers` response group to return the top sellers in a browse node.

For example, the following request returns the top sellers in browse node 20.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate Tag]&
Operation=BrowseNodeLookup&
BrowseNodeId=20&
ResponseGroup=TopSellers
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The following response snippet has the name and ASIN of the two top sellers in browse node 20.

```
<TopSeller>
<ASIN>0446578622</ASIN>
<Title>The Notebook Girls</Title>
</TopSeller>
<TopSeller>
<ASIN>1400062586</ASIN>
```

```
<Title>You're Wearing That? : Understanding Mothers and Daughters in Conversation</Title>
</TopSeller>
```

When you have the browse node ID of an item, you can find the top sellers in that browse node. Use the `BrowseNodes` response group with the operation `ItemLookup`, `ItemSearch`, or `SimilarityLookup`.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate Tag]&
Operation=ItemLookup&
ItemId=B00008OE6I&
ResponseGroup=Browsenodes
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The following snippet displays the top seller ASIN B000002ADT, in browse node 63926.

```
<Item>
<ASIN>B000002ADT</ASIN>n
<BrowseNodes>
<BrowseNode>
<BrowseNodeId>63926</BrowseNodeId>
<Name>General</Name>
```

To find top-level browse nodes for your locale, see [Locale Reference \(p. 328\)](#).

Find Top Sellers in a Browse Node

You can return the top sellers in a root browse node. For example, if a customer is buying a comedy movie, you might assume the customer is interested in other comedy items. In this case, use the `BrowseNodeLookup` operation. Its default response group is `BrowseNodes`.

The following example request searches for DVD comedies with browse node 163357.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate Tag]&
Operation=BrowseNodeLookup&
BrowseNodeId=163357
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The response returns the browse node IDs for the parent and child browse nodes of 163357.

```
<BrowseNode>
<BrowseNodeId>163357</BrowseNodeId>
<Name>Comedy</Name>
<Children>
<BrowseNode>
<BrowseNodeId>599826</BrowseNodeId>
<Name>Boxed Sets</Name>
```

```
</BrowseNode>
...
<Item>
  <Ancestors>
    <BrowseNode>
      <BrowseNodeId>549726</BrowseNodeId>
      <Name>Performing Arts</Name>
    <Ancestors>
      <BrowseNode>
        <BrowseNodeId>5</BrowseNodeId>
        <Name>Entertainment</Name>
      <Ancestors>
        <BrowseNode>
          <BrowseNodeId>2000</BrowseNodeId>
          <Name>Subjects</Name>
        <Ancestors>
          <BrowseNode>
            <BrowseNodeId>1000</BrowseNodeId>
            <Name>Books</Name>
```

The response returns one of the browse node's children and the browse node ancestry. As you move down the response, you move up the browse node hierarchy. In this example, the eldest ancestor of browse node ID 163357 is browse node ID 1000, Books.

If you have the browse node ID of the root product category, you can use the `BrowseNodeLookup` operation again with the `TopSellers` response group. This returns the top sellers in the root browse node category.

Note

`BrowseNodeLookup` returns one ancestor of a browse node, even if a node has multiple ancestors. The ancestor returned is not predictable. Any root browse node ID returned is relevant to the browse node ID in the request.

Get Top Sellers from the Shopping Cart

The `CartTopSellers` response group returns the ASINs and titles of the top five best sellers in the root category of the item specified in the cart operation. For example, when you add a television to a cart, the five top sellers in the root category, electronics, such as computers or cameras, are returned.

The `CartTopSellers` response group can be used with most cart operations. The following request searches for top sellers that share the same root browse node as the item in the cart, ASIN B000062TU1.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate Tag]&
Operation=CartCreate&
Item.1.ASIN=B000062TU1&
Item.1.Quantity=2&
ResponseGroup=CartTopSellers
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The following response shows two top sellers.

```
<TopSellers>
  <TopSeller>
    <ASIN>B00005JOFQ</ASIN>
    <Title>Brokeback Mountain (Widescreen Edition)</Title>
  </TopSeller>
  <TopSeller>
    <ASIN>B000E6EK3S</ASIN>
    <Title>Harry Potter and the Goblet of Fire (Widescreen Two-Disc Deluxe Edition) (Harry Potter 4)</Title>
  </TopSeller>
</TopSellers>
```

New Releases

You can return new releases with the `NewReleases` and `CartNewReleases` response groups.

The `NewReleases` response group returns the ASIN and title of new releases in a specified browse node. This response group works only with `BrowseNodeLookup` requests, as shown in the following request.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate Tag]&
Operation=BrowseNodeLookup&
BrowseNodeId=4229&
ResponseGroup=NewReleases
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The response returns new releases associated with browse node 4229.

```
<NewReleases>
  <NewRelease>
    <ASIN>0446578622</ASIN>
    <Title>The Notebook Girls</Title>
  </NewRelease>
</NewReleases>
```

To find the browse node ID of an item, use the `BrowseNodes` response group.

Return New Releases from Shopping Cart Items

To encourage add-on sales, display new releases that are similar to items in a customer's shopping cart. You can use the `CartNewReleases` response group, which works with most cart operations.

The `CartNewReleases` response group returns the ASINs and titles of the top five new releases in the root category of the item specified in the cart operation. For example, if you add a television to a cart, the top five new releases in the root category, electronics, are returned.

In the following request, the item B000062TU1 is added twice to a new shopping cart.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
```

```
AssociateTag=[Associate Tag]&
Operation=CartCreate&
Item.1.ASIN=B000062TU1&
Item.1.Quantity=2&
ResponseGroup=CartNewReleases
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The response returns a new release from the root product category.

```
<NewReleases>
  <NewRelease>
    <ASIN>B00005JOFQ</ASIN>
    <Title>Brokeback Mountain (Widescreen Edition)</Title>
  </NewRelease>
</NewReleases>
```

Sort by Popularity, Price, or Condition

How you display items is important to customers. Customers want information, such as price, popularity, and which items are collectibles. You can organize items with the `Sort` parameter in `ItemSearch` requests.

The valid values for the `Sort` parameter in `ItemSearch` requests vary by locale and search index. For more information, see [Locale Reference \(p. 328\)](#).

The most common `Sort` parameter values are described in the following table.

Sort Parameter	Description
pricerank, price, +price	Order items by price, from least to most expensive. The three versions of this value are the same, but are valid in different search indices.
inversepricerank, -price	Order items by price from most to least expensive. The two versions of this value are the same, but are valid in different search indices.
salesrank	Order items from best to worst selling.
relevancerank	Order items by keywords. Rank is determined by the keywords in the product description, if there are multiple keywords, how closely they occur in descriptions, and how often customers purchased items they found using the keyword. Keyword placement is also important. For example, the rank is higher when keywords are in titles.
reviewrank	Order items by customer reviews, from highest to lowest ranked.

For example, the following request returns a list of toy rockets sorted from least expensive to most expensive.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate Tag]&
Operation=ItemSearch&
Keywords=Rocket&
```

```
SearchIndex=Toys&
Sort=price&
ResponseGroup=Offers&
ItemPage=10
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The following is a snippet of the response.

```
<Item>
  <ASIN>B000BOWQWA</ASIN>
  <OfferSummary>
    <LowestNewPrice>
      <Amount>210</Amount>
      <CurrencyCode>USD</CurrencyCode>
      <FormattedPrice>$2.10</FormattedPrice>
    </LowestNewPrice>
    <TotalNew>3</TotalNew>
    <TotalUsed>0</TotalUsed>
    <TotalCollectible>0</TotalCollectible>
    <TotalRefurbished>0</TotalRefurbished>
  </OfferSummary>
</Item>
<Item>
  <ASIN>B0006N6MGW</ASIN>
  <OfferSummary>
    <LowestNewPrice>
      <Amount>211</Amount>
      <CurrencyCode>USD</CurrencyCode>
      <FormattedPrice>$2.11</FormattedPrice>
    </LowestNewPrice>
    <TotalNew>2</TotalNew>
    <TotalUsed>0</TotalUsed>
    <TotalCollectible>0</TotalCollectible>
    <TotalRefurbished>0</TotalRefurbished>
  </OfferSummary>
</Item>
```

This response lists two toy rockets, which are sorted by price. The same ASIN can be offered for sale by multiple merchants and sellers. The Offers response group includes a summary of the lowest price per condition. For example, the first item, B000BOWQWA, is sold as new by three sellers. The lowest price for the item in new condition is \$2.10. No one is selling the item in used, collectible, or refurbished condition.

All other Sort parameter values work in a similar way.

Returning Price and Availability Information

Topics

- [Returning Prices \(p. 146\)](#)
- [Determining Availability \(p. 151\)](#)

Previous chapters have shown how to find items and how to suggest additional items to customers. Returning price and availability information is important for the customer.

Returning Prices

Topics

- Minimum Advertised Price (p. 146)
- Offer Summaries (p. 146)
- Returning All Offers (p. 147)
- Returning More Information About the Offer (p. 147)
- Providing Price Ranges for Product Categories (p. 148)
- Items That Do Not Have Offers (p. 149)
- Determining Parent Items (p. 149)
- Returning Offers From Item Variations (p. 149)
- Variation Dimensions (p. 150)

Items for sale on Amazon can be sold by more than one seller or merchant. For example, the same camera might be sold by three different merchants or sellers. The following image shows an example where two merchants are selling the same book.

The screenshot displays two separate Amazon product listing snippets. The top snippet is for a new book priced at \$11.67. It includes information about shipping, fulfillment by Amazon, and delivery options. The bottom snippet is for the same book, also new, but sold by a different seller (pbshopus) for \$12.19. It includes the seller's rating and shipping information.

Offer Summary	Price	Condition	Fulfillment	Shipping	Seller Rating
Amazon	\$11.67	New	Fulfilled by Amazon	In Stock	88% positive ratings
pbshopus	\$12.19	New	Ships from NJ, United States	In Stock	88% positive ratings

Notice that the merchants are selling the same book for different prices. In other examples, the same item might be offered in multiple conditions, such as New and Used.

Any item being sold is associated with an offer. An offer is a combination of price, condition, and vendor. For example, one offer might be Amazon selling the new book for \$11.67. Therefore, to find an item's price, you return the offers made by the vendors selling the item.

Offer information is made available by the OfferSummary, Offers, and OfferFull response groups, which can be part of `ItemLookup`, `ItemSearch`, and `SimilarityLookup` requests.

Minimum Advertised Price

Some manufacturers have a minimum advertised price (MAP) that can be displayed on Amazon.com. When the Amazon price is lower than the MAP, the manufacturer does not allow the price to be shown until the customer takes further action, such as placing the item in their shopping cart, or in some cases, proceeding to the final checkout stage.

When performing an `ItemSearch` or `ItemLookup` operation, the string "Too Low to Display" is returned instead of the actual price. Customers need to go to Amazon to see the price on the retail website, but won't be required to purchase the product.

Offer Summaries

The OfferSummary response group returns summary information about offers, including:

- Total number of offers per condition
- Lowest price per condition

For example, the following response snippet shows the lowest prices for an item in New and Used condition.

```
<OfferSummary>
<LowestNewPrice>
  <Amount>801</Amount>
  <CurrencyCode>USD</CurrencyCode>
  <FormattedPrice>$8.01</FormattedPrice>
</LowestNewPrice>
<LowestUsedPrice>
  <Amount>799</Amount>
  <CurrencyCode>USD</CurrencyCode>
  <FormattedPrice>$7.99</FormattedPrice>
</LowestUsedPrice>
<TotalNew>45</TotalNew>
<TotalUsed>20</TotalUsed>
<TotalCollectible>0</TotalCollectible>
<TotalRefurbished>0</TotalRefurbished>
</OfferSummary>
```

Returning All Offers

Amazon returns offers for new and used items. The Small response group, and all of its parent response groups, return a link that displays all new and used offers for the item in the response. The link is similar to the following.

```
http://www.amazon.com/gp/offer-listing/B000IHL51S%3FSubscriptionId%3DAKIAIOSFODNN7EXAMPLE
%26Code%3Dxm%26camp%3D2025%26creative%3D386001%26creativeASIN%3DB000IHL51S
```

Returning More Information About the Offer

Perhaps you want to display more than just the cheapest offer of an item in each condition. In this case, you would use the Offers response group.

The following response snippet shows the data included for a single offer.

```
<Offer>
  <OfferAttributes>
    <Condition>Used</Condition>
  </OfferAttributes>
  <OfferListing>
    <OfferListingId>f0ctUVF4LATNxevdXAj5g0j3UbEE%2B1gm%2Fy3ZNjJ%2BrxgGUqPn%2FNy%2BqBcVFoVLQWAugwlj05Zt9zqm865o%2BTk3zGIGDF5oWogFW7frgBq77QgTs%2Baxm%2B2gLQ%3D%3D
    </OfferListingId>
    <Price>
      <Amount>21995</Amount>
      <CurrencyCode>USD</CurrencyCode>
      <FormattedPrice>$219.95</FormattedPrice>
    </Price>
  </OfferListing>
</Offer>
```

```
</Price>
<Availability>Usually ships in 1-2 business days
</Availability>
<IsEligibleForSuperSaverShipping>0
</IsEligibleForSuperSaverShipping>
</OfferListing>
</Offer>
```

If you want more information about the vendor than is included in the Offers response group, use the OfferFull response group. That response group adds to the Offers response the:

- Merchant's name
- Merchant's nickname

Providing Price Ranges for Product Categories

You can use the SearchBins response group to return the price ranges of items for sale in a specified product category. The following response to an [ItemSearch \(p. 185\)](#) request shows how the SearchBins response group can provide price ranges.

```
<SearchBinSets>
  <SearchBinSet NarrowBy="PriceRange">
    <Bin>
      <BinName>$25-$49</BinName>
      <BinItemCount>316</BinItemCount>
      <BinParameter>
        <Name>MinimumPrice</Name>
        <Value>2500</Value>
      </BinParameter>
      <BinParameter>
        <Name>MaximumPrice</Name>
        <Value>4999</Value>
      </BinParameter>
    </Bin>
  </SearchBinSet>
  <Bin>
    <BinName>$0-$24</BinName>
    <BinItemCount>280</BinItemCount>
    <BinParameter>
      <Name>MinimumPrice</Name>
      <Value>0</Value>
    </BinParameter>
    <BinParameter>
      <Name>MaximumPrice</Name>
      <Value>2499</Value>
    </BinParameter>
  </Bin>
</SearchBinSets>
```

This response shows that there are 316 items in this product category that cost between \$25 and \$49 and 280 items that cost between \$0 and \$24.

Knowing what price ranges are available for a product category enables you to use the `MaximumPrice` and `MinimumPrice` parameters in a second `ItemSearch` request to retrieve items only in the specified

price range. Using one of the offer response groups in the same request enables you to display the prices of the items in a specified price range.

Items That Do Not Have Offers

There are two kinds of items returned by Product Advertising API:

- Regular items
- Parent items

A parent item is an abstraction of a collection of items. For example, shirt is a parent item. A shirt that is a specific size and color would be a child of the parent item, which is called a regular item. Each regular item has an item identifier, such as an ASIN. That means that a red, large shirt would have a different ASIN from the same shirt that is blue and size large. The child items are also called variations.

Because a parent item is an abstraction, it cannot be sold so it does not have an `OfferListingId`.

Determining Parent Items

How do you determine if the item returned in a response is a parent item? If you use either the `Variations` or `VariationSummary` response groups in a request, the responses will include a `VariationSummary` tag if the item is a parent item.

Secondly, if you use one of the offer response groups, parent items do not have offers. So, the summary would be as follows.

```
<OfferSummary>
<LowestNewPrice>
<Amount>0</Amount>
<CurrencyCode>USD</CurrencyCode>
<FormattedPrice>$0.00</FormattedPrice>
</LowestNewPrice>
<TotalNew>0</TotalNew> <TotalUsed>0</TotalUsed> <TotalCollectible>0</TotalCollectible>
<TotalRefurbished>0</TotalRefurbished>
</OfferSummary>
```

Returning Offers From Item Variations

The Book, Music, Video, and DVD search indices have very few parent items. The Apparel, Jewelry, and SportingGoods search indices have many parent items and variations.

If a parent item is returned in a response, it will not have an offer. Instead, you must find the offers associated with the variations of the parent item. You do this by adding the `Variations` response group to the request, as follows.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[Access Key ID]&
AssociateTag=[Associate Tag]&
Operation=ItemSearch&
Keywords=Ralph%20Lauren&
SearchIndex=Apparel&
Sort=pricerank&
ResponseGroup=Offers,Variations&
Availability=Available&
```

```
Condition=All
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The Variations and Offers response groups returns the first 10 variations of the parent item and their offers, which is shown in the following response snippet.

```
<Variations>
<TotalVariations>6</TotalVariations>
<TotalVariationPages>1</TotalVariationPages>
<Item>
<ASIN>B000FG8I6W</ASIN>
<ItemAttributes>
<Binding>Apparel</Binding>
<Brand>Polo Ralph Lauren</Brand>
<ClothingSize>10 - 13</ClothingSize>
<Color>Beige Heather</Color>
<Department>mens</Department>
...
</ItemAttributes>
<Offers>
<Offer>
<Merchant>
<Name>Polo.com</Name>
</Merchant>
<OfferAttributes>
<Condition>New</Condition>
</OfferAttributes>
<OfferListing>
<OfferListingId>WjQ49eW5WwGDJ4Ga6u06I156YzTQKzVh0G2ag0k9vZtbj
cojcnUwkkbogxg6dzeDmu9alzXSn04nz0DvdnZT8gqeRSRRx2shyER2SDHWREJth
dUmXvDYO%2Bf0z%2F0dpye</OfferListingId>
<Price>
<Amount>1400</Amount>
<CurrencyCode>USD</CurrencyCode>
<FormattedPrice>$14.00</FormattedPrice>
</Price>
<SalePrice>
<Amount>560</Amount>
<CurrencyCode>USD</CurrencyCode>
<FormattedPrice>$5.60</FormattedPrice>
</SalePrice>
<Availability>Usually ships in 1-2 business days</Availability>
</OfferListing>
</Offer>
</Offers>
```

Variation Dimensions

A variation is a child ASIN. The parent ASIN is an abstraction of the children items. For example, Shirt is a parent ASIN. Parent ASINs cannot be sold. A child ASIN of it would be a blue shirt, size 16, sold by MyApparelStore. This child ASIN is one of potentially many variations. The ways in which variations differ are called dimensions.

In the preceding example, size and color are the dimensions. Parent ASINs therefore return two related elements:

- VariationDimensions
- VariationDimension

The following response snippet shows these elements.

```
<VariationDimensions>
  <VariationDimension>ClothingSize</VariationDimension>
  <VariationDimension>Color</VariationDimension>
</VariationDimensions>
```

The values returned by these elements are the dimensions listed in the child ASIN's response, as shown.

```
<Item>
  ...
  <ItemAttributes>
    ...
    <ClothingSize>Large</ClothingSize>
    <Color>Black</Color>
    ...
  </ItemAttributes>
  ...
</Item>
```

Determining Availability

Topics

- [Availability Values \(p. 152\)](#)
- [Checking for an Offer Listing ID \(p. 153\)](#)
- [Using the ItemSearch Availability Parameter \(p. 154\)](#)

Only items that are available can be added to the remote shopping cart. Unavailable items can be added to the SaveForLater area of the cart.

Amazon defines available items as those that are:

- Currently for sale
- Pre-release orders, such as buying a Harry Potter book before it is released
- Special orders
- e-mail me when the items become available

The availability of an item can change without notice. Putting an item in a cart does not reserve it. Items in carts can become unavailable and their prices can change as soon as the item goes into the cart.

Determining if an item is available is determined differently for merchants and sellers.

Availability of Merchant Items

1	Check the Availability element value returned by the Offers or OfferFull response groups.
2	Make sure the item has an offer listing ID.

3	Use the Availability request parameter along with the Condition parameter in an ItemSearch request. Items might be available, for example, in one condition but not another.
---	--

With Merchants, you use one of the Offer response groups to determining item availability.

Availability Values

The Offer Summary Report lists the availability of an item using the Availability element. The value of the element indicates if the item can be purchased and how soon it will be shipped, as shown.

```
<Availability>Usually ships in 24 hours</Availability>
```

This response, which is returned by the Offers and OfferFull response groups, confirms that an item is available to buy.

The value returned by the Availability element may not match the one on the Amazon retail web site's product detail page because typically there is a short and long version of an availability message. Product Advertising API returns the short version. The more verbose availability message is used on the retail web site.

For non-Amazon products in JP, FR, DE, and CA; the value returned by the Availability element is "1-2 business days", or the localized equivalent, regardless of the actual availability.

The following table describes the possible Availability element values.

Message	Description
Usually ships in %X	A dynamic response where %X represents a variable amount of time.
Not yet released	The item is not available for purchase. The item may or may not have a projected release date. If there is a release date, it may show up in the ReleaseDate element of the item attributes.
Not yet published	The item is not available for purchase. The item may or may not have a projected release date. If there is a release date, it may show up in the ReleaseDate element of the item attributes.
This item is not stocked or has been discontinued.	The item is not available for purchase.
Out of Stock	The item is currently not available for purchase, but may be in the future.
Limited Availability	Used for items sold by third-parties if an item is out of stock, but may be available for purchase later.
Out of Print--Limited Availability	Customers can choose to be notified if a copy becomes available.
Special Order	Titles occasionally go out of print or publishers run out of stock. The buyer is notified if the item becomes unavailable.
This item is currently not available by this merchant	The message is sent primarily for Amazon offers that are missing an availability message.

These availability messages apply to Amazon only. Sellers might return a completely different set of availability messages.

Checking for an Offer Listing ID

An offer listing ID is an alphanumeric token that uniquely identifies an item that is sold by any merchant, including Amazon. Whereas an offer is a combination of Condition and Price, an offer listing ID is similar to a price tag, one is associated with each item for sale, as shown in the following figure.



This figure shows that three vendors are selling the same item, a shirt. The vendor, labeled Offer 1, has three shirts in stock and each has an OfferListingId.

If an item is for sale, it has an offer listing ID. This ID is returned by the Offers and OfferFull response groups, as shown in the following response snippet.

```
<OfferListing>
  <OfferListingId>[Offer Listing ID]</OfferListingId>
  <Price>
    <FormattedPrice>Too low to display</FormattedPrice>
  </Price>
  <Availability>Usually ships in 24 hours</Availability>
  <IsEligibleForSuperSaverShipping>0</IsEligibleForSuperSaverShipping>
  <IsEligibleForPrime>1</IsEligibleForPrime>
</OfferListing>
```

Offers contain availability and shipping information.

Using the OfferSummary Response Group

The OfferSummary response group returns, in part, the total number of items available in each condition, for example:

```
<Item>
  <ASIN>B000BWFJQ2</ASIN>
  <OfferSummary>
    <LowestNewPrice>
      <Amount>295</Amount>
      <CurrencyCode>USD</CurrencyCode>
      <FormattedPrice>$2.95</FormattedPrice>
    </LowestNewPrice>
    <TotalNew>1</TotalNew> <TotalUsed>0</TotalUsed> <TotalCollectible>0</TotalCollectible> <TotalRefurbished>0</TotalRefurbished>
  </OfferSummary>
```

This response shows that only one of the specified items, B000BWFJQ2, is available and it is only available in "New" condition.

Using the ItemSearch Availability Parameter

The `Availability` parameter enables you to search only for items that are available. The only valid value for `Availability` is "Available" as shown in the next example.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=ItemSearch&  
Condition=All&  
Availability=Available&  
SearchIndex=Apparel&  
Keywords=Shirt  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Items can become unavailable quickly. Even though this parameter filters out unavailable items, checking that an item has an `OfferListingId` is still recommended.

Working With Remote Shopping Carts

Topics

- [Shopping Cart Concepts \(p. 154\)](#)
- [Remote Shopping Cart Tasks \(p. 158\)](#)
- [Add to Cart Form \(p. 166\)](#)

Instead of physically adding and removing items from a real shopping cart, Product Advertising API operations provide similar functionality, called the Product Advertising API remote shopping cart.

After customers find an item to purchase, they typically click an HTML button or link called, for example, **Add To Cart**. This action adds the item(s) to their existing shopping cart. If this is the first item they have chosen to place in a shopping cart, the shopping cart is created.

In Product Advertising API, the shopping cart is called remote because the cart is hosted by Amazon servers. In that sense, the shopping cart is remote to the seller's or Associate's servers where the customer is shopping.

The opposite of a remote shopping cart is a local shopping cart, which is the shopping cart customers use when shopping on Amazon.com. It is local because Amazon hosts the shopping web pages and the shopping cart. Product Advertising API operations work solely with remote shopping carts.

The following sections describe remote shopping carts and how to work with them.

Shopping Cart Concepts

Topics

- [Cart Identifiers \(p. 155\)](#)
- [Cart Limitations \(p. 155\)](#)

- [Active and SaveForLater Areas \(p. 155\)](#)
- [Items That Cannot Be Added to the Active Cart Area \(p. 155\)](#)
- [Remote Shopping Carts are Hosted by Amazon \(p. 157\)](#)
- [Cart Lifespan \(p. 157\)](#)

The following sections describe remote shopping carts in detail.

Cart Identifiers

When you create a remote shopping cart using `CartCreate`, Product Advertising API returns the new cart's ID (`CartId`). The cart ID is also an alphanumeric token that is used to identify a cart. The cart ID and HMAC values must be used in all Product Advertising API requests related to the cart (except `CartCreate`). HMAC (Hash-based Message Authentication Code) is a specific construction for calculating a message authentication code (MAC) involving a cryptographic hash function in combination with a secret key. You can use it to verify both the data integrity and the authenticity of a message at the same time. AWS calculates the HMAC using a standard, cryptographic hash algorithm, such as SHA-256.

Cart Limitations

A cart can contain up to 50 entries. Each entry can represent one or more (up to 999) of the same items. That means that a Product Advertising API shopping cart can contain a maximum of almost 50,000 items.

Many products have a limit on the quantity that can be added to a cart. Sometimes this is inherent to the product, for example, one-of-a-kind jewelry items. Sometimes it is a limit set by the vendor, for example, they might set a limit on big ticket items, such as a motorcycle, of one per customer. If you add an item to the cart with a quantity that exceeds the maximum allowed, `CartAdd` automatically resets the quantity to the maximum value. No error is generated. Therefore, you must verify the quantity of items in the cart with the number ordered to determine if the quantity has been capped. If so, you must notify the customer.

Active and SaveForLater Areas

Shopping carts have two distinct areas:

- **Active**—Contains the items that are ready to be purchased.
- **SaveForLaterItem**—Contains items that a customer has chosen to buy but are currently unavailable, or items that a customer has designated they want to save and buy later.

Available items placed in a shopping cart are automatically added to the Active area unless the customer specifies otherwise. Unavailable items or items in the shopping cart that become unavailable are automatically put into the SaveForLater area. When those items become available Amazon automatically moves the items to the Active area of the shopping cart.

The SaveForLater area can also be used by customers as a holding place for items they are interested in but not ready to purchase. The prices and availability for items in this area are regularly updated by Amazon.

Items That Cannot Be Added to the Active Cart Area

Contents

- [Out of Stock Items \(p. 156\)](#)
- [Items With Limited Quantities \(p. 156\)](#)
- [Digital Items \(p. 156\)](#)
- [Variation Parents \(p. 156\)](#)

- [Collection Parents \(p. 157\)](#)

Product Advertising API operations sometimes return items that cannot be added to the Active cart area. The following sections explain those cases:

- Out of stock items
- Items with limited quantities
- Digital items
- Variation parent items
- Collection parent items

Out of Stock Items

Most items for sale are available immediately. There are times, however, when that is not true. Items can go out of stock or very popular items, such as a new Harry Potter book, are pre-sold, that is, the book is sold before the book is even published.

When a customer adds an item to their cart that is not available, it is added to the SavedForLater area. Also, if an item in the cart, for some reason, becomes unavailable, Amazon automatically moves the item in the cart to the SavedForLater items section.

It is also possible for a customer to add an item directly to the SavedForLater items area in their cart so that they can easily purchase the item at a later date.

When items become available, you can use the `CartModify` operation with the `Action` parameter to move items from the SaveForLater section of the cart to the Active section, which is fully described in [Moving Adding Items as Saved For Later \(p. 163\)](#). Or, if Amazon automatically moved an item in the Active area to the SaveForLater area because the item went out of stock, Amazon will move it back into the Active area automatically when it becomes available.

Items With Limited Quantities

Many products have a limit on the quantity that can be added to a cart. Sometimes this is inherent to the product, for example, one-of-a-kind jewelry items. Sometimes it is a limit set by the vendor, for example, they might set a limit for big ticket items of one per customer. If you add an item to the cart with a quantity that exceeds the maximum, `CartAdd` or `CartCreate` automatically resets the quantity to the maximum. No error is generated. Therefore, you must verify the quantity of items in the cart with the number ordered to determine if the quantity has been capped. If so, you must notify the customer.

Digital Items

Digital Items, such as Kindle books, Amazon Instant Video, and MP3 files, can be purchased only by using Amazon 1-click. They cannot be placed in the cart. The following values for the `Binding` response element (`ItemAttributes`) are digital media formats:

- MP3 Music
- Amazon Instant Video
- Kindle Edition

Variation Parents

Variation parent items cannot be added to a cart. For example, you cannot add Apparel to a cart because it is not clear what apparel you really want to buy. You could add one of its children, however, such as a red shirt, size large. If you try to add a parent item, you get an error message similar to the following.

The item you specified, **[ASIN]**, is not eligible to be added to the cart. Check the item's availability to make sure it is available.

Variation parents do not have corresponding OfferListingId's.

Collection Parents

Collection parent items are abstractions of the items in the collection. The collection parent cannot be added to a cart. The Collection parent is used as a name holder for the collection. Collection parents do not have corresponding OfferListingId's.

Remote Shopping Carts are Hosted by Amazon

Although a Product Advertising API application or web site runs on your servers, Product Advertising API remote shopping carts are hosted by Amazon servers. You use Product Advertising API operations to create and modify the cart and its contents. Keeping a local copy of a shopping cart is not recommended because the price and availability of items change often and Amazon automatically updates items in shopping carts. Also, Amazon automatically updates the status of lists, such as Wishlists, when a customer buys an item on a list. Keeping a local copy of a shopping cart runs the risk of cart items getting out of sync with their real price, availability, and status.

Because Product Advertising API shopping carts are hosted by Amazon instead of a local host, the shopping carts are called "remote." The cart used by a customer shopping on the Amazon retail web site, www.amazon.com in the US locale, is considered the "local" shopping cart. Amazon maintains for each customer ID only one local shopping cart. Developers can maintain more than one remote shopping cart for a customer but maintaining one cart per customer ID is recommended.

Cart Lifespan

Contents

- [Cart Creation \(p. 157\)](#)
- [Modifying a Cart \(p. 157\)](#)
- [Cart Expiration \(p. 158\)](#)

Shopping carts have a lifespan. A remote shopping cart is created and, in time, it expires. In between, the contents of the cart can be modified and purchased.

Cart Creation

The first time a customer wants to add an item to their shopping cart, a remote shopping cart must be created, which is accomplished using the `CartCreate` operation. It is not possible to create an empty cart. At least, one item must be added.

Modifying a Cart

If a customer has an existing remote shopping cart, it should be used and modified accordingly. Modification can come in many forms:

- Items can be added to and deleted from the cart
- Items can be moved from the Active area of the cart to the SaveForLater area, or the reverse

The `CartId` and `HMAC` are used in the `CartModify`, `CartAdd`, or `CartClear` operations to modify the contents of the cart.

Cart Expiration

It is not possible to delete a remote shopping cart. Instead, it expires automatically after ninety days of disuse if there are items in the cart, or, if the cart is empty, after seven days of disuse. Carts are emptied either by the `CartClear` or `CartModify` operations, or automatically when the customer purchases the items in their cart. The expiration of the cart is reset when the cart is modified. For example, if, on day eighty-nine of disuse, a customer modifies the items in their shopping cart, the life time of the cart is reset to an additional ninety days. In that way, shopping carts can last indefinitely.

Once the items in a cart have been purchased using the `PurchaseURL` value, the cart is not deleted immediately but it should no longer be used. From a customer's point of view, once he or she has purchased the items in their cart and there are no items remaining in the `SaveForLater` section, the old cart is gone. The next time they want to purchase an item, they receive a new cart.

Remote Shopping Cart Tasks

Topics

- [Creating a Remote Shopping Cart \(p. 158\)](#)
- [Retrieving the Contents of a Cart \(p. 165\)](#)

Product Advertising API operations give you complete control of the items in a remote shopping cart. The tasks you can implement using Product Advertising API cart operations are described in the following sections.

Creating a Remote Shopping Cart

A remote shopping cart must be created the first time a customer decides to add an item to a shopping cart. Once the cart is created, it is reused and modified appropriately until it expires. Only one local cart can be created and only one cart per customer per vendor should be created.

Typically, an HTML button labeled, for example, Add to Cart, implements a `CartCreate` request, which includes:

- At least one item to add to the cart.
You cannot create an empty cart.
- An `AssociateTag`.
Including the `AssociateTag` value gives the Associate credit for the customer's purchase.

Example Creating a Cart

The following request creates a cart that will be merged with the customer's cart when the customer uses `PurchaseURL`.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=CartCreate&  
Item.1.OfferListingId=B000062TU1&  
Item.1.Quantity=2  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

In this example, two of the same item (Quantity=2) whose ASIN is B000062TU1 are added to the newly created (Operation=CartCreate) shopping cart. The Associate specified will receive a referral fee if the customer purchases the item.

In the lifespan of a shopping cart, CartCreate can be used only once. If you call CartCreate a second time, you create a new shopping cart. We recommend that you only create one cart per customer.

Specifying the Items in the CartCreate Request

In one CartCreate request, you can add up to ten different items to the cart. The quantity of each item can be between 1 and 50. That means in one CartCreate request, you could add up to five-hundred items (10 items * 50 quantity).

Items are specified in parameter pairs, as follows:

```
Item.N.ItemIdType=[Item identifier]  
Item.N.Quantity=[Number of Item.N items]
```

N is a positive integer. The N value associates the ItemId with the quantity of those items that should be added to the cart, for example:

```
Item.1.OfferListingId=[An OfferListingId]&  
Item.1.Quantity=3&  
Item.2.OfferListingId=[An OfferListingId]&  
Item.2.Quantity=5&
```

This request adds three number 1 items and five number 2 items to the remote shopping cart. You could interchange lines without affecting the end result, for example:

```
Item.1.OfferListingId=[An OfferListingId]&  
Item.2.OfferListingId=[An OfferListingId]&  
Item.1.Quantity=3&  
Item.2.Quantity=5&
```

The values for N do not have to be listed in ascending or descending order, nor do the values of N need to be sequential. You could, for example, use the values, 1, 4, 7 for N. Also, the value of N is not retained between requests. For that reason, it is permissible to use Item.1 in a CartCreate request and then in a CartAdd request, which adds an item to the cart. In these requests, Item.1 can refer to two completely different items.

The value of adding multiple items to the cart in one request is performance. It takes less time to add ten items in one request than it does to add ten items to the cart in ten separate requests.

Using OfferListingId

You can specify items to add to a cart in the following ways.

Identifiers	Description
ASIN	An alphanumeric token that uniquely identifies an item sold by Amazon. Amazon assigns ASINs to items. Using OfferListingId is the preferred method of identifying items to add to the cart.

Identifiers	Description
OfferListingId	An alphanumeric token that uniquely identifies an item sold by a seller or merchant. Amazon assigns OfferListingIds to items. You can find an OfferListingId using the ItemLookup and ItemSearch operations with Offers and OfferFull response groups. Using OfferListingId is the preferred method of identifying items to add to the cart.

The `CartCreate` example uses OfferListingId to identify the items to add to the cart. An OfferListingId is returned by the Offers and OfferFull response groups. An alternative is using an item's ASIN. An ASIN is assigned to every item offered by Amazon. An OfferListingId is assigned to every item that is available for sale.

The advantage of using OfferListingId in the `CartCreate` request is that items that cannot be purchased, such as Variation parent items, do not have OfferListingId's. By using the OfferListingId, you are assured that an item can be purchased.

Offers and Identifiers

ASINs identify items, such as a Harry Potter book. ASINs do not, however, identify sales information related to those items.



As you can see in the preceding figure, an ASIN identifies an item but not an instance of it that a customer can purchase. Every ASIN can be sold by many merchants. Each merchant lists the condition and price of the item. This combination of data represents an *OfferListingId*. Taken together, all *OfferListingIds* are referred to as offers. If an ASIN does not have an *OfferListingId*, the item cannot be purchased.

For this reason, it is recommended that you add items to a shopping cart by their *OfferListingId* rather than by their ASIN.

Using Values Returned by `CartCreate` in Other Cart Operations

The following XML is a snippet of the response to the `CartCreate` request.

```
<CartId>002 2197248 2529608</CartId>
<HMAC>/WrekkZAPx782xttLFbZqviNUOA=</HMAC>
<URLEncodedHMAC>%2FWrekkZAPx782xttLFbZqviNUOA%3D</URLEncodedHMAC>
<PurchaseURL>https://www.amazon.com/gp/cart/aws_merge.html?cart_id=002 2197248
2529608%26associate_id=ws%26hmac=/WrekkZAPx782xttLFbZqviNUOA=%26AWSAccessKey
Id=[AWS Access Key ID]</PurchaseURL>
<SubTotal>
```

```

<Amount>2998</Amount>
<CurrencyCode>USD</CurrencyCode>
<FormattedPrice>$29.98</FormattedPrice>
</SubTotal>
<CartItems>
  <SubTotal>
    <Amount>2998</Amount>
    <CurrencyCode>USD</CurrencyCode>
    <FormattedPrice>$29.98</FormattedPrice>
  </SubTotal>
  <CartItem>
    <CartItemId>UV3W10T4V7PCZ</CartItemId>
    <ASIN>B000062TU1</ASIN>
    <Quantity>2</Quantity>
    <Title>Harry Potter and the Sorcerer's Stone (Full Screen Edition) (Harry
Potter 1)</Title>
    <ProductGroup>DVD</ProductGroup>
    <Price>
      <Amount>1499</Amount>
      <CurrencyCode>USD</CurrencyCode>
      <FormattedPrice>$14.99</FormattedPrice>
    </Price>
    <ItemTotal>
      <Amount>2998</Amount>
      <CurrencyCode>USD</CurrencyCode>
      <FormattedPrice>$29.98</FormattedPrice>
    </ItemTotal>
  </CartItem>
</CartItems>

```

Many of the item attributes in the response are used in future cart requests. Those item attributes are explained in the following table.

Attribute	Description
CartId	An alphanumeric token that uniquely identifies a remote shopping cart. This value must accompany every cart operation associated with the newly created cart.
HMAC	Hash Message Authentication Code. This is an encrypted alphanumeric token used to authenticate requests. A URL-encoded version of this value must accompany every cart operation associated with the newly created cart. The alternative is to include, instead, the <i>URLEncodedHMAC</i> value.
URLEncodedHMAC	This is the <i>HMAC</i> value with all of the characters converted in to a URL-compliant form. Some <i>HMAC</i> characters, such as plus (+), are incompatible with URLs. This is a problem because the <i>HMAC</i> must be included in every cart operation and thus is part of the URL request. The <i>URLEncodedHMAC</i> value, then, is a convenience function that relieves developers of the need to create their own URL-encoded <i>HMAC</i> value. This value must accompany every cart operation.
	<pre> <HMAC>Cw1g4IbVzOtzFkJR/zBj1GNnZMA</HMAC> <URLEncodedHMAC>Cw1g4IbVzOtzFkJR/zBj1GNnZMA </URLEncodedHMAC> </pre>

Attribute	Description
PurchaseURL	<p>This is the URL that is submitted, like a request, to enter the Order Pipeline and purchase the items in a cart. PurchaseURL includes the Associate's Tag. It is important that this URL is used to make the purchase otherwise the Associate will not get credit for the sale.</p> <pre style="border: 1px solid black; padding: 5px; margin-top: 10px;"><PurchaseURL>https://www.amazon.com/gp/cart/aws-merge.html?cart-id=102-9464231-2184159&associate-id=ws&hmac=Cwlg4IbVzOtZFkJR/zBj1GNnZMA=%26AWSAccessKeyId=1VEXAMPLE9C02</PurchaseURL></pre>
CartItemId	<p>An alphanumeric token that uniquely identifies an item in the cart. Although the items added to a remote shopping cart using <code>CartCreate</code> were identified by an <code>ASIN</code> or <code>OfferListingId</code>, those items lose those associations. Instead, to refer to an item in a cart in future cart operation requests, you must use the item's <code>CartItemId</code> value.</p>

Modifying the Items in a Remote Shopping Cart

Product Advertising API operations give you a lot of flexibility to modify the number of items in a cart. For example, use:

- **CartAdd**—To add new items to a cart.
- **CartClear**—To remove all items from a cart.
- **CartModify**—To increase or decrease the number of items that are already in a cart, and to move items between the Active and the SaveForLater cart areas.

You can use this operation to delete a single item from a cart by setting its quantity to zero.

All of these operations can be used only on an existing remote shopping cart.

Adding Items to a Cart

Often a customer, after creating a shopping cart, wants to keep shopping and add additional items to an existing shopping cart. You can facilitate this activity using the Product Advertising API operations `CartAdd` and `CartModify`.

If the item being added is already in the cart, you have to use the `CartModify` operation to change the quantity of the items already in the cart. You cannot use `CartAdd` to add items that are already in a cart. In the following example, the quantity of the specified item is changed to 10.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate ID]&
Operation=CartModify &
CartId=102-5929035-5792105&
HMAC=[HMAC]&
Item.1.CartItemId=[Cart Item ID]& Item.1.Quantity=10
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

Notice that the item in the cart is referred to by its `CartItemId`. The `Quantity` value, 10, is the total number of those items that should be in the cart.

If the item being added to the cart is not already in the cart, you must use the `CartAdd` operation. `CartAdd` cannot increase the quantity of items that are already in the cart. If you try to do that, you get an error.

The following is an example of a request using `CartAdd`.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
CartId=102-5929035-5792105&  
HMAC=[HMAC]=&  
Operation=CartAdd &  
Item.1.OfferListingId=1400042127& Item.1.Quantity=2  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

This request adds to the cart two (Quantity=2) of the same items, which are identified by the `OfferListingId` (1400042127).

As you can see, the method of identifying the item and its quantity to add to the cart is the same as it was for `CartCreate`. For more information, see [CartCreate \(p. 214\)](#).

Adding Items as Saved For Later

To add an item to the `SaveForLater` area, use "SaveForLater" as the value for the `Action` parameter in a `CartModify` operation, for example:

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=CartModify&  
CartId=[Cart ID]&  
HMAC=[HMAC]=&  
Item.1.CartItemId=[Cart Item ID]&  
Item.1.Quantity=1  
Item.1.Action=SaveForLater  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

In this request, the item identified by `Item.1` is moved to the cart's `SaveForLater` area. The `Quantity` parameter enables you to move only some of one type of item into the `SaveForLater` area, for example, if you had fifteen copies of the book, *Saving Miss Oliver's*, in the cart, you could move eight of them to the `SaveForLater` area by setting `Item.1.Quantity=8` in the preceding request.

Notice that only `CartModify` can move an item between the `Active` and `SaveForLater` areas. This means that an item cannot be added directly to the `SaveForLater` area; it must first be added to the cart using `CartCreate` or `CartAdd` and then moved to `SaveForLater`. Amazon will automatically move an item to the `SaveForLater` area if the item is currently unavailable. `CartModify` will not be able to move that item into the `Active` area until it becomes available. At that time, Amazon will move the item automatically into the `Active` area.

Note

The value for the `Action` parameter is "SaveForLater." Items in the `SaveForLater` area are tagged in the XML response with the element, `SavedForLaterItem`, as shown in the following XML snippet from a response.

```
<SavedForLaterItem>
  <CartItemId>ULI7S9IYFJHX0</CartItemId>
  <ASIN>B0009GZV4A</ASIN>
  <Quantity>2</Quantity>
  <Title>Mark VII Men's Short Sleeve Golf Shirts with
    Tri -Colored Stripe Trim</Title>
  <ItemTotal>
    <Amount>1288</Amount>
    <CurrencyCode>USD</CurrencyCode>
    <FormattedPrice>$12.88</FormattedPrice>
  </ItemTotal>
</SavedForLaterItem>
```

To move an item from the SaveForLater area to the Active area, use "MoveToCart" as the value for the Action parameter in a CartModify operation:

```
Item.1.CartItemId=[Cart Item ID]&
Item.1.Quantity=1
Item.1.Action=MoveToCart
```

Removing Items From a Cart

A customer might, at times, decide to remove some or all of the items in their remote shopping cart.

To remove all items from a cart

- Use CartClear to remove all items from a cart.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate Tag]&
Operation=CartClear&
CartId=002-2041347-9034467&
HMAC=[HMAC]
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

If the customer, however, wants to remove only some of the items in their cart, use CartModify and set the Quantity and CartItemId parameters appropriately, for example:

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate Tag]&
Operation=CartModify&
CartId=[cart ID]&
Item.1.CartItemId=U1I8M9790QFD07&
Item.1.Quantity=15&
Item.2.CartItemId=U3K5GRHEXU6FHK&
Item.2.Quantity=0
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

This request sets the quantity of the item specified by `CartItemID=U3K5GRHEXU6FHK&` to 0, thereby removing it from the cart.

Retrieving the Contents of a Cart

You should not maintain a local copy of the remote shopping cart. Instead, use `CartGet` to retrieve the items in a shopping cart.

Retrieving the items in a cart

- Use the cart's `HMAC` and `CartID` values in a `CartGet` request.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=CartGet&  
CartId=002-2041347-9034467&  
HMAC=[HMAC]  
  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

By default, `CartGet` uses the `Cart` response group. This response group provides a lot of information about items, including, for example, their price, quantity, and the seller ID of the seller selling the item, as shown in the following response snippet.

```
<PurchaseURL>https://www.amazon.com/gp/cart/aws-merge.html?cart-id=002-9918938-  
1696046%26associate-id=ws%26hmac=b0ogzvivVYLNjSZ9WwoBRFesFYU=%26AWSAccessKey  
Id=[AWS Access Key ID]</PurchaseURL>  
<CartItems>  
  <SubTotal>  
    <Amount>1994</Amount>  
    <CurrencyCode>USD</CurrencyCode>  
    <FormattedPrice>$19.94</FormattedPrice>  
  </SubTotal>  
  <CartItem>  
    <CartItemID>U3KYV0C66V3PAA</CartItemID>  
    <ASIN>B000062TU1</ASIN>  
    <Quantity>2</Quantity>  
    <Title>Harry Potter and the Sorcerer's Stone (Full Screen Edition)(Harry  
Potter 1)</Title>  
    <ProductGroup>DVD</ProductGroup>  
    <Price>  
      <Amount>997</Amount>  
      <CurrencyCode>USD</CurrencyCode>  
      <FormattedPrice>$9.97</FormattedPrice>  
    </Price>  
    <ItemTotal>  
      <Amount>1994</Amount>  
      <CurrencyCode>USD</CurrencyCode>  
      <FormattedPrice>$19.94</FormattedPrice>  
    </ItemTotal>
```

```
</CartItem>  
</CartItems>
```

You could, for example, use the `CartItem.Id` in a `CartModify` operation to change the quantity of that item. You could also use the `PurchaseURL` to purchase the items in the cart.

Add to Cart Form

The Add to Cart form enables you to add any number of items to a customer's shopping cart and send the customer to the Amazon retail web site.

Note

There is no direct way for a customer to get back to your site after getting sent to the Amazon site. To improve a customer's experience on your site, you may want to note this on your site or provide a way to get back to your site (using a new window, for example).

Although the parameters are optional, but you must specify at least one of the following parameters: ASIN or OfferListingId.

To add more than one item to the customer's cart

- Append each set of parameters with a period, then a unique identifier, which establishes a relationship between the parameters (for example, "ASIN.1=[[ASIN](#)]&Quantity.1=1&ASIN.2=[Another ASIN]&Quantity.2=12").

The Add to Cart form works with all locales. Send the Add to Cart form data to one of the following URLs:

- <http://www.amazon.com/gp/aws/cart/add.html>
- <http://www.amazon.co.jp/gp/aws/cart/add.html>
- <http://www.amazon.co.uk/gp/aws/cart/add.html>
- <http://www.amazon.de/gp/aws/cart/add.html>
- <http://www.amazon.fr/gp/aws/cart/add.html>
- <http://www.amazon.ca/gp/aws/cart/add.html>

The following table describes the input parameters for the Add to Cart form.

Parameter	Description	Required
ASIN.x	Specifies one or more product ASINs to add, where x is a unique identifier.	Optional
OfferListingId.x	An alternative way to specify one or more product offer listings from third-party sellers, where x is a unique identifier.	Optional
AWSAccessKeyId	Your Access Key ID Here. You may want to specify this as a hidden parameter.	Required
AssociateTag	Your associate tag. You may want to specify this as a hidden parameter.	Required

Example HTML for the Add to Cart Form

```
<form method="GET" action="http://www.amazon.com/gp/aws/cart/add.html">
<input type="hidden" name="AWSAccessKeyId" value="Access Key ID" /><br/>
<input type="hidden" name="AssociateTag" value="Associate Tag" /><br/>
<p>One Product<br/>
ASIN:<input type="text" name="ASIN.1"/><br/>
Quantity:<input type="text" name="Quantity.1"/><br/>
<p>Another Product<br/>
ASIN:<input type="text" name="ASIN.2"/><br/>
Quantity:<input type="text" name="Quantity.2"/><br/>
</p>
<input type="submit" name="add" value="add" />
</form>
```

Purchase Items in a Remote Shopping Cart

Topics

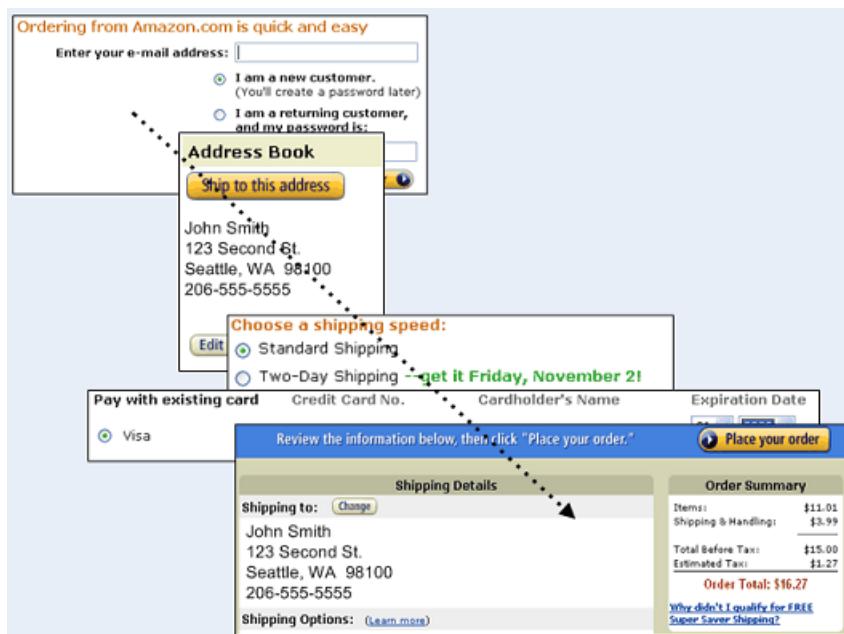
- [Order Pipeline \(p. 167\)](#)
- [Implement the Proceed to Checkout Button \(p. 168\)](#)
- [PurchaseURL \(p. 168\)](#)

When customers finish shopping and are ready to purchase items in their shopping cart, they choose a button, for example, **Proceed to Checkout**. This button must submit the value of the PurchaseURL. When customers click this button, the items in the Active area are moved from the shopping cart to the Order Pipeline.

This section explains how to submit the contents of a remote shopping cart for purchase.

Order Pipeline

The Order Pipeline is a series of Amazon web pages that guides the customer through the checkout process, which includes adding the customer's name, shipping and billing addresses, payment method, and a purchase confirmation web page, as shown in the following.



Several report types are available, such as Earnings, Daily Trends, and so on. Conversion rate reports help you understand how often carts are turning in to orders. For more information, see [Amazon Associates Reports](#).

Note

Note that Amazon does not provide per-customer order identification.

Implement the Proceed to Checkout Button

Amazon hosts the web pages in the Order Pipeline. Your application plays no role in purchasing the items or order fulfillment. However, your application must use the `PurchaseURL` returned by Product Advertising API cart operations to add the shopping cart items to the Order Pipeline. The `PurchaseURL` is often an HTML form and button labeled, for example, **Proceed to Checkout**.



PurchaseURL

All cart operations except `CartClear` return a value for `PurchaseURL`.

Example:

```
<PurchaseURL>https://www.amazon.com/gp/cart/aws-merge.html?cart-id=002-9918938-1696046%26associate-id=[Your ID]%26hmac=b0ogzvivVYLxjSZ9WwoBRFesFYU=%26AWSAccessKeyId=[Access Key ID]<PurchaseURL>
```

This value is a URL that should be submitted, like a request, to purchase the items in a remote shopping cart. The value contains the Associate's Tag, which is specified in the `CartCreate` operation. The Associate's Tag value in the `PurchaseURL` links the customer's purchase to the Associate. If `PurchaseURL` is not used, the Associate will not receive credit for the sale.

The information in bold includes:

- Cart identity (cart-id, hmac)

- Associate identity (associate-id)
- Request submitter (AWSAccessKeyId)

You can manually change any of the values in the `PurchaseURL`, but this is not recommended.

Troubleshooting

Topics

- [Syntax and Parameter Errors \(p. 169\)](#)
- [Retrieving Errors \(p. 171\)](#)
- [Troubleshooting Applications \(p. 171\)](#)
- [Error Codes and Messages \(p. 172\)](#)

Product Advertising API provides specific and descriptive errors to help you troubleshoot problems with your requests. There are two kinds of errors, as explained in the following sections.

Syntax and Parameter Errors

Topics

- [Processing Error \(p. 170\)](#)
- [Results and Errors \(p. 171\)](#)

All responses contain an `IsValid` element, for example:

```
<IsValid>False</IsValid>
```

The `IsValid` element tells you whether or not there is an error in the syntax of any elements of the request and if all required parameters are included. For example, if you were to omit the `Operation` parameter, which is required in every request, Product Advertising API would set `IsValid` to False and not process your request. Typically, these kinds of errors are a little more subtle. Typical errors are incorrectly capitalized parameters or values, or the failure to include a required parameter in the request. For example, in the following request, the `SearchIndex` parameter is entered as "Searchindex".

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
Keywords=Potter&  
Searchindex=Books  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Product Advertising API returns the following error:

```
<Errors>  
<Error>  
  <Code>AWS.MissingParameters</Code>  
  <Message>Your request is missing required parameters. Required parameters
```

```
include SearchIndex.  
  </Message>  
  </Error>  
</Errors>
```

Product Advertising API believes that the request is missing the `SearchIndex` parameter just because the "I" was not capitalized in the parameter name.

In the sample response, notice that the error consists of an error code that identifies the error, and an error message that describes the error. The error messages is in the language of the locale.

For a detailed list of error codes and messages, see [Error Codes and Messages \(p. 172\)](#) section of the API Reference.

Processing Error

Is it possible to submit a valid request and still have an error? Yes. If you were to submit a request and no items in Amazon satisfied the request, you would receive an error. The following request is an example of this problem.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
Condition=Refurbished&  
ResponseGroup=Images&  
SearchIndex=Automotive&  
Title=Harry%20Potter  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

The following response snippet shows the error.

```
<IsValid>True</IsValid>  
...  
<Errors>  
  <Error>  
    <Code>AWS.ECommerceService.NoExactMatches</Code>  
    <Message>We did not find any matches for your request.  
    </Message>  
  </Error>  
</Errors>
```

Looking for a refurbished edition of a Harry Potter book in the Automotive section doesn't make much sense. So, it's not surprising that there were no items that satisfied that request. Similar problems can be much more subtle, however. For example, the default value of `Condition` is "New". In part, this is because Amazon only sells new items. If you submitted a request without specifying a value for `Condition` and received no results, you might assume that there are no items that satisfy the request. By setting `Condition` to "All", you might find that items do satisfy the request.

A similar error message is displayed when, for example, you use `ItemLookup` to find an item that does not exist in the Amazon catalog.

```
<Errors>
<Error>
<Code>AWS.InvalidParameterValue</Code>
<Message>B0111NOSUCHITEM is not a valid value for ItemId. Please change
this value and retry your request.
</Message>
</Error>
</Errors>
```

ItemId was set equal to "B0111NoSuchItem" in the request. The error message shows that Product Advertising API converts *ItemId* values to all caps.

Results and Errors

As long as the request is valid, Product Advertising API will try to process it. There are times when Product Advertising API returns a result and an error. For example, if you had a batch request, one item might be found, the other not. In that case, *IsValid* would be true, the response would contain the item attributes of the item found, and the response would also contain an error message that the second item could not be found. Product Advertising API attempts to return as much information as possible even when an error prevents the successful completion of a portion of a request.

Retrieving Errors

Typically, you want your application to check whether or not a request generated an error before spending any time processing results. The easiest way to find out if an error occurred is to look for an *Error* node in the response.

XPath syntax provides a simple way to search for the presence of an *Error* node, as well as an easy way to retrieve the error code and message. The following code snippet uses Perl and the XML::XPath module to determine if an error occurred during a request. If an error occurred, the code prints the first error code and message in the response.

```
use XML::XPath;
my $xp = XML::XPath->new(xml =>$response);
if ( $xp->find("//Error") )
{print "There was an error processing your request:\n", " Error code: ",
$xp->findvalue("//Error[1]/Code"), "\n", " ",
$xp->findvalue("//Error[1]/Message"), "\n\n"; }
```

Troubleshooting Applications

We recommend the following processes to diagnose and resolve problems with your Product Advertising API–enabled websites or applications.

- Use the [Product Advertising API Scratchpad](#) tool to troubleshoot your requests.
- Verify that Product Advertising API is running correctly.

To do this, open a browser window and submit a REST request. This will confirm that the service is available and responding to requests. Normally, Product Advertising API is available 24 hours a day, 7 days a week.

- Check that you are specifying inclusive response groups.

You may not receive the expected information in your Product Advertising API response if you have specified a response group(s) that does not return the data you want.

- **Check the structure of your request.**

Each Product Advertising API operation has a reference page. Double check that you are using parameters correctly. Try the request on one of the other locales. If SOAP is not working, try your request with REST through your browser. That will help you determine if the problem lies with your code, your SOAP client, or with Product Advertising API.

- **Try your request with multiple ASINs or keyword/title search strings.**

The Amazon catalog is extremely large and sometimes individual products have irregular data results.

- **Check how quickly your application is submitting requests.**

If your application is submitting requests faster than once per second per IP address, you may receive error messages from Product Advertising API until you decrease your rate of requests.

Efficiency Guidelines

If your application is trying to submit requests that exceed the maximum request limit for your account, you may receive error messages from Product Advertising API. The request limit for each account is calculated based on revenue performance. Each account used to access the Product Advertising API is allowed an initial usage limit of 1 request per second. Each account will receive an additional 1 request per second (up to a maximum of 10) for every \$4,600 of shipped item revenue driven in a trailing 30-day period (about \$0.11 per minute). You can verify that your sales are being attributed to your calls to the Product Advertising API by checking for the following:

- You are using the links provided by the API when linking back to Amazon.
- Your Associate account and Product Advertising API account were created using the same Amazon account (i.e. email address).
- You are passing your Associate tag in all your requests to the API.

Error Codes and Messages

Product Advertising API errors provide information about syntactical errors in your requests, as well as errors that occur during the execution of your request; for example, a search for products returns no results.

Errors are composed of two elements:

code

The error code is a unique string that identifies the error.

message

The error message is a human-readable description of the error to help you debug the issue.

These elements will be nested within an Error element. If a request generates more than one error, all Errors will appear in the response.

Errors may appear at different levels in your response. Their location reflects at what stage in the execution of the request the error was generated and the type of error. Errors in syntax that prevent requests from being executed will appear as children of the response's root element. An error associated with a particular item in the response will be a child of the Item element. See the sample requests for examples of each of these situations.

Error Codes

Product Advertising API returns errors in three categories, so you can easily determine how best to handle the problem:

2XX errors

These errors are caused by mistakes in the request. For example, your request might be missing a required parameter. See the error message in the response to see what is wrong.

4XX errors

These errors indicate an issue with the request. A 403 indicates the request was not authenticated correctly.

5XX errors

These errors are transient errors reflecting an error internal to Amazon. A 503 error means that you are submitting requests too quickly and your requests are being throttled. If this is the case, you need to reduce the number of requests you are sending per second.

Error Messages

Product Advertising API returns error messages in English for the Amazon.com (US), Amazon.co.uk (UK), Amazon.de (DE), Amazon.fr (FR), and Amazon.ca (CA) locales.

Error messages are in Japanese for the Amazon.co.jp (JP) locale.

Error Code & Description	Message	Affected Operations
AWS.ExactParameterRequirement You will receive this message when the value of your parameter is longer than permitted by Product Advertising API.	Your request contains too much data for <i>[ParameterName]</i> . This parameter can have a maximum length of <i>[Maximum-Number]</i> .	All
AWS.ExceededMaximumParameterValue You will receive this error message when you specify too many values for one or more parameters in your request.	Your request contains too many values for <i>[ParameterName]</i> . This parameter can have a maximum of <i>[MaximumNumber]</i> values.	ItemLookup SimilarityLookup
AWS.InsufficientParameterValues You will receive this error message when your request contains an insufficient number of values for a required parameter.	Your request contains too few values for <i>[Parameter Name]</i> . This parameter must have a minimum of <i>[Minimum Value]</i> values.	All
AWS.InternalError You will receive this error if Product Advertising API is unable to complete your request due to an internal problem or outage. For SOAP, this will be presented as a SOAP fault rather than an error.	We are unable to process your request at this time. Please retry your request. If you encounter this error repeatedly, please post a message on the AWS discussion board.	All

Error Code & Description	Message	Affected Operations
AWS.InvalidAccount You will receive this error when you try to use an AwsAccessKeyId in a locale where it is not registered for the Product Advertising API.	Your AccessKey Id is not registered for Product Advertising API. Please use the AccessKey Id obtained after registering at [RegistrationUrlForThatLocale]	All
AWS.InvalidEnumeratedParameter You will receive this error message when your request contains an invalid value for a parameter that has an explicit list of valid values, such as SearchIndex.	The value you specified for [<i>ParameterName</i>] is invalid. Valid values include [<i>EnumeratedValuesList</i>].	CartAdd CartCreate CartModify ItemLookup ItemSearch SimilarityLookup
AWS.InvalidISO8601Time You will receive this error when your request contains a date or time value that is not formatted according to the profile of the ISO-8601 date/time standard that is described at http://www.w3.org/TR/NOTE-datetime . For example, this error will be returned if your request contains an invalid value for the Version parameter.	[<i>ParameterName</i>] has an invalid value. It must contain a valid ISO 8601 date and time.	All
AWS.InvalidOperationForMarketplace You will receive this error message when you try to execute an operation in a locale where the operation is not supported.	This operation, [<i>OperationName</i>], is not available for this locale.	All
AWS.InvalidOperationParameter You will receive this error message when the operation name you entered is not available from Product Advertising API. For instance, if you tried to use AsinSearch (from AWS 3.0) as an operation name, you would get this error since AsinSearch is not a valid operation name in Product Advertising API 4.0.	The Operation parameter is invalid. Please modify the Operation parameter and retry. Valid values for the Operation parameter include [<i>ListOfOperationValues</i>].	All

Error Code & Description	Message	Affected Operations
AWS.InvalidParameterCombination You will receive this error message when two or more of the request parameters you have entered can not be used in the same request. For example, if you are using the CartAdd operation, you would receive this error if you tried to add items to the cart by both ASIN and OfferListingId.	Your request contains an invalid parameter combination. [ParameterName] and [ParameterName] cannot appear in the same request.	ItemSearch CartCreate CartAdd
AWS.InvalidParameterValue You will receive this error message when your request contains an invalid value for an ID parameter, such as ItemId.	[ParameterValue] is not a valid value for [ParameterName] . Please change this value and retry your request.	CartAdd CartCreate CartModify ItemLookup ItemSearch SimilarityLookup
AWS.InvalidResponseGroup You will receive this error message when the response group name you entered in your request is incompatible with the operation you would like to perform.	Your ResponseGroup parameter is invalid. Valid response groups for [Operation Name] requests include [Available Response Group List] .	All
AWS.InvalidServiceParameter You will receive this error message when the service name you provide in your request is not recognized or supported by Amazon. All Product Advertising API requests should use the service name "AWSECommerceService."	The Service parameter is invalid. Please modify the Service parameter and retry. Valid values for the Service parameter include [ValidServicesList] .	All
AWS.InvalidSubscriptionId You will receive this error message when the subscription ID you use in your request is not recognized by AWS.	Your request contains an invalid subscription ID. Please retry your request with a valid subscription ID.	All
AWS.MaximumParameterRequirement You receive this error message when your request contains the wrong number of parameters from an exclusive group.	Your request should have at most [Maximum Number] of the following parameters: [Parameter Names] .	All
AWS.MinimumParameterRequirement You receive this error message when your request contains the wrong number of parameters from an exclusive group.	Your request should have at least [Minimum Number] of the following parameters: [Parameter Names] .	All

Error Code & Description	Message	Affected Operations
AWS.MissingOperationParameter You will receive this error message when your request does not include the Operation parameter and the name of the operation you would like to perform.	Your request is missing the Operation parameter. Please add the Operation parameter to your request and retry. Valid values for the Operation parameter include [ValidOperationsList] .	All
AWS.MissingParameterCombination You will receive this error message when your request does not contain a combination of two or more parameters that must be present together in your request.	Your request is missing a required parameter combination. Required parameter combinations include [Parameter One] .	ItemLookup
AWS.MissingParameters You will receive this error when your request does not include all of the parameters required by the operation.	Your request is missing required parameters. Required parameters include [RequiredParameterList] .	All
AWS.MissingParameterValueCombination You will receive this error message when your request requires a combination of parameters, one or more of which must have a specific value. For example, when you make an ItemLookup request for a product based on its Universal Product Code (or UPC), you are required to include the IdType and ItemId parameters. The value of the IdType parameter must be UPC.	Your request is missing a required parameter combination. When [Parameter One] equals [Restricted Value] , [Parameter Two] must be present.	ItemLookup
AWS.MissingServiceParameter You will receive this error message when the your request does not contain the required Service parameter.	Your request is missing the Service parameter. Please add the Service parameter to your request and retry. Valid values for the Service parameter include [ValidServicesList] .	All
AWS.ParameterOutOfRange You will receive this error message when you submit a parameter value that exceeds or is lower than the range of valid values for the parameter. For example, ItemSearch allows you to fetch search results page using the ItemPage parameter. The range of values for ItemPage is 1 to 10. If you supply a value outside that range (less than 1 or greater than 10), you will receive this error.	The value you specified for [ParameterName] is invalid. Valid values must be between [LowerBound] and [UpperBound] .	ItemSearch ItemLookup

Error Code & Description	Message	Affected Operations
AWS.ParameterRepeatedInRequest You receive this error message when you include the same parameter more than once in your request.	The parameter, <i>[ParameterName]</i> , appeared more than once in your request.	All
AWS.RestrictedParameterValueCombination You will receive this error message when your request contains a combination of parameter values that are not permitted in the same request.	Your request contains a restricted parameter combination. When <i>[Parameter One]</i> equals <i>[Restricted Value]</i> , <i>[Parameter Two]</i> cannot be present.	All
AWS.ECommerceService.ExceededMaximumCartItems You will receive this error message when you exceed the maximum quantity value allowed for items being added to a shopping cart.	You may not add more than <i>[Maximum Item Quantity]</i> items to the cart.	CartAdd CartCreate
AWS.ECommerceService.InvalidCartId You will receive this error message when the CartId you entered into your request is not recognized.	Your request contains an invalid value for CartId. Please check your CartId and retry your request.	CartAdd CartClear CartGet CartModify
AWS.ECommerceService.InvalidHMAC You will receive this error message when the shopping cart HMAC value you use in your request is not recognized by Product Advertising API. The HMAC value is a unique token that is used to associate a cart with an Amazon user and a particular session on the Amazon web site.	Your request contains an invalid value for HMAC. Please check your HMAC and retry your request. Remember that the HMAC must be URL-encoded if you are using REST.	CartAdd CartClear CartGet CartModify
AWS.ECommerceService.InvalidQuantity You will receive this error message when the quantity in your request is not valid for the current item.	You have exceeded the maximum quantity allowed for the following item(s): <i>[ItemId]</i> .	CartAdd CartCreate CartModify
AWS.ECommerceService.ItemAlreadyInCart You will receive this error message when you try to add an item to a shopping cart that already contains that item.	The item you specified, <i>[ItemID]</i> , is already in your cart.	CartAdd CartCreate

Error Code & Description	Message	Affected Operations
AWS.ECommerceService.ItemNotAccessible Some products cannot be manipulated or viewed using Product Advertising API. You will receive this error message when the product ID you use in your request is not available through Product Advertising API.	This item is not accessible through Product Advertising API.	ItemLookup
AWS.ECommerceService.ItemNotEligibleForCart Some products cannot be manipulated or viewed using Product Advertising API. You will receive this error message when you attempt to add such an item to a remote shopping cart.	The item you specified, <i>[ItemID]</i> , is not eligible to be added to the cart. Check the item's availability to make sure it is available.	CartAdd CartCreate CartModify
AWS.ECommerceService.NoExactMatches	We did not find any matches for your request.	ItemSearch
AWS.ECommerceService.NoSimilarities	There are no similar items for this ASIN(s): <i>[ItemID]</i> .	SimilarityLookup
RequestThrottled You will receive this error if your application is submitting requests faster than allowed by IP or account. For more information about rates, see the Efficiency Guidelines section in Troubleshooting Applications (p. 171) .	Request from <i>[ID IP]</i> is throttled.	All

Best Programming Practices

Topics

- [Read the Product Advertising API Terms and Conditions \(p. 179\)](#)
- [Use the Latest API Version \(p. 179\)](#)
- [Understand Available Operations \(p. 179\)](#)
- [Use the Right Response Group \(p. 179\)](#)
- [Use Your Associate Tag in Product Advertising API Requests \(p. 180\)](#)
- [Handling Errors \(p. 180\)](#)
- [Use Caches Carefully \(p. 180\)](#)
- [Use the Correct AWSAccessKeyId \(p. 180\)](#)

The following checklist of best practices describes how you can increase the effectiveness of your Product Advertising API 4.0 applications.

Read the Product Advertising API Terms and Conditions

The Product Advertising API Terms and Conditions spell out in detail the limitations that Amazon enforces on all Product Advertising API applications. The thrust of all Product Advertising API applications should be to direct sales to Amazon and thus earn Associate sales commissions. If your application is designed around another purpose, please reconsider and make sure your proposed Product Advertising API application falls within the guidelines of the Product Advertising API Terms and Conditions. Applications that do not meet the Product Advertising API Terms and Conditions will be blocked from accessing Product Advertising API.

Use the Latest API Version

Product Advertising API makes frequent releases. Each release either adds functionality or increases the accuracy, speed, and stability of Product Advertising API.

The Product Advertising API uses the default API version 2013-08-01 unless you supply a different value for the `Version` parameter. To use a different version, you must include it in the `Version` parameter in your requests.

Product Advertising API 3.0 has been deprecated. Make sure you develop against Product Advertising API 4.0.

Understand Available Operations

Product Advertising API provides many different operations to facilitate product discovery. Using the right operations can dramatically enhance your customer's shopping experience and increase your Associate commissions. See the API Reference for a full description of all Product Advertising API operations. Some enable you to:

Task	Operations
Find products and categories	<code>ItemSearch</code> , <code>ItemLookup</code> , and <code>BrowseNodeLookup</code>
Find similar items	<code>SimilarityLookup</code>
Provide shopping cart functionality for your application or website	<code>CartCreate</code> , <code>CartAdd</code> , <code>CartModify</code> , and <code>CartGet</code>

Use the Right Response Group

One of the great features of Product Advertising API is the control you have over the amount of information returned in responses. A response group is a collection of data returned by Product Advertising API. Product Advertising API has over 55 response groups, each serving a different need. So, you can get exactly the information you need and no more.

We recommend that you specify response groups that return only the information your application needs. Response groups, such as Large and ItemAttributes return lots of data. Such large data sets sometimes incur performance penalties both in Product Advertising API fulfilling the request and in your application's processing of the response.

Use Your Associate Tag in Product Advertising API Requests

To earn commissions for selling Amazon items, you must register with Amazon as an Associate. In return, you receive an Associate tag, which identifies you. By including your Associate tag in each Product Advertising API request, you receive commissions for customer purchases.

Amazon also uses your Associate tag to monitor your use of Product Advertising API, which helps us determine how to improve our web service.

Handling Errors

Make sure your application handles errors gracefully. One way to do that is to check the status of the `IsValid` element. `IsValid` is returned with every request. If its value is "False," there will be an error message with a description of why your request was not valid.

It's generally a good practice to log any unexpected error that is returned by Product Advertising API. Errors are returned with an error code and message. The code is a descriptive string that identifies the error. The error message is a more 'human friendly' message that can be displayed to your customers. Your application should be able to handle expected error messages.

Your application, for example, should display error messages that are meaningful to the customer. For example, when Product Advertising API responds with the error, "AWS.ECommerceService.NoExactMatches," your application should display an explanation, such as, "We did not find any matches for your request."

If your application exceeds the number of allowed requests submitted per second, Product Advertising API returns a 503 error, which means that Product Advertising API is restricting the number of requests it is processing from your application. The Product Advertising API Terms and Conditions outline the number of allowed requests permitted per second.

Use Caches Carefully

Product Advertising API product data changes often. Prices can change hourly, Browsenode values change without notice, and product availability is volatile. For these reasons, you should not cache product data.

It is against the Product Advertising API Terms and Conditions to cache customer information derived from Amazon.

You can enhance the performance of your application by caching identifiers, such as the `CartId`, `HMAC`, and `PurchaseURL`.

Use the Correct AWSAccessKeyId

To become a Product Advertising API developer, you must retrieve your AWS credentials. Your credentials are a `AWSAccessKeyId` and a `SecretKey`. Every Product Advertising API request you submit must include the `AWSAccessKeyId`. Because your account is linked to your email address, Amazon can contact you easily. For more information, see [Becoming a Product Advertising API Developer \(p. 5\)](#).

Locale Considerations

Topics

- [Locales \(p. 181\)](#)
- [Associate IDs \(p. 182\)](#)
- [General Differences \(p. 182\)](#)
- [Shipping Restrictions \(p. 182\)](#)

There are subtle differences in Product Advertising API functionality across locales. The operations are the same for all locales but the valid response groups, search indices, and sort values vary for locales.

Locales

The Product Advertising API is available in the following locales:

Locale	URL
Brazil	http://www.amazon.com.br
Canada	http://www.amazon.ca
China	http://www.amazon.cn
France	http://www.amazon.fr
Germany	http://www.amazon.de
India	http://www.amazon.in/
Italy	http://www.amazon.it
Japan	http://www.amazon.co.jp
Mexico	http://www.amazon.com.mx
Spain	http://www.amazon.es
United Kingdom	http://www.amazon.co.uk/
United States	http://www.amazon.com

Each of these locales is serviced by an Amazon web site that uses the local language, local customs, and local formatting. For example, when you look at the DE homepage for Amazon, you see the listings in German. If you purchased an item, you would find the price in Euros, and, if you were to purchase a movie, you would find that the movie rating would conform to the movie rating system used in Germany. Product Advertising API responses contain the same localized information. Product Advertising API determines the correct locale by examining the endpoint in the request. For example, the endpoints for the DE locale are:

```
http://webservices.amazon.de/onca/xml
https://webservices.amazon.de/onca/xml
```

For a list of the other endpoints, see [Anatomy Of a REST Request \(p. 41\)](#).

Currency, for example is localized, as follows.

```
<SubTotal>
<Amount>4082</Amount>
<CurrencyCode>USD</CurrencyCode>
```

```
<FormattedPrice>$40.82</FormattedPrice>
</SubTotal>
```

The values for *CurrencyCode* are similar to the names of Product Advertising API locales. In the preceding example, the currency is formatted according to the conventions in the US locale.

Associate IDs

To be an Associate, you must sign up in each locale in which you intend to do business. That means, for example, if you have an Associate ID for the US locale, you will not get credit if you submit a *PurchaseURL* in the DE locale. To get credit, you must get an Associate ID in the DE locale. For more information, see [Becoming an Associate \(p. 4\)](#).

General Differences

Amazon marketplaces are localized by language, custom, and formatting conventions. In addition, the locales may vary, as described in the following table.

Difference	Description
Items for sale	The selection of DVDs, for example, sold in the JP locale are different than those sold in the DE locale. The same is true for books. Not only are the languages of the books different, the selection of books varies by locale.
Customer feedback	Amazon encourages customers to share their feedback on items, sellers and merchants. Customer reviews are restricted to the locale in which they are entered.
Sellers and merchants	Sellers and merchants can do business in multiple locales but they don't have to. As a result, items from a seller might be available in only one locale. The largest merchants, beside Amazon, are available in the US locale only.
Customer accounts	Customer accounts are restricted to a locale. A customer can create accounts in every locale.
Item identifiers	Item identifiers, such as ASINs, are unique to a locale, that is, the same ASIN value can refer to different items in different locales.
Availability of some response groups	The availability of some response groups, sort parameters, and search indices varies by locale.

Shipping Restrictions

Amazon places shipping restrictions based on item and locale. The sale of items must obey the rules and regulations of the host country. It is possible, for example, that a host country would restrict the sale of some electronic equipment to specific countries. Some locales restrict shipping to specified countries regardless of the item. For example, in the UK locale, shipping is restricted to Ireland, Scotland, and England.

Because Amazon is used to fulfill the orders, Amazon takes charge of restricting shipping. Your application or website, however, should be aware of the shipping restrictions.

Shipping costs also play an important role in helping your customers purchase items for the lowest possible price. For example, Harry Potter books are available in all locales. A customer in the US could purchase a copy of a Harry Potter book from the UK locale. The shipping charges, however, would be significantly higher than if the same item were purchased through the US locale.

To see the latest details on shipping restrictions for each locale, see [About Shipping Restrictions](#).

API Reference

Topics

- [Operations \(p. 184\)](#)
- [Common Request Parameters \(p. 228\)](#)
- [Response Groups \(p. 230\)](#)
- [Response Elements Common to All Response Groups \(p. 309\)](#)
- [Response Elements \(p. 310\)](#)
- [Locale Reference \(p. 328\)](#)
- [ItemSearch Sort Values \(p. 500\)](#)

The following sections of the guide provide reference material for the Product Advertising API. For more information about any concepts or programming tasks associated with the reference material, refer to the previous chapters in this guide.

Operations

The following operations are available in the Product Advertising API.

Search

- [ItemSearch \(p. 185\)](#)

Lookup

- [BrowseNodeLookup \(p. 194\)](#)
- [ItemLookup \(p. 197\)](#)
- [SimilarityLookup \(p. 203\)](#)

Cart

- [CartAdd \(p. 207\)](#)
- [CartClear \(p. 212\)](#)
- [CartCreate \(p. 214\)](#)

- [CartGet \(p. 220\)](#)
- [CartModify \(p. 224\)](#)

ItemSearch

Description

The `ItemSearch` operation returns items that satisfy the search criteria, including one or more search indices.

`ItemSearch` returns up to ten search results per page. When `condition` equals "All", `ItemSearch` returns additional offers for those items, one offer per condition type (if an offer exists)—for example, one new, one used, one refurbished, and one collectible item. Or, for example, if there are no collectible or refurbished offers, `ItemSearch` returns one new and one used offer.

Because there are thousands of items in each search index, `ItemSearch` requires that you specify the value for at least one parameter in addition to a search index. The additional parameter value must reference items within the specified search index. For example, you might specify a browse node (`BrowseNode` is an `ItemSearch` parameter), Harry Potter Books, within the Books product category. You would not get results, for example, if you specified the search index to be Automotive and the browse node to be Harry Potter Books. In this case, the parameter value is not associated with the search index value.

The `ItemPage` parameter enables you to return a specified page of results. The maximum `ItemPage` number that can be returned is 10. An error is returned if you try to access higher numbered pages. If you do not include `ItemPage` in your request, the first page will be returned by default. There can be up to ten items per page.

The `ItemSearch` operation is commonly used. In general, use this operation to find an item.

For more discussion about finding items by `BrowseNode`, see [Understanding BrowseNode Results When Drilling Down \(p. 94\)](#).

Availability

All locales.

Request Parameters

`ItemSearch` has a lot of parameters. Not all of them pertain, however, to all search indices. For example, when the search index is apparel, it would be inappropriate to use the `Actor` parameter. As a result, each search index can use only a subset of all of the parameters. For a complete list of the `ItemSearch` parameters that can be used with a specific search index in a specific locale, see [Locale Reference \(p. 328\)](#).

The parameters that apply to the largest number of search indices are shown in the following table.

Parameter	Valid Search Indices
<code>BrowseNode</code>	Every index except All and Blended
<code>Condition</code>	Every index except All and Blended
<code>Keywords</code>	All
<code>MaximumPrice</code>	Every index except All and Blended

Parameter	Valid Search Indices
<i>MinimumPrice</i>	Every index except All and Blended
<i>Title</i>	Every index except All and Blended

ItemSearch requires that you specify a search index and at least one of the following parameters:

<ul style="list-style-type: none"> • Actor • Artist • AudienceRating • Author • Brand • BrowseNode 	<ul style="list-style-type: none"> • Composer • Conductor • Director • Keywords • Manufacturer 	<ul style="list-style-type: none"> • MusicLabel • Orchestra • Power • Publisher • Title
--	---	--

Name	Description	Required
<i>Actor</i>	Name of an actor associated with the item. You can enter all or part of the name. Type: String Default: None	No
<i>Artist</i>	Name of an artist associated with the item. You can enter all or part of the name. Type: String Default: None	No
<i>AudienceRating</i>	Movie ratings based on MPAA ratings or age, depending upon the locale. You may specify one or more values in a comma-separated list in a REST request or by using multiple elements in a SOAP request. Type: String. Type: String Default: None Valid Values: See Movie Ratings by Locale (p. 192) , which follows this table.	No
<i>Author</i>	Name of an author associated with the item. You can enter all or part of the name. Type: String Default: None	No

Name	Description	Required
<i>Availability</i>	<p>Enables ItemSearch to return only those items that are available. This parameter must be used in combination with a merchant ID and <i>Condition</i>. For more information, see Availability Parameter (p. 74), which follows this table. When Availability is set to "Available", the Condition parameter cannot be set to "New".</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Available</p>	Yes
<i>Brand</i>	<p>Name of a brand associated with the item. You can enter all or part of the name.</p> <p>Type: String, for example, Timex, Seiko, Rolex.</p> <p>Type: String</p> <p>Default: None</p>	No
<i>BrowseNode</i>	<p>Browse nodes are positive integers that identify product categories, for example, Literature & Fiction: (17), Medicine: (13996), Mystery & Thrillers: (18), Nonfiction: (53), Outdoors & Nature: (290060).</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Positive integer.</p>	No
<i>Composer</i>	<p>Name of an composer associated with the item. You can enter all or part of the name.</p> <p>Type: String</p> <p>Default: None</p>	No

Name	Description	Required
<i>Condition</i>	<p>Use the <i>Condition</i> parameter to filter the offers returned in the product list by condition type. By default, <i>Condition</i> equals "New". If you do not get results, consider changing the value to "All". When the Availability parameter is set to "Available", the Condition parameter cannot be set to "New".</p> <p>ItemSearch returns up to ten search results at a time. When <i>Condition</i> equals "All", ItemSearch returns up to three offers per condition (if they exist), for example, three new, three used, three refurbished, and three collectible items. Or, for example, if there are no collectible or refurbished offers, ItemSearch returns three new and three used offers.</p> <p>Type: String</p> <p>Default: New</p> <p>Valid Values: Used Collectible Refurbished All</p>	No
<i>Conductor</i>	<p>Name of a conductor associated with the item. You can enter all or part of the name.</p> <p>Type: String</p> <p>Default: None</p>	No
<i>Director</i>	<p>Name of a director associated with the item. You can enter all or part of the name.</p> <p>Type: String</p> <p>Default: None</p>	No
<i>IncludeReviewsSummary</i>	<p>When set to <code>true</code>, returns the reviews summary within the Reviews iframe.</p> <p>Type: Boolean</p> <p>Default: True</p> <p>Valid Values: True False</p>	No

Name	Description	Required
<i>ItemPage</i>	<p>Retrieves a specific page of items from all of the items in a response. Up to ten items are returned on a page unless <i>Condition</i> equals "All". In that case, <i>ItemSearch</i> returns additional offers for those items, one offer per condition type (if an offer exists)—for example, one new, one used, one refurbished, and one collectible item. Or, for example, if there are no collectible or refurbished offers, <i>ItemSearch</i> returns one new and one used offer. If you do not include <i>ItemPage</i> in your request, the first page is returned. The total number of pages of items found is returned in the <i>TotalPages</i> response tag.</p> <p>Valid Values: 1 to 10 (1 to 5 when the search index = "All")</p> <p>Type: String</p> <p>Default: None</p>	No
<i>Keywords</i>	<p>A word or phrase associated with an item. The word or phrase can be in various product fields, including product title, author, artist, description, manufacturer, and so forth. When, for example, the search index equals "Music-Tracks", the <i>Keywords</i> parameter enables you to search by song title. If you enter a phrase, the spaces must be URL-encoded as %20.</p> <p>Type: String</p> <p>Default: None</p>	No
<i>Manufacturer</i>	<p>Name of a manufacturer associated with the item. You can enter all or part of the name.</p> <p>Type: String</p> <p>Default: None</p>	No
<i>MaximumPrice</i>	<p>Specifies the maximum price of the items in the response. Prices are in terms of the lowest currency denomination, for example, pennies. For example, 3241 represents \$32.41.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Positive integer</p>	No
<i>MerchantId</i>	<p>An optional parameter you can use to filter search results and offer listings to only include items sold by Amazon. By default, Product Advertising API returns items sold by various merchants including Amazon. Use the <i>Amazon</i> to limit the response to only items sold by Amazon.</p> <p>Type: String</p> <p>Valid Values: Amazon</p>	No

Product Advertising API Developer Guide
ItemSearch

Name	Description	Required
<i>MinimumPrice</i>	<p>Specifies the minimum price of the items to return. Prices are in terms of the lowest currency denomination, for example, pennies, for example, 3241 represents \$32.41.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Positive integer</p>	No
<i>MinPercentageOff</i>	<p>Specifies the minimum percentage off for the items to return.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Positive integer</p>	No
<i>Orchestra</i>	<p>Name of an orchestra associated with the item. You can enter all or part of the name.</p> <p>Type: String</p> <p>Default: None</p>	No
<i>Power</i>	<p>Performs a book search using a complex query string. Only works when the search index is set equal to "Books".</p> <p>Valid Values: See, Power Searches (p. 81) following this table.</p> <p>Type: String</p> <p>Default: None</p>	No
<i>Publisher</i>	<p>Name of a publisher associated with the item. You can enter all or part of the name.</p> <p>Type: String</p> <p>Default: None</p>	No
<i>RelatedItemPage</i>	<p>This optional parameter is only valid when the <i>RelatedItems</i> response group is used. Each <i>ItemLookup</i> request can return, at most, ten related items. The <i>RelatedItemPage</i> value specifies the set of ten related items to return. A value of 2, for example, returns the second set of ten related items.</p>	No
<i>RelationshipType</i>	<p>This parameter is required when the <i>RelatedItems</i> response group is used. The type of related item returned is specified by the <i>RelationshipType</i> parameter. Sample values include Episode, Season, and Tracks. A complete list of values follows this table.</p> <p>Constraint: Required when <i>RelatedItems</i> response group is used</p>	Conditional

Product Advertising API Developer Guide
ItemSearch

Name	Description	Required
<i>SearchIndex</i>	<p>The product category to search. Many <code>ItemSearch</code> parameters are valid with only specific values of <i>SearchIndex</i>.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: A search index, for example, Apparel, Beauty, Blended, Books, and so forth. For Blended searches, go to Blended Searches (p. 80). For a complete list of search indices, see Locale Reference (p. 328).</p>	Yes
<i>Sort</i>	<p>Means by which the items in the response are ordered.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Valid values vary significantly by search index. For a list of valid values, see Locale Reference (p. 328).</p>	No
<i>Title</i>	<p>The title associated with the item. You can enter all or part of the title. <i>Title</i> searches are a subset of <i>Keyword</i> searches. If a <i>Title</i> search yields insufficient results, consider using a <i>Keywords</i> search.</p> <p>Type: String</p> <p>Default: None</p>	No
<i>TruncateReviewsAt</i>	<p>By default, reviews are truncated to 1000 characters within the Reviews iframe. To specify a different length, enter the value. To return complete reviews, specify 0.</p> <p>Type: Integer</p> <p>Default: 1000</p> <p>Constraints: Must be a positive integer or 0 (returns entire review)</p>	No
<i>VariationPage</i>	<p>Retrieves a specific page of variations returned by <code>ItemSearch</code>. By default, <code>ItemSearch</code> returns all variations. Use <i>VariationPage</i> to return a subsection of the response. There are 10 variations per page. To examine offers 11 through 20, for example, set <i>VariationPage</i> to 2. The total number of pages is returned in the <i>TotalPages</i> element.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Positive integer</p>	No

Name	Description	Required
<i>ResponseGroup</i>	<p>Specifies the types of values to return. You can specify multiple response groups in one request by separating them with commas.</p> <p>Type: String</p> <p>Default: Small (p. 293)</p> <p>Valid Values: Accessories (p. 231) BrowseNodes (p. 236) EditorialReview (p. 245) ItemAttributes (p. 249) ItemIds (p. 253) Large (p. 254) Medium (p. 259) Offer-Full (p. 267) Offers (p. 271) OfferSummary (p. 275) Reviews (p. 283) RelatedItems (p. 279) SearchBins (p. 286) Similarities (p. 291) Small (p. 293) Tracks (p. 296) Variations (p. 297) VariationSummary (p. 307) </p>	No

ItemSearch also accepts the parameters that all operations can use. For more information, see [Common Request Parameters](#) (p. 228).

Movie Ratings Vary by Locale

Movie rating values captured in the *AudienceRating* parameter, vary by locale. The following table shows the valid values of *AudienceRating*.

Locale	AudienceRating Values
CA	G, PG, PG-13, R, NC-17, NR, Unrated, Family Viewing
DE	6, 12, 16
FR	PG, 12, 16, 18
US	G, PG, PG-13, R, NC-17, NR, Unrated

Response

Name	Description
<i>ASIN</i>	Amazon Standard Identification Number, which is an alphanumeric token assigned by Amazon to an item that uniquely identifies it.
<i>Item</i>	Container for item information, including ASIN and ItemAttributes.
<i>ItemAttributes</i>	Container for information about an item, including Manufacturer, ProductGroup, and Title.
<i>Manufacturer</i>	Item's manufacturer.
<i>MoreSearchResultsURL</i>	The URL where the complete search results are displayed. The URLs provided in the search results are the exact ones that you should use when you link back to Amazon.com. They are tagged with your Associate tag and contain other tracking information to increase your hourly request limit as the sales that you generate increase.
<i>ProductGroup</i>	Product category; similar to search index.

Name	Description
<i>Title</i>	Item's title.
<i>TotalPages</i>	Total number of pages in response. There are up to ten items per page.
<i>TotalResults</i>	Total number of items found.

For more information about the parent elements of these tags, see the appropriate response group in [Response Groups \(p. 230\)](#).

Examples

Use the search index, Toys, and the parameter, *Keywords*, to return information about all toy rockets sold in by Amazon.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate ID]&
Operation=ItemSearch&
Keywords=Rocket&
SearchIndex=Toys
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The response to this request is shown in [Sample Response \(p. 194\)](#).

Use a blended search to look through multiple search indices for items that have “Mustang” in their name or description. A blended search looks through multiple search indices at the same time. For more information, see [Blended Searches. \(p. 80\)](#)

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate ID]&
Operation=ItemSearch&
Keywords=Mustang&
SearchIndex=Blended
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

Use the *Availability* parameter to only return shirts that are available:

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate ID]&
Operation=ItemSearch&
Condition=All&
Availability=Available&
SearchIndex=Apparel&
Keywords=Shirt
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

Set the search index to MusicTracks and *Keywords* to the title of a song to find a song title.

Use the [BrowseNodes \(p. 236\)](#) response group to find the browse node of an item.

Use the [Variations \(p. 35\)](#) response group and the *BrowseNode* parameter to find all of the variations of a parent browse node.

Sample Response

The following XML is a snippet of the full response to the first sample request.

```
<TotalResults>372</TotalResults>
<TotalPages>38</TotalPages>
<Item>
  <ASIN>B00021HBN6</ASIN>
  <ItemAttributes>
    <Manufacturer>Radio Flyer</Manufacturer>
    <ProductGroup>Toy</ProductGroup>
    <Title>Radio Flyer Retro Rocket</Title>
  </ItemAttributes>
</Item>
<Item>
  <ASIN>B0007MZV3C</ASIN>
  <ItemAttributes>
    <Manufacturer>Razor USA LLC</Manufacturer>
    <ProductGroup>Toy</ProductGroup>
    <Title>Razor Dirt Rocket MX350 Bike</Title>
  </ItemAttributes>
</Item>
```

The *TotalResults* and *TotalPages* tags indicate the number of items found and the number of pages those items are on. Use *TotalPages* with any of the page parameters, such as *ReviewsPage*, to select the page of results to view. Typically, there are up to ten results on a page.

Related Operations

- [ItemLookup \(p. 197\)](#)

BrowseNodeLookup

Description

Given a browse node ID, `BrowseNodeLookup` returns the specified browse node's name, children, and ancestors. The names and browse node IDs of the children and ancestor browse nodes are also returned. `BrowseNodeLookup` enables you to traverse the browse node hierarchy to find a browse node.

As you traverse down the hierarchy, you refine your search and limit the number of items returned. For example, you might traverse the following hierarchy: DVD>Used DVDs>Kids and Family, to select out of all the DVDs offered by Amazon only those that are appropriate for family viewing. Returning the items associated with Kids and Family produces a much more targeted result than a search based at the level of Used DVDs.

Alternatively, by traversing up the browse node tree, you can determine the root category of an item. You might do that, for example, to return the top seller of the root product category using the *TopSeller* response group in an [ItemSearch \(p. 185\)](#) request.

You can use `BrowseNodeLookup` iteratively to navigate through the browse node hierarchy to reach the node that most appropriately suits your search. Then you can use the browse node ID in an [ItemSearch \(p. 185\)](#) request. This response would be far more targeted than, for example, searching through all of the browse nodes in a search index.

Availability

All locales

Request Parameters

Name	Description	Required
<code>BrowseNodeId</code>	A positive integer assigned by Amazon that uniquely identifies a product category. Type: String Default: None Valid Values:A positive integer. For a list of valid browse node IDs, see Locale Reference (p. 328) .	Yes
<code>ResponseGroup</code>	Specifies the types of values to return. You can specify multiple response groups in one request by separating them with commas. Default: BrowseNodeInfo (p. 234) Valid Values: MostGifted (p. 263) NewReleases (p. 265) MostWishedFor (p. 264) TopSellers (p. 294)	No

`BrowseNodeLookup` also accepts the parameters that all operations can use. For more information, see [Common Request Parameters \(p. 228\)](#).

Response

Name	Description
<code>Ancestors</code>	Container object for a parent browse node.
<code>BrowseNode</code>	Container object for all browse node data, including browse node ID, browse node name, browse node children and ancestors.
<code>BrowseNodeId</code>	A positive integer that uniquely identifies a product group, such as Literature & Fiction: (17), Medicine: (13996), and Mystery & Thrillers: (18).
<code>Children</code>	Container for one or more browse nodes, which are the children of the browse node submitted in the request.
<code>Name</code>	Name of the <code>BrowseNode</code> , for example, the name of <code>BrowseNode 17</code> is Literature & Fiction.

For more information about the parent elements of these tags, see the appropriate response group in [Response Groups \(p. 230\)](#).

Examples

Use BrowseNodeLookup iteratively to navigate through the hierarchy of browse nodes. In this way, customers can refine their searches, for example.

```
DVD>Actors & Actresses>Steve Martin  
DVD>Used DVDs>Kids & Family
```

The first hierarchy narrows the search down to DVDs in which Steve Martin plays a part. The second hierarchy narrows the list of DVDs down to those that are suitable for Kids and Family. This node, for example, might contain thirty such DVD titles.

The following request returns the name, parent, and children of the browse node for DVD comedies (163357)

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=BrowseNodeLookup&  
BrowseNodeId=163357  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

The response to this request is shown in [Sample Response \(p. 196\)](#).

Use the [NewReleases \(p. 265\)](#) response group to display the newly released items for a specified browse node:

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=BrowseNodeLookup&  
BrowseNodeId=163357&  
ResponseGroup=NewReleases  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response

The following XML is a snippet of the full response to the first sample request.

```
<BrowseNode>  
  <BrowseNodeId>163357</BrowseNodeId>  
  <Name>Comedy</Name>  
  <Children>  
    <BrowseNode>  
      <BrowseNodeId>599826</BrowseNodeId>  
      <Name>Boxed Sets</Name>  
    </BrowseNode>  
    <BrowseNode>  
      <BrowseNodeId>538712</BrowseNodeId>  
      <Name>African American Comedy</Name>
```

```
</BrowseNode>
<BrowseNode>
  <BrowseNodeId>163358</BrowseNodeId>
  <Name>Animation</Name>
</BrowseNode>
<BrowseNode>
  <BrowseNodeId>720556</BrowseNodeId>
  <Name>Black Comedy</Name>
</BrowseNode>
<BrowseNode>
  <BrowseNodeId>291102</BrowseNodeId>
  <Name>British</Name>
</BrowseNode>
</Children>
<Item>
  <Ancestors>
    <BrowseNode>
      <BrowseNodeId>549726</BrowseNodeId>
      <Name>Performing Arts</Name>
    <Ancestors>
      <BrowseNode>
        <BrowseNodeId>5</BrowseNodeId>
        <Name>Entertainment</Name>
      <Ancestors>
        <BrowseNode>
          <BrowseNodeId>1000</BrowseNodeId>
          <Name>Subjects</Name>
        <Ancestors>
          <BrowseNode>
            <BrowseNodeId>1000</BrowseNodeId>
            <Name>Books</Name>

```

This response shows that the browse node, Comedy, which is specified in the request, has five direct descendants (Boxed Sets, African American Comedy, Animation, Black Comedy, and British) and its ancestors are Books>Subjects>Entertainment>Performing>Comedy.

Related Operations

- [ItemSearch \(p. 185\)](#)
- [ItemLookup \(p. 197\)](#) using the [BrowseNodes \(p. 236\)](#) response group.

ItemLookup

Description

Given an Item identifier, the `ItemLookup` operation returns some or all of the item attributes, depending on the response group specified in the request. By default, `ItemLookup` returns an item's `ASIN`, `Manufacturer`, `ProductGroup`, and `Title` of the item.

`ItemLookup` supports many response groups. Response groups return product information, called item attributes. Item attributes include product reviews, variations, similar products, pricing, availability, images of products, accessories, and other information.

To look up more than one item at a time, separate the item identifiers by commas.

Availability

All locales, however, the parameter support varies by locale.

Request Parameters

Name	Description	Required
<i>Condition</i>	<p>Specifies an item's condition. If Condition is set to "All", a separate set of responses is returned for each valid value of Condition. The default value is "New" (not "All"). So, if your request does not return results, consider setting the value to "All". When the value is "New", the ItemLookup availability parameter cannot be set to "Available". Amazon only sells items that are "New".</p> <p>Type: String</p> <p>Default: New</p> <p>Valid Values: Used Collectible Refurbished, All</p>	No
<i>IdType</i>	<p>Type of item identifier used to look up an item. All <i>IdTypes</i> except ASINx require a <i>SearchIndex</i> to be specified.</p> <p>Type: String</p> <p>Default: ASIN</p> <p>Valid Values: SKU UPC EAN ISBN (US only, when search index is Books). UPC is not valid in the CA locale.</p>	No
<i>IncludeReviewsSummary</i>	<p>When set to true, returns the reviews summary within the Reviews iframe.</p> <p>Type: Boolean</p> <p>Default: True</p> <p>Valid Values: True False</p>	No
<i>ItemId</i>	<p>One or more (up to ten) positive integers that uniquely identify an item. The meaning of the number is specified by <i>IdType</i>. That is, if <i>IdType</i> is ASIN, the <i>ItemId</i> value is an ASIN. If <i>ItemId</i> is an ASIN, a search index cannot be specified in the request.</p> <p>Type: String</p> <p>Default: None</p> <p>Constraints: Must be a valid item ID. For more than one ID, use a comma-separated list of up to ten IDs.</p>	Yes

Name	Description	Required
<i>MerchantId</i>	<p>An optional parameter you can use to filter search results and offer listings to only include items sold by Amazon. By default, the API will return items sold by various merchants including Amazon. Enter <code>Amazon</code> if you only want to see items sold by Amazon in the response.</p> <p>Type: String</p> <p>Valid Values: <code>Amazon</code></p>	No
<i>RelatedItemPage</i>	<p>This optional parameter is only valid when the <code>RelatedItems</code> response group is used. Each <code>ItemLookup</code> request can return, at most, ten related items. The <code>RelatedItemPage</code> value specifies the set of ten related items to return. A value of 2, for example, returns the second set of ten related items</p>	No
<i>RelationshipType</i>	<p>This parameter is required when the <code>RelatedItems</code> response group is used. The type of related item returned is specified by the <code>RelationshipType</code> parameter. Sample values include Episode, Season, and Tracks. For a complete list of types, see Relationship Types (p. 137).</p> <p>Required when <code>RelatedItems</code> response group is used.</p>	Conditional
<i>SearchIndex</i>	<p>The product category to search.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: A search index, for example, Apparel, Beauty, Blended, Books, and so forth. For a complete of search indices, see Locale Reference (p. 328).</p> <p>Constraint: If <code>ItemId</code> is an ASIN, a search index cannot be specified in the request. Required for non-ASIN ItemIds.</p>	Conditional
<i>TruncateReviewsAt</i>	<p>By default, reviews are truncated to 1000 characters within the Reviews iframe. To specify a different length, enter the value. To return complete reviews, specify 0.</p> <p>Type: Integer</p> <p>Default: 1000</p> <p>Constraints: Must be a positive integer or 0 (returns entire review)</p>	No

Name	Description	Required
<i>VariationPage</i>	<p>Page number of variations returned by <code>ItemLookup</code>. By default, <code>ItemLookup</code> returns all variations. Use <code>VariationPage</code> to return a subsection of the response. There are 10 variations per page. To examine offers 11 through 20, for example, set <code>VariationPage</code> to 2.</p> <p>Type: String</p> <p>Default: All</p> <p>Valid Values: Integer between 1 and 150, inclusive</p>	No
<i>ResponseGroup</i>	<p>Specifies the types of values to return. You can specify multiple response groups in one request by separating them with commas.</p> <p>Type: String</p> <p>Default: Small (p. 293)</p> <p>Valid Values: Accessories (p. 231) BrowseNodes (p. 236) EditorialReview (p. 245) Images (p. 247) ItemAttributes (p. 249) ItemIds (p. 253) Large (p. 254) Medium (p. 259) OfferFull (p. 267) Offers (p. 271) PromotionSummary (p. 277) OfferSummary (p. 275) RelatedItems (p. 279) Reviews (p. 283) SalesRank (p. 285) Similarities (p. 291) Small (p. 293) Tracks (p. 296) VariationImages (p. 300) Variations (p. 297) (US only) VariationSummary (p. 307)</p>	No

`ItemLookup` also accepts the parameters that all operations can use. For more information, see [Common Request Parameters](#) (p. 228).

Response

Name	Description
<i>ASIN</i>	Amazon Standard Identification Number, which is an alphanumeric token assigned by Amazon to an item that uniquely identifies it.
<i>Item</i>	Container for information about the item, including <code>ASIN</code> , <code>Title</code> , <code>ProductGroup</code> , and <code>Manufacturer</code> .
<i>ItemAttributes</i>	Container for information about an item, including <code>Title</code> , <code>ProductGroup</code> , and <code>Manufacturer</code> .
<i>Items</i>	Container for one or more Item(s).
<i>Manufacturer</i>	Name of the company that manufactured the item.
<i>ProductGroup</i>	Category of the item, for example, "Book" and "DVD".
<i>Title</i>	Title of the item.

For more information about the parent elements of these tags, see the appropriate response group in [Response Groups](#) (p. 230).

Examples

The following request returns the information associated with *ItemId* B00008OE6I.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=B00008OE6I  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

The response to this request is shown in, [Sample Response \(p. 202\)](#).

The following request returns an offer for a refurbished item that is not sold by Amazon.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=B00008OE6I&  
IdType=ASIN&  
ResponseGroup=OfferFull&  
Condition=All&  
Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

In the following request, the *ItemId* is an SKU, which requires that you also specify the *IdType*.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=[SKU]&  
IdType=SKU  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

In the following request, the *ItemId* is a UPC, which requires that you also specify the *SearchIndex* and *ItemType*.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=[UPC]&  
SearchIndex=Books&  
&IdType=UPC  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

In the following request, the *ItemId* is an EAN, which requires that you also specify the *SearchIndex* and *ItemType*.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=[EAN]&  
SearchIndex=Electronics&  
IdType=EAN  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Use the [BrowseNodes \(p. 236\)](#) response group to find the browse node of an item.

Use the [Tracks \(p. 296\)](#) response group to find the track, title, and number for each track on each CD in the response.

Use the [Similarities \(p. 291\)](#) response group to find the ASIN and Title for similar products returned in the response.

Use the [Reviews \(p. 283\)](#) response group to find reviews written by customers about an item, and the total number of reviews for each item in the response.

Use the [OfferSummary \(p. 275\)](#) response group to find the number of offer listings and the lowest price for each of the offer listing condition classes, including New, Used, Collectible, and Refurbished.

Use the [Accessories \(p. 231\)](#) response group to find the a list of accessory product ASINs and Titles for each product in the response that has accessories.

The following requests an iframe that contains customer reviews for the specified item.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=0316067938&  
ResponseGroup=Reviews&  
TruncateReviewsAt="256"&  
IncludeReviewsSummary="False"&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response

The following code snippet is a response to the first request. It shows all of the item attributes that are returned by default.

```
<Items>  
  <Request>  
    <IsValid>True</IsValid>  
    <ItemLookupRequest>  
      <ItemId>B00008OE6I</ItemId>
```

```
</ItemLookupRequest>
</Request>
<Item>
  <ASIN>B00008OE6I</ASIN>
  <ItemAttributes>
    <Manufacturer>Canon</Manufacturer>
    <ProductGroup>Photography</ProductGroup>
    <Title>Canon PowerShot S400 4MP Digital Camera w/ 3x Optical Zoom</Title>
  </ItemAttributes>
</Item>
</Items>
```

The following code snippet is part of a response for an iframe that contains customer reviews.

```
<ItemLookupResponse>
<Items>
<Item>
  <ASIN>0316067938</ASIN>
  <CustomerReviews>
    <IFrameURL>
      http://www.amazon.com/reviews/iframe?akid=AKIAIOSFODNN7EXAMPLE&asin=0316067938&exp=2013-08-01T17%3A54%3A07Z&linkCode=xm2&summary=0&tag=ws&truncate=256&v=2&sig=rhBGpRiaQ3L78yqgE5y30u8bF6zXsK3HL%2F2KpNMJns0%3D
    </IFrameURL>
  </CustomerReviews>
</Item>
</Items>
</ItemLookupResponse>
```

Related Operations

- [ItemSearch \(p. 185\)](#)

SimilarityLookup

Description

The `SimilarityLookup` operation returns up to ten products per page that are similar to one or more items specified in the request. This operation is typically used to pique a customer's interest in buying something similar to what they've already ordered.

If you specify more than one item, `SimilarityLookup` returns the intersection of similar items each item would return separately. Alternatively, you can use the `SimilarityType` parameter to return the union of items that are similar to any of the specified items. A maximum of ten similar items are returned; the operation does not return additional pages of similar items. If there are more than ten similar items, running the same request can result in different answers because the ten that are included in the response are picked randomly. The results are picked randomly only when you specify multiple items and the results include more than ten similar items.

When you specify multiple items, it is possible for there to be no intersection of similar items. In this case, the operation returns the following error:

```
<Error>
<Code>AWS.ECommerceService.NoSimilarities</Code>
<Message>There are no similar items for this ASIN: B00006WREH.</Message>
</Error>
```

This result is very often the case if the items belong to different search indices. The error can occur, however, even when the items share the same search index.

Similarity is a measurement of similar items purchased, that is, customers who bought X also bought Y and Z. It is not a measure, for example, of items viewed, that is, customers who viewed X also viewed Y and Z.

Items returned can be filtered by:

- Condition—Describes the status of an item. Valid values are All, New (default), Used, Refurbished or Collectible. When the Availability parameter is set to "Available", the Condition parameter cannot be set to "New".

Examples

Return items that are similar to a list of items.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate ID]&
Operation=SimilarityLookup&ItemId=ASIN1,ASIN2,ASIN3
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

This request returns the intersection of the similarities for each ASIN. The response to this request is shown in [Sample Response \(p. 206\)](#).

Return up to ten items that are similar to any of the ASINs specified.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate ID]&
Operation=SimilarityLookup&ItemId=ASIN1,ASIN2,ASIN3&
SimilarityType=Random
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

This request returns the union of items that are similar to all of the ASINs specified. Only ten items can be returned and those are picked randomly from all of the similar items. Repeating the operation could produce different results.

Request Parameters

Name	Description	Required
<i>Condition</i>	<p>Specifies an item's condition. If Condition is set to "All", a separate set of responses is returned for each valid value of Condition.</p> <p>Type: String</p> <p>Default: New</p> <p>Valid Values: All Collectible Refurbished Used</p>	No
<i>ItemId</i>	<p>Specifies the item you want to look up. An ItemId is an alphanumeric identifier assigned to an item. You can specify up to ten ItemIds separated by commas.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: ASIN</p>	Yes
<i>MerchantId</i>	<p>An optional parameter that can be used to filter search results and offer listings to only include items sold by Amazon. By default, Product Advertising API returns items sold by various merchants including Amazon. Enter "Amazon" to return only items sold by Amazon.</p> <p>Type: String</p> <p>Valid Values: Amazon</p>	No
<i>SimilarityType</i>	<p>"Intersection" returns the intersection of items that are similar to all of the ASINs specified. "Random" returns the union of items that are similar to all of the ASINs specified. Only ten items are returned. So, if there are more than ten similar items found, a random selection from the group is returned. For this reason, running the same request multiple times can yield different results.</p> <p>Type: String</p> <p>Default: Intersection</p> <p>Valid Value: Random</p>	No

Name	Description	Required
<i>ResponseGroup</i>	<p>Specifies the types of values to return. You can specify multiple response groups in one request by separating them with commas.</p> <p>Type: String</p> <p>Default: Request (p. 281) Small (p. 293)</p> <p>Valid Values: Accessories (p. 231) BrowseNodes (p. 236) EditorialReview (p. 245) Images (p. 247) Large (p. 254) ItemAttributes (p. 249) ItemIds (p. 253) Medium (p. 259) Offers (p. 271) OfferSummary (p. 275) PromotionSummary (p. 277) Reviews (p. 283) SalesRank (p. 285) Similarities (p. 291) Small (p. 293) Tracks (p. 296) Variations (p. 297) VariationSummary (p. 307) </p>	No

SimilarityLookup also accepts the parameters that all operations can use. For more information, see [Common Request Parameters \(p. 228\)](#).

Sample Response

The following XML is a snippet of the full response to the first sample request.

```
<Item>
  <ASIN>B0009VX8XQ</ASIN>
  <ItemAttributes>
    <ProductGroup>Apparel</ProductGroup>
    <Title>Mark VII Men's Short Sleeve Herringbone Knit Golf Shirt</Title>
  </ItemAttributes>
</Item>
<Item>
  <ASIN>B0009VX8VI</ASIN>
  <ItemAttributes>
    <ProductGroup>Apparel</ProductGroup>
    <Title>Mark VII Men's Short Sleeve Knit Golf Shirt</Title>
  </ItemAttributes>
</Item>
```

This response shows that two items were similar to the ASINs submitted. The information included with each item is sufficient to display the item for sale.

Response

Name	Description
<i>ASIN</i>	An alphanumeric token that uniquely identifies an item for sale.
<i>Item</i>	Container for information about an item, including, ItemAttributes, and ASIN.
<i>ItemAttributes</i>	Container for information describing an item, including, ProductGroup and Title.

Name	Description
<i>ProductGroup</i>	Category to which an item belongs. Similar to Search Index.
<i>Title</i>	Name of the item for sale.

For more information about the parent elements of these tags, see the appropriate response group in [Response Groups \(p. 230\)](#).

Related Operations

[ItemSearch \(p. 185\)](#) using the [Similarities \(p. 291\)](#) response group

CartAdd

Description

The `CartAdd` operation enables you to add items to an existing remote shopping cart. `CartAdd` can only be used to place a new item in a shopping cart. It cannot be used to increase the quantity of an item already in the cart. If you would like to increase the quantity of an item that is already in the cart, you must use the `CartModify` operation.

You add an item to a cart by specifying the item's `OfferListingId`, or `ASIN` and `ListItemId`. Once in a cart, an item can only be identified by its `CartItemId`. That is, an item in a cart cannot be accessed by its `ASIN` or `OfferListingId`. `CartItemId` is returned by [CartCreate \(p. 214\)](#), [CartGet \(p. 220\)](#), and [CartAdd \(p. 207\)](#).

To add items to a cart, you must specify the cart using the `CartId` and `HMAC` values, which are returned by the `CartCreate` operation.

If the associated [CartCreate \(p. 214\)](#) request specified an `AssociateTag`, all `CartAdd` requests must also include a value for Associate Tag otherwise the request will fail.

Note

Some manufacturers have a minimum advertised price (MAP) that can be displayed on Amazon's retail web site. In these cases, when performing a Cart operation, the MAP is returned instead of the actual price. The only way to see the actual price is to add the item to a remote shopping cart and follow the `PurchaseURL`. The actual price will be the MAP or lower.

Availability

All locales.

Request Parameters

Name	Description	Required
<code>ASIN</code>	<p>Specifies ASIN of item to be added to the cart where N is a positive integer between 1 and 10, inclusive. Up to ten items can be added at a time. Using an item's <i>OfferListingId</i> is preferred instead of the item's ASIN.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Valid ASIN</p> <p>Constraint: Required if an <i>OfferListingId</i> is not specified.</p>	Conditional
<code>AssociateTag</code>	<p>Specifies the Associate who is making the request. An Associate tag, also known as an Associate ID, is an automatically generated unique identifier. When you log in to the Amazon Associates website for your locale, the page that you are directed to includes a message that says "Signed in as" followed by your Associate tag.</p> <p>The Associate tag that you used to create a cart must also be used on all other operations associated with the same cart.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Value: The same Associate tag that was used in the related <i>CartCreate</i> request.</p> <p>Constraint:</p>	Yes
<code>CartId</code>	<p>Alphanumeric token returned by <i>CartCreate</i> (p. 214) that identifies a cart.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Value returned by <i>CartCreate</i>.</p>	Yes
<code>HMAC</code>	<p>The Hash Message Authentication Code is an encrypted alphanumeric token that is used to authenticate requests.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Value is calculated using request parameters, their values, a cryptographic function, and the Secret Key, which acts as the "key" for the function.</p>	Yes
<code>Item</code>	<p>Container for <code>ASIN</code> or <code>OfferListingId</code>, and <code>Quantity</code>. This is a SOAP only parameter.</p> <p>Valid Values: An <code>ASIN</code> or an <code>OfferListingId</code> is required.</p>	Yes (SOAP only)
<code>Items</code>	<p>Container for one or more <code>Item</code>(s). This is a SOAP only parameter.</p> <p>Valid Values: Item</p>	Yes (SOAP only)

Name	Description	Required
<i>MergeCart</i>	This parameter is deprecated. When a customer purchases items in a remote shopping cart, the items are added to the customer's Amazon retail shopping cart.	Deprecated
<i>OfferListingId</i>	An offer listing ID is a token that uniquely identifies an item that is sold by any merchant, including Amazon. This parameter is preferred to using an ASIN to add an item to a cart. Type: String Default: None Valid Values: Valid offer listing ID Constraint: Required if ASIN is not offered .	Conditional
<i>Quantity</i>	Specifies number of items to be added to the cart where N is a positive integer. The Quantity value is matched to the Item.N.ASIN value by the index number, N. For example, Item.1.ASIN is associated with Item.1.Quantity. Type: String Default: None Valid Values: Positive integer between 1 and 999, inclusive. Conditional Required for REST	Conditional
<i>ResponseGroup</i>	Specifies the types of values to return. You can specify multiple response groups in one request by separating them with commas. Type: String Default: Cart (p. 238) Valid Values: CartSimilarities (p. 244) CartTopSellers (p. 242) NewReleases (p. 265)	No

CartAdd also accepts the parameters that all operations can use. For more information, see [Common Request Parameters \(p. 228\)](#).

Response

Name	Description
<i>Amount</i>	Price of the item in terms of the lowest currency denomination, for example, pennies.
<i>ASIN</i>	An alphanumeric token that uniquely identifies an item sold by Amazon.
<i>CartAddRequest</i>	Container for <i>CartId</i> , <i>HMAC</i> , and <i>Items</i> .
<i>CartId</i>	Alphanumeric token returned by <i>CartCreate</i> that identifies a cart.
<i>CurrencyCode</i>	Format for the display of the money
<i>FormattedPrice</i>	The price to display on the web site.

Name	Description
<i>HMAC</i>	Hash Message Authentication Code returned by <code>CartCreate</code> that identifies a cart. This is an encrypted alphanumeric token that is used to authenticate cart operations.
<i>Item</i>	Container for <code>ASIN</code> or <code>OfferListingId</code> , and <code>Quantity</code> .
<i>Items</i>	Container for one or more item(s).
<i>PurchaseURL</i>	URL that customers should use to purchase the items in the cart. It includes the Associate's ID. It is important that they use this URL otherwise the Associate will not get credit for the purchase.
<i>Quantity</i>	Number of items added to cart.
<i>Subtotal</i>	Container for Amount, CurrencyCode, and FormattedPrice

For more information about the parent elements of these tags, see the appropriate response group in [Response Groups \(p. 230\)](#).

Examples

Add a single item to the cart:

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate ID]&
CartId=[CartId]&
HMAC=[HMAC]&
Operation=CartAdd&
Item.1.OfferListingId=[Offer Listing ID]&
Item.1.Quantity=1
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

The response to this request is shown in, [Sample Response \(p. 211\)](#).

Add multiple items to the cart.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate ID]&
CartId=[CartId]&
HMAC=[HMAC]&
Operation=CartAdd&
Item.1.OfferListingId=[Offer Listing ID]&
Item.1.Quantity=1&
Item.2.OfferListingId=[Offer Listing ID]&
Item.2.Quantity=3&
Item.3.OfferListingId=[Offer Listing ID]&
Item.3.Quantity=1
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

This request adds three different items to a cart. A cart can carry up to 50 different items and each item can have a maximum *Quantity* value of 999.

Add to the cart an item from a list.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
CartId=[CartId]&  
HMAC=[HMAC]&  
Operation=CartAdd&  
Item.1.ASIN=[ASIN]&  
Item.1.Quantity=3&  
Item.1.ListItemId=[List item ID]  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

This request adds three items from a list.

Sample Response

The following XML is a snippet of the full response to the first sample request.

```
<CartAddRequest>  
  <CartId>102-6666665-5792105</CartId>  
  <HMAC>oRqNBTpiRBugxEKEJ+FyUZkVbEg=</HMAC>  
  <Items>  
    <Item>  
      <ASIN>1400042127</ASIN>  
      <Quantity>1</Quantity>  
    </Item>  
  </Items>  
</CartAddRequest>  
<Request>  
  <CartId>102-6666665-5792105</CartId>  
  <HMAC>oRqNBTpiRBugxEKEJ+FyUZkVbEg=</HMAC>  
  <URLEncodedHMAC>oRqNBTpiRBugxEKEJ%2BFyUZkVbEg=</URLEncodedHMAC>  
<PurchaseURL>https://www.amazon.com/gp/cart/aws-merge.html?cart-id=102-5929035-  
5792105%26associate-id=ws%26hmac=oRqNBTpiRBugxEKEJ%2BFyUZkVbEg=%26AWSAccessKey  
Id=[AWS Access Key ID]</PurchaseURL>  
  <SubTotal>  
    <Amount>3432</Amount>  
    <CurrencyCode>USD</CurrencyCode>  
    <FormattedPrice>$34.32</FormattedPrice>  
  </SubTotal>
```

Related Operations

- [CartCreate \(p. 214\)](#)
- [CartGet \(p. 220\)](#)
- [CartModify \(p. 224\)](#)
- [CartClear \(p. 212\)](#)

CartClear

Description

The `CartClear` operation enables you to remove all of the items in a remote shopping cart, including `SavedForLater` items. To remove only some of the items in a cart or to reduce the quantity of one or more items, use [CartModify \(p. 224\)](#).

To delete all of the items from a remote shopping cart, you must specify the cart using the `CartId` and `HMAC` values, which are returned by the `CartCreate` operation. A value similar to the `HMAC`, `URLEncodedHMAC`, is also returned. This value is the URL encoded version of the `HMAC`. This encoding is necessary because some characters, such as + and /, cannot be included in a URL. Rather than encoding the `HMAC` yourself, use the `URLEncodedHMAC` value for the `HMAC` parameter.

`CartClear` does not work after the customer has used the `PurchaseURL` to either purchase the items or merge them with the items in their Amazon cart.

Carts exist even though they have been emptied. The lifespan of a cart is 7 days since the last time it was acted upon. For example, if a cart created 6 days ago is modified, the cart lifespan is reset to 7 days.

Availability

All locales.

Request Parameters

Name	Description	Required
<code>AssociateTag</code>	<p>Specifies the Associate who is making the request. An Associate tag, also known as an Associate ID, is an automatically generated unique identifier. When you log in to the Amazon Associates website for your locale, the page that you are directed to includes a message that says "Signed in as" followed by your Associate tag.</p> <p>The Associate tag that you used to create a cart must also be used on all other operations associated with the same cart.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Value: The same Associate tag that was used in the related <code>CartCreate</code> request.</p> <p>Constraint:</p>	Yes
<code>CartId</code>	<p>Alphanumeric token returned by <code>CartCreate</code> that identifies a cart.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Value returned by <code>CartCreate</code>.</p>	Yes

Name	Description	Required
<i>HMAC</i>	<p>The Hash Message Authentication Code is an encrypted alphanumeric token that is used to authenticate requests.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Value is calculated using request parameters, their values, a cryptographic function, and the Secret Key, which acts as the "key" for the function.</p>	Yes
<i>MergeCart</i>	This parameter is deprecated. When a customer purchases items in a remote shopping cart, the items are added to the customer's Amazon retail shopping cart.	Deprecated
<i>ResponseGroup</i>	<p>Specifies the types of values to return. You can specify multiple response groups in one request by separating them with commas.</p> <p>Type: String</p> <p>Default: Cart (p. 238)</p>	No

CartClear also accepts the parameters that all operations can use. For more information, see [Common Request Parameters](#) (p. 228).

Response

Name	Description
<i>Cart</i>	Container for remote shopping cart related response elements.
<i>CartClearRequest</i>	Container for <i>CartClear</i> and <i>HMAC</i> .
<i>CartId</i>	Alphanumeric token returned by <i>CartCreate</i> that uniquely identifies a cart.
<i>HMAC</i>	Hash Message Authentication Code returned by <i>CartCreate</i> that identifies a cart. This is an encrypted alphanumeric token that is used to authenticate cart operations.
<i>URLEncodedHMAC</i>	URL encoded version of the <i>HMAC</i> . This encoding is necessary because some characters, such as + and /, cannot be included in a URL.

For more information about the parent elements of these tags, see the appropriate response group in [Response Groups](#) (p. 230).

Examples

Remove all items from a specified cart:

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
```

```
AssociateTag=[Associate Tag]&
Operation=CartClear&
CartId=[Cart ID]&
HMAC=[URL-encoded HMAC]
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

This operation removes all of the items in the specified cart. Because the *HMAC* includes the *AssociateTag*, you must supply it in this request.

The response to this request is shown in [Sample Response \(p. 214\)](#).

Sample Response

The following XML is a snippet of the full response to the first sample request.

```
<Cart>
  <Request>
    <IsValid>True</IsValid>
    <CartClearRequest>
      <CartId>102-2689399-8023324</CartId>
      <HMAC>iLYO/W0hft0Te4uXAbMiBhs36T8=</HMAC>
    </CartClearRequest>
  </Request>
  <CartId>102-2689399-8023324</CartId>
  <HMAC>iLYO/W0hft0Te4uXAbMiBhs36T8=</HMAC>
  <URLEncodedHMAC>iLYO/W0hft0Te4uXAbMiBhs36T8=</URLEncodedHMAC>
</Cart>
```

Related Operations

- [CartAdd \(p. 207\)](#)
- [CartCreate \(p. 214\)](#)
- [CartGet \(p. 220\)](#)
- [CartModify \(p. 224\)](#)

CartCreate

Description

The *CartCreate* operation enables you to create a remote shopping cart. A shopping cart is the metaphor used by most e-commerce solutions. It is a temporary data storage structure that resides on Amazon servers. The structure contains the items a customer wants to buy. In Product Advertising API, the shopping cart is considered remote because it is hosted by Amazon servers. In this way, the cart is remote to the vendor's web site where the customer views and selects the items they want to purchase.

Once you add an item to a cart by specifying the item's ASIN or *OfferListing* ID, the item is assigned a *CartItemID* and is accessible only by that value. That is, in subsequent requests, an item in a cart cannot be accessed by its ASIN or *OfferListingID*. *CartItemID* is returned by [CartCreate \(p. 214\)](#), [CartGet \(p. 220\)](#), and [CartAdd \(p. 207\)](#).

Because the contents of a cart can change for different reasons, such as item availability, you should not keep a copy of a cart locally. Instead, use the other cart operations to modify the cart contents. For example, to retrieve contents of the cart, which are represented by *CartItemIds*, use [CartGet \(p. 220\)](#).

Available products are added as cart items. Unavailable items, for example, items out of stock, discontinued, or future releases, are added as *SaveForLaterItems*. No error is generated. The Amazon database changes regularly. You may find a product with an offer listing ID but by the time the item is added to the cart the product is no longer available. The checkout page in the Order Pipeline clearly lists items that are available and those that are *SaveForLaterItems*.

It is impossible to create an empty shopping cart. You have to add at least one item to a shopping cart using a single *CartCreate* request. You can add specific quantities (up to 999) of each item.

CartCreate can be used only once in the life cycle of a cart. To modify the contents of the cart, use one of the other cart operations.

Carts cannot be deleted. They expire automatically after being unused for 7 days. The lifespan of a cart restarts, however, every time a cart is modified. In this way, a cart can last for more than 7 days. If, for example, on day 6, the customer modifies a cart, the 7 day countdown starts over.

Specifying the Items

In one *CartCreate* request, you can add up to ten items; the quantity of each item is set separately, as follows:

```
Item.1.OfferListingId=[Offer Listing ID]  
Item.1.Quantity=3  
Item.2.OfferListingId=[Offer Listing ID]  
Item.2.Quantity=1
```

This code adds two items identified by their *OfferListingIds* and the quantity of each item is set to 3 and 1, respectively.

You can use any combination of *ASINS* or *OfferListingIds* in a single request to specify the items to add to a cart. You can only use one type of item identifier to add an item to a cart. You cannot, for example, use an *ASIN* and an *OfferListingId* in the same request to add the same item to a cart.

For more information the formatting used to add items to a cart, see [CartAdd \(p. 207\)](#).

Availability

All locales.

Request Parameters

Name	Description	Required
<i>ASIN</i>	An alphanumeric token that uniquely identifies an item. Using <i>OfferListingId</i> is preferred instead of <i>ASIN</i> when adding items to a cart. Valid Values: <i>ASIN</i> Constraint: Required if <i>OfferListingId</i> is not used	Conditional

Name	Description	Required
<i>AssociateTag</i>	<p>Specifies the Associate who is making the request. An Associate tag, also known as an Associate ID, is an automatically generated unique identifier. When you log in to the Amazon Associates website for your locale, the page that you are directed to includes a message that says "Signed in as" followed by your Associate tag.</p> <p>The Associate tag that you use to create a cart must also be used on all other operations associated with the same cart.</p> <p>Type: String Default: None Valid Values: An Associate Tag.</p>	Yes
<i>Item</i>	<p>For REST, a prefix for <i>ASIN</i> and quantity, both of which are used to specify the item to add to the cart, for example, item.1.<i>ASIN</i>=1234abcd, item.1.quantity=2.</p> <p>Valid Values: Does not take a value.</p> <p>Type: String Default: None</p>	Yes
<i>Items</i>	Container for one or more Item objects. This parameter is required for SOAP only.	Yes
<i>MergeCart</i>	This parameter is deprecated. When a customer purchases items in a remote shopping cart, the items are added to the customer's Amazon retail shopping cart. Type: String	Deprecated
<i>OfferListingId</i>	<p>An offer listing ID is an alphanumeric token that uniquely identifies an item. Use the OfferListingId instead of an item's ASIN to add the item to the cart.</p> <p>Type: String Default: None Valid Values: An Offer Listing ID Constraint: Required if <i>ASIN</i> is not used.</p>	Conditional
<i>Quantity</i>	<p>The suffix used with Item to specify the number of items to be added to the cart, for example, Item.1.<i>ASIN</i>=0976925524 Item.1.Quantity=2.</p> <p>See <i>Item</i>.</p> <p>Type: String Default: None Valid Values: Positive integer</p>	No

Name	Description	Required
<i>ResponseGroup</i>	<p>Specifies the types of values to return. You can specify multiple response groups in one request by separating them with commas.</p> <p>Type: String</p> <p>Default: Cart (p. 238)</p> <p>Valid Values: CartSimilarities (p. 244) CartTopSellers (p. 242) CartNewReleases (p. 241)</p>	No

`CartCreate` also accepts the parameters that all operations can use. For more information, see [Common Request Parameters](#) (p. 228).

Response

Name	Description
<i>Amount</i>	Price of the item in terms of the lowest currency denomination, for example, pennies.
<i>ASIN</i>	An alphanumeric token that uniquely identifies an item.
<i>Cart</i>	Container for all of the elements in the cart.
<i>CartCreateRequest</i>	Container for all of the items requested.
<i>CartId</i>	An alphanumeric token that uniquely identifies a cart. This value must be included in all future cart operations related to this cart.
<i>CartItems</i>	Container for purchase related information about each item in the cart
<i>CurrencyCode</i>	Locale of the sale that specifies the formatting of the price.
<i>FormattedPrice</i>	The price of the item as it should appear to the customer.
<i>HMAC</i>	An encoded value used to authenticate a request. This value must be included in all future cart operations related to this cart.
<i>ItemTotal</i>	Amount due for any number one kind of item.
<i>PurchaseURL</i>	The URL the customer should use to purchase the item. The URL includes the <i>AssociateTag</i> so that the Associate gets credit for the purchase.
<i>Quantity</i>	Number of individual items ordered.
<i>SavedForLaterItems</i>	Container object for items added to the cart that are currently unavailable.
<i>SubTotal</i>	Pre-tax and shipping subtotal of all items in the cart.
<i>URLEncodedHMAC</i>	A URL encoded version of the HMAC. The HMAC must be put in a URL and so it must be URL compliant. Use this value instead of the HMAC in all future requests related to this cart.

For more information about the parent elements of these tags, see the appropriate response group in [Response Groups \(p. 230\)](#).

Examples

Create a remote shopping cart and add multiple items of varying quantities to it.

```
http:// webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=CartCreate&  
Item.1.ASIN=[ASIN]&  
Item.1.Quantity=2&  
Item.2.ASIN=[ASIN]&  
Item.2.Quantity=7&  
Item.3.ASIN=[ASIN]&  
Item.3.Quantity=5  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

This request creates a remote shopping cart and adds three different items, 2 of the first item, 7 of the second item, and 5 of the third item. The response to this request is shown in [Sample Response \(p. 218\)](#).

Create a shopping cart by adding items not necessarily sold by Amazon.

```
http:// webservices.amazon.com/onca/xml  
?Service=AWSECommerceService&  
AWSAccessKeyId=[Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=CartCreate&  
Item.1.OfferListingId=[Offer Listing ID]&  
Item.1.Quantity=2&  
Item.2.OfferListingId=[Offer Listing ID]&  
Item.2.Quantity=7&  
Item.3.OfferListingId=[Offer Listing ID]&  
Item.3.Quantity=5  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

This request creates a remote shopping cart and adds three different items that are not sold by Amazon: 2 of the first item, 7 of the second item, and 5 of the third item. OfferListingId returns items for sale by all sellers and merchants, including Amazon.

Sample Response

The following XML is a snippet of the full response to the first sample request.

```
<Cart>  
  <Request>  
    <IsValid>True</IsValid>  
    <CartCreateRequest>  
      <Items>  
        <Item>  
          <ASIN>B000062TU1</ASIN>  
          <Quantity>2</Quantity>
```

```

        </Item>
    </Items>
</CartCreateRequest>
</Request>
<CartId>102-5014548-4857758</CartId>
<HMAC>O2p9hhZwJShnp6ZDWvZD06FhpAI=</HMAC>
<URLEncodedHMAC>O2p9hhZwJShnp6ZDWvZD06FhpAI=</URLEncodedHMAC>

<PurchaseURL>https://www.amazon.com/gp/cart/aws-merge.html?cart-id=102-5014548-4857758%26associate-id=ws%26mac=O2p9hhZwJShnp6ZDWvZD06FhpAI=%26AWSAccessKeyId=1VMEXAMPLE9C02</PurchaseURL>
<SubTotal>
    <Amount>1994</Amount>
    <CurrencyCode>USD</CurrencyCode>
    <FormattedPrice>$19.94</FormattedPrice>
</SubTotal>
<CartItems>
    <SubTotal>
        <Amount>1994</Amount>
        <CurrencyCode>USD</CurrencyCode>
        <FormattedPrice>$19.94</FormattedPrice>
    </SubTotal>
    <CartItem>
        <CartItemId>U31XY1DHZEGCTB</CartItemId>
        <ASIN>B000062TU1</ASIN>
        <Quantity>2</Quantity>
        <Title>Harry Potter and the Sorcerer's Stone (Full Screen Edition) (Harry Potter 1)</Title>
        <ProductGroup>DVD</ProductGroup>
        <Price>
            <Amount>997</Amount>
            <CurrencyCode>USD</CurrencyCode>
            <FormattedPrice>$9.97</FormattedPrice>
        </Price>
        <ItemTotal>
            <Amount>1994</Amount>
            <CurrencyCode>USD</CurrencyCode>
            <FormattedPrice>$19.94</FormattedPrice>
        </ItemTotal>
    </CartItem>
</CartItems>
</Cart>

```

Sample SavedForLaterItem XML Snippet

The following snippet from the XML response shows the addition of an item to a cart that is currently unavailable. On the retail web site, these items are displayed as "Saved For Later".

```

<SavedForLaterItems>
    <SubTotal>
        <Amount>1288</Amount>
        <CurrencyCode>USD</CurrencyCode>
        <FormattedPrice>$12.88</FormattedPrice>
    </SubTotal>
    <SavedForLaterItem>
        <CartItemId>ULI7S9IYFJHX0</CartItemId>
        <ASIN>B0009GZV4A</ASIN>

```

```
<Quantity>2</Quantity>
<Title>Mark VII Men's Short Sleeve Golf Shirts with Tri -Colored Stripe Trim</Title>
<ProductGroup>Apparel</ProductGroup>
<Price>
  <Amount>644</Amount>
  <CurrencyCode>USD</CurrencyCode>
  <FormattedPrice>$6.44</FormattedPrice>
</Price>
<ItemTotal>
  <Amount>1288</Amount>
  <CurrencyCode>USD</CurrencyCode>
  <FormattedPrice>$12.88</FormattedPrice>
</ItemTotal>
</SavedForLaterItem>
</SavedForLaterItems>
```

Related Operations

- [CartAdd \(p. 207\)](#)
- [CartAdd \(p. 207\)](#)
- [CartGet \(p. 220\)](#)
- [CartModify \(p. 224\)](#)

CartGet

Description

The `CartGet` operation enables you to retrieve the IDs, quantities, and prices of all of the items, including `SavedForLater` items in a remote shopping cart.

Because the contents of a cart can change for different reasons, such as availability, you should not keep a copy of a cart locally. Instead, use `CartGet` to retrieve the items in a remote shopping cart.

To retrieve the items in a cart, you must specify the cart using the `CartId` and `HMAC` values, which are returned in the `CartCreate` operation. A value similar to `HMAC`, `URLEncodedHMAC`, is also returned. This value is the URL encoded version of the `HMAC`. This encoding is necessary because some characters, such as + and /, cannot be included in a URL. Rather than encoding the `HMAC` yourself, use the `URLEncodedHMAC` value for the `HMAC` parameter.

`CartGet` does not work after the customer has used the `PurchaseURL` to either purchase the items or merge them with the items in their Amazon cart.

All `CartGet` requests must also include a value for `AssociateTag`. Otherwise, the request will fail.

Availability

All locales.

Request Parameters

Name	Description	Required
<i>AssociateTag</i>	<p>Specifies the Associate who is making the request. An Associate tag, also known as an Associate ID, is an automatically generated unique identifier. When you log in to the Amazon Associates website for your locale, the page that you are directed to includes a message that says "Signed in as" followed by your Associate tag.</p> <p>The Associate tag that you used to create a cart must also be used on all other operations associated with the same cart.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Value: The same Associate tag that was used in the related CartCreate request.</p> <p>Constraint:</p>	Yes
<i>CartId</i>	<p>Alphanumeric token returned by CartCreate that identifies a cart.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Value returned by CartCreate</p>	Yes
<i>CartItemID</i>	<p>Alphanumeric token that uniquely identifies an item in a cart. Once an item, specified by an ASIN or OfferListingId, has been added to a cart, you must use the <i>CartItemID</i> to refer to it. The other identifiers will not work.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Value returned by CartCreate</p>	Yes
<i>HMAC</i>	<p>The Hash Message Authentication Code is an encrypted alphanumeric token that is used to authenticate requests.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Value is calculated using request parameters, their values, a cryptographic function, and the Secret Key, which acts as the "key" for the function.</p>	Yes
<i>MergeCart</i>	This parameter is deprecated. When a customer purchases items in a remote shopping cart, the items are added to the customer's Amazon retail shopping cart.	Deprecated

Name	Description	Required
<i>ResponseGroup</i>	<p>Specifies the types of values to return. You can specify multiple response groups in one request by separating them with commas.</p> <p>Type: String</p> <p>Default: Cart (p. 238)</p> <p>Valid Values: CartSimilarities (p. 244) CartTopSellers (p. 242) CartNewReleases (p. 241)</p>	No

`CartGet` also accepts the parameters that all operations can use. For more information, see [Common Request Parameters](#) (p. 228).

Response

Name	Description
<i>Amount</i>	Price of the item in terms of the lowest currency denomination, for example, pennies.
<i>CartId</i>	Alphanumeric token returned by <code>CartCreate</code> that identifies a cart.
<i>CartItem</i>	Container for <i>CartItemId</i> , ASIN, Quantity, Title, ProductGroup, Price, <i>ItemTotal</i>
<i>CartItems</i>	Container for Subtotal and <i>CartItem</i>
<i>CurrencyCode</i>	Format for the display of the money
<i>FormattedPrice</i>	The price to display on the web site.
<i>HMAC</i>	Hash Message Authentication Code returned by <code>CartCreate</code> that identifies a cart. This is an encrypted alphanumeric token that is used to authenticate cart operations.
<i>ItemTotal</i>	Container for <i>Amount</i> , <i>CurrencyCode</i> , <i>FormattedPrice</i> . Cost of the item multiplied by the quantity ordered.
<i>Price</i>	Container for <i>Amount</i> , <i>CurrencyCode</i> , <i>FormattedPrice</i> . Price of a single item.
<i>PurchaseURL</i>	URL that customers should use to purchase the items in their cart. The URL includes the Associate's ID, called the Associate Tag. It is important to use the <code>PurchaseURL</code> otherwise the Associate will not get credit for the customer's purchase.
<i>Subtotal</i>	Container for <i>Amount</i> , <i>CurrencyCode</i> , and <i>FormattedPrice</i> . Subtotal is the total price of all of the items in a cart but not including tax or shipping.
<i>URLEncodedHMAC</i>	A URL encoded version of the <i>HMAC</i> . This encoding is necessary because some characters, such as + and /, cannot be included in a URL. The value of this parameter is a convenience function. Otherwise, you would have to URL encode the HMAC yourself.

For more information about the parent elements of these tags, see the appropriate response group in [Response Groups \(p. 230\)](#).

Examples

Retrieve all of the items in a specified cart:

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate Tag]&  
Operation=CartGet&  
CartId=[Cart ID]&  
HMAC=[URL-encoded HMAC]
```

This operation retrieves all of the items in the specified remote shopping cart. Because the *HMAC* includes the *AssociateTag*, you must supply it in this request.

The response to this request is shown in [Sample Response \(p. 223\)](#).

Sample Response

The following XML is a snippet of the full response to the first sample request.

```
<CartId>002-9918938-1696046</CartId>  
<HMAC>b0ogzvivVYLNjSZ9WwoBRFesFYU=</HMAC>  
<URLEncodedHMAC>b0ogzvivVYLNjSZ9WwoBRFesFYU=</URLEncodedHMAC>  
<PurchaseURL>https://www.amazon.com/gp/cart/aws-merge.html?cart-id=002-9918938-  
1696046%26associate-id=ws%26hmac=b0ogzvivVYLNjSZ9WwoBRFesFYU=%26AWSAccessKey  
Id=1V293857EXAMPLEFW9C02</PurchaseURL>  
<SubTotal>  
  <Amount>1994</Amount>  
  <CurrencyCode>USD</CurrencyCode>  
  <FormattedPrice>$19.94</FormattedPrice>  
</SubTotal>  
<CartItems>  
  <SubTotal>  
    <Amount>1994</Amount>  
    <CurrencyCode>USD</CurrencyCode>  
    <FormattedPrice>$19.94</FormattedPrice>  
</SubTotal>  
  <CartItem>  
    <CartItemID>U3KYV0C66V3PAA</CartItemID>  
    <ASIN>B000062TU1</ASIN>  
    <Quantity>2</Quantity>  
    <Title>Harry Potter and the Sorcerer's Stone (Full Screen Edition) (Harry  
Potter 1)</Title>  
    <ProductGroup>DVD</ProductGroup>  
    <Price>  
      <Amount>997</Amount>  
      <CurrencyCode>USD</CurrencyCode>  
      <FormattedPrice>$9.97</FormattedPrice>  
    </Price>  
    <ItemTotal>  
      <Amount>1994</Amount>  
      <CurrencyCode>USD</CurrencyCode>
```

```
<FormattedPrice>$19.94</FormattedPrice>
</ItemTotal>
</CartItem>
</CartItems>
</Cart>
```

This response shows all of the items in the specified cart.

Related Operations

- [CartAdd \(p. 207\)](#)
- [CartClear \(p. 212\)](#)
- [CartCreate \(p. 214\)](#)
- [CartModify \(p. 224\)](#)

CartModify

Description

The `CartModify` operation enables you to change the quantity of items that are already in a remote shopping cart and move items from the active area of a cart to the `SaveForLater` area or the reverse.

To modify the number of items in a cart, you must specify the cart using the `CartId` and `HMAC` values that are returned in the [CartCreate \(p. 214\)](#) operation. A value similar to `HMAC`, `URLEncodedHMAC`, is also returned. This value is the URL encoded version of the `HMAC`. This encoding is necessary because some characters, such as + and /, cannot be included in a URL. Rather than encoding the `HMAC` yourself, use the `URLEncodedHMAC` value for the `HMAC` parameter.

You can use `CartModify` to modify the number of items in a remote shopping cart by setting the value of the `Quantity` parameter appropriately. You can eliminate an item from a cart by setting the value of the `Quantity` parameter to zero. Or, you can double the number of a particular item in the cart by doubling its `Quantity`. You cannot, however, use `CartModify` to add new items to a cart.

All `CartModify` requests must also include the value for `AssociateTag` that was used in the associated [CartCreate \(p. 214\)](#) request; otherwise, the request will fail.

Availability

All locales.

Request Parameters

Name	Description	Required
<i>Action</i>	<p>The Action parameter is a child element of the Item parameter in both SOAP requests. Use the Action parameter to change cart items to move items to the Saved-For-Later area, or change Saved-For-Later (SaveForLater) items to the active cart area (MoveToCart).</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: MoveToCart SaveForLater</p>	No
<i>AssociateTag</i>	<p>Specifies the Associate who is making the request. An Associate tag, also known as an Associate ID, is an automatically generated unique identifier. When you log in to the Amazon Associates website for your locale, the page that you are directed to includes a message that says "Signed in as" followed by your Associate tag.</p> <p>The Associate tag that you used to create a cart must also be used on all other operations associated with the same cart.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Value: The same Associate tag that was used in the related CartCreate request.</p> <p>Constraint:</p>	Yes
<i>CartId</i>	<p>Alphanumeric token returned by CartCreate (p. 214) that identifies a cart.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Value returned by CartCreate (p. 214).</p>	Yes
<i>CartItemNId</i>	<p>Specifies an item to be modified in the cart where N is a positive integer between 1 and 10, inclusive. Up to ten items can be modified at a time. <i>CartItemNId</i> is neither an ASIN nor an OfferListingId. It is, instead, an alphanumeric token returned by CartCreate and CartAdd. This parameter is used in conjunction with <i>Item.N.Quantity</i> to modify the number of items in a cart. See Item, that follows, for more information.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Value returned by CartCreate (p. 214) or CartAdd (p. 207).</p>	Yes

Name	Description	Required
<i>HMAC</i>	<p>The Hash Message Authentication Code is an encrypted alphanumeric token that is used to authenticate requests.</p> <p>Type: String</p> <p>Default: None</p> <p>Valid Values: Value is calculated using request parameters, their values, a cryptographic function, and the Secret Key, which acts as the "key" for the function. Value returned by CartCreate (p. 214).</p>	Yes
<i>Item</i>	Specifies the item to modify. Typical construction is Item.1.CartItemId=[ID]; Item.1.Quantity=[number].	Yes
<i>Items</i>	<p>Container for one or more Item objects. This is a SOAP only parameter.</p> <p>Valid Value: Item</p> <p>Constraint: Required and valid only for SOAP requests.</p>	Conditional
<i>MergeCart</i>	This parameter is deprecated. When a customer purchases items in a remote shopping cart, the items are added to the customer's Amazon retail shopping cart.	Deprecated
<i>Quantity</i>	<p>Specifies the revised number of items that the customer would like in the cart where N is a positive integer. This parameter is used in conjunction with <i>Item.N.CartItemId</i> to modify the number of items in a cart. The Quantity value is matched to the Item.N.CartItemId value by the index number, N. For example, Item.1.CartItemId is associated with Item.1.Quantity.</p> <p>Type: Positive integer</p> <p>Type: String</p> <p>Default: None</p> <p>Constraint: N is a positive integer between 0 and 999, inclusive. Required with REST queries.</p>	Conditional
<i>ResponseGroup</i>	<p>Specifies the types of values to return. You can specify multiple response groups in one request by separating them with commas.</p> <p>Type: String</p> <p>Default: Cart (p. 238)</p> <p>Valid Values:CartSimilarities (p. 244) CartTopSellers (p. 242) CartNewReleases (p. 241)</p>	No

CartModify also accepts the parameters that all operations can use. For more information, see [Common Request Parameters \(p. 228\)](#).

Response

Name	Description
<i>CartId</i>	Alphanumeric token returned by CartCreate (p. 214) that identifies a cart.
<i>CartItemID</i>	Specifies an item in a cart. <i>CartItemID</i> is neither an ASIN nor an OfferListingId. It is, instead, an alphanumeric token returned by CartCreate (p. 214) and CartAdd. Child of Item.
<i>HMAC</i>	Hash Message Authentication Code returned by CartCreate (p. 214) that identifies a cart. This is an encrypted alphanumeric token that is used to authenticate cart operations.
<i>Item</i>	For REST, a prefix for CartItemId and Quantity, both of which are used to specify the item to be modified, for example, item.1.CartItemId=1234abcd, item.1.Quantity=2 changes the quantity of item 1234abcd to 2.
<i>Items</i>	Container for Item when specifying more than one Item.
<i>Quantity</i>	The suffix used with Item to specify the number of items to be added to the cart. See Item.

For more information about the parent elements of these tags, see the appropriate response group in [Response Groups \(p. 230\)](#).

Examples

Increase the quantity of an item (Item.1) already in the cart and remove an item (Item.2).

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate ID]&
Operation=CartModify&
CartId=[Cart ID]&
HMAC=[HMAC]&
Item.1.CartItemId=[Cart Item ID]&
Item.1.Quantity=10&
Item.2.CartItemId=[Cart Item ID]&
Item.2.Quantity=0
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

Notice that items in the cart are referred to by *CartItemID*, not their *ASIN* or *OfferListingId*. *CartItemID* is returned by [CartCreate \(p. 214\)](#), [CartGet \(p. 220\)](#), and [CartAdd \(p. 207\)](#). If you want to use *CartModify* to modify the items in a cart, you must refer to the items using their *CartItemID*. You use [CartGet \(p. 220\)](#) to retrieve the *CartItemIDs*.

The response to this request is shown in [Sample Response \(p. 228\)](#).

Move an item (Item.1) to the SaveForLater area in the cart and reduce then number of items (Item.2) to 1.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=CartModify&  
CartId=[Cart ID]&  
HMAC=[HMAC]&  
Item.1.CartItemId=[Cart Item ID]&  
Item.1.Action=SaveForLater  
Item.2.CartItemId=[Cart Item ID]&  
Item.2.Quantity=1  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response

The following XML is a snippet of the full response to the first sample request.

```
<CartModifyRequest>  
  <CartId>103-8104506-5223005</CartId>  
  <HMAC>uiFluRr5yj+GGL3SvWD1mFDyvP0=</HMAC>  
  <Items>  
    <Item>  
      <CartItemId>UQIKL90WK14UD</CartItemId>  
      <Quantity>10</Quantity>  
    </Item>  
  </Items>  
</CartModifyRequest>
```

Notice that the quantity of the item was changed to 10 (from 5). The rest of the `CartModify` request is identical to the [CartCreate \(p. 214\)](#) response.

Related Operations

- [CartAdd \(p. 207\)](#)
- [CartClear \(p. 212\)](#)
- [CartCreate \(p. 214\)](#)
- [CartGet \(p. 220\)](#)

Common Request Parameters

The following table describes the request parameters that all Product Advertising API operations can use.

Product Advertising API Developer Guide
Common Request Parameters

Parameter	Definition	Required
AssociateTag	<p>An alphanumeric token that uniquely identifies an Associate. This token is the means by which Amazon identifies the Associate to credit for a sale. If a request is made without identifying an Associate, Associate fees are not paid by Amazon. If the AssociateTag is included in the CartCreate request, the value for AssociateTag is automatically included in the PurchaseURL, which is returned by CartCreate. To obtain an Associate Tag, see Becoming an Associate (p. 4).</p> <p>Valid values: An alphanumeric token distributed by Amazon that uniquely identifies an Associate. Use this value in all requests to receive credit for the customer purchases.</p>	Yes
AWSAccessKeyId	<p>An alphanumeric token that uniquely identifies a seller. For information about getting an AWS access key ID, see Becoming a Product Advertising API Developer (p. 5).</p> <p>Valid value: AWS access key ID.</p>	Yes
ContentType	<p>Specifies the format of the content in the response. For example, to transform your Product Advertising API response into HTML, set ContentType to text/html.</p> <p>Valid values: text/html, text/xml</p>	No
MerchantId	<p>An optional parameter that can be used to filter search results and offer listings to include only items sold by Amazon. By default, the API returns items sold by various merchants including Amazon. If the value "Amazon" is supplied, only items sold by Amazon are included in the response.</p> <p>Valid values: Amazon</p>	No
Operation	<p>Specifies the Product Advertising API operation to execute. For more information, see Operations (p. 184).</p> <p>Valid values: An Product Advertising API operation, for example, ItemLookup.</p>	Yes
Service	<p>Specifies the Product Advertising API service. There is only one value for all Product Advertising API operations.</p> <p>Valid value: AWSECommerceService</p>	Yes
Validate	<p>Prevents an operation from executing. Set the Validate parameter to True to test your request without actually executing it. When present, Validate must equal True; the default value is False. If a request is not actually executed (Validate=True), only a subset of the errors for a request may be returned because some errors (for example, no_exact_matches) are only generated during the execution of a request.</p> <p>Default: False</p> <p>Valid values: True, False</p>	No

Parameter	Definition	Required
Version	<p>The version of the Product Advertising API software and WSDL to use. By default, the 2013-08-01 version is used. Alternately, specify a software version, such as 2013-08-01.</p> <p>Default: 2013-08-01</p> <p>Valid values: Valid WSDL version date, for example, 2013-08-01.</p>	No
XMLEscaping	<p>Specifies whether responses are XML-encoded in a single pass or a double pass. By default, XMLEscaping is Single, and Product Advertising API responses are encoded only once in XML. For example, if the response data includes an ampersand character (&), the character is returned in its regular XML encoding (&). If XMLEscaping is Double, the same ampersand character is XML-encoded twice (&amp;). The Double value for XMLEscaping is useful in some clients, such as PHP, that do not decode text within XML elements.</p> <p>Default: Single</p> <p>Valid values: Single, Double</p>	No

Response Groups

Response groups help filter the product information you want returned. Each operation can only use some of the available response groups. Each section includes the following:

- Description of the response group.
- Descriptions of the XML elements returned by the response group.
- The ancestry of the element presented using X-path notation.
- Child response groups. For example, if you specify the Large response group, the response includes the child response group, ItemAttributes.
- Parent response groups. For example, the Large response group is a parent of the ItemAttributes response group.

Product Advertising API includes the following response groups:

<ul style="list-style-type: none">• Accessories (p. 231)• AlternateVersions (p. 232)• BrowseNodeInfo (p. 234)• BrowseNodes (p. 236)• Cart (p. 238)• CartNewReleases (p. 241)• CartTopSellers (p. 242)• CartSimilarities (p. 244)• EditorialReview (p. 245)• Images (p. 247)• ItemAttributes (p. 249)• ItemIds (p. 253)• Large (p. 254)	<ul style="list-style-type: none">• Medium (p. 259)• MostGifted (p. 263)• MostWishedFor (p. 264)• NewReleases (p. 265)• OfferFull (p. 267)• OfferListings (p. 269)• Offers (p. 271)• OfferSummary (p. 275)• PromotionSummary (p. 277)• RelatedItems (p. 279)• Request (p. 281)• Reviews (p. 283)• SalesRank (p. 285)• SearchBins (p. 286)• Similarities (p. 291)• Small (p. 293)	<ul style="list-style-type: none">• TopSellers (p. 294)• Tracks (p. 296)• Variations (p. 297)• VariationImages (p. 300)• VariationMatrix (p. 302)• VariationOffers (p. 304)• VariationSummary (p. 307)
--	---	--

Accessories

The Accessories response group returns up to five ASINs and titles of accessories associated with items in the response. For example, if you performed an [ItemLookup \(p. 197\)](#) for a camera, and included the Accessories response group in the request, the response would include, by default, the titles and ASINs of accessories associated with the ASIN. The accessories for the camera might include a camera case, lens, filters, and so on.

Relevant Operations

Operations that can use this response group include:

- [SimilarityLookup \(p. 203\)](#)
- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)

Response Elements

The following table describes the elements returned by Accessories.

- [ASIN \(p. 311\)](#)
- [Title \(p. 325\)](#)
- [TotalPages \(p. 326\)](#)
- [TotalResults \(p. 326\)](#)

Parent Response Group

The following response groups are parent response groups of [Accessories \(p. 231\)](#).

- [Large \(p. 254\)](#)

Child Response Group

The following response groups are child response groups of [Accessories \(p. 231\)](#).

- None

Sample REST Use Case

The following request uses the Accessories response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
IdType=ASIN&  
ItemId=B00008OE6I&  
ResponseGroup=Accessories&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by Accessories.

```
<Item>  
  <ASIN>B00008OE6I</ASIN>  
  <Accessories>  
    <Accessory>  
      <ASIN>B00003G1RG</ASIN>  
      <Title>Viking 128 MB CompactFlash Card (CF128M)</Title>  
    </Accessory>  
    <Accessory>  
      <ASIN>B00004WCCT</ASIN>  
      <Title>Canon Soft Leather Case for Canon Digital ELPH Cameras  
        (Black)</Title>  
    </Accessory>  
    <Accessory>  
      <ASIN>B000051408</ASIN>  
      <Title>SimpleTech STI-CF/128 128MB CompactFlash Card</Title>  
    </Accessory>  
  </Accessories>
```

AlternateVersions

The AlternateVersions response group returns all of the available media formats for a book or music title. Sample formats include Paperback, Audio CD, and MP3 Download. This response group works only with the Books, ForeignBooks, KindleStore, MP3Downloads, and Music product categories.

Relevant Operations

Operations that can use this response group include:

- ItemSearch, when the search index is Music, KindleStore, MP3Downloads, or Books or, in non-US locales, ForeignBooks
- ItemLookup when the specified item is music, a book or foreign book. If a search index is specified, the value must be Music, KindleStore, MP3Downloads, or Books or, in non-US locales, ForeignBooks

Response Elements

The following table describes and shows the parentage of the elements returned by AlternateVersions.

- [AlternateVersion](#) (p. 311)
- [ASIN](#) (p. 311)
- [Binding](#) (p. 312)
- [Title](#) (p. 325)

AlternateVersions also returns common response elements, which is described in [Elements Common to All Response Groups](#) (p. 309).

Parent Response Group

The following response groups are parent response groups of AlternateVersions.

- None

Child Response Group

The following response groups are children response groups of AlternateVersions.

- None

Sample REST Use Case

The following request uses the AlternateVersions response group.

```
http://webservices.amazon.com/onca/xml?
Service=AWSECommerceService&
Operation=ItemSearch&
AWSAccessKeyId=[AWS Access Key ID]&
AssociateTag=[Associate ID]&
SearchIndex=Books&
Keywords=potter&
ResponseGroup=AlternateVersions&
Version=2013-08-01
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by AlternateVersions.

```
<AlternateVersions>
  <AlternateVersion>
```

```
<ASIN>030728364X</ASIN>
<Title>Harry Potter and the Half-Blood Prince (Book 6)</Title>
<Binding>Audio Cassette</Binding>
</AlternateVersion>
<AlternateVersion>
<ASIN>0307283658</ASIN>
<Title>Harry Potter and the Half-Blood Prince (Book 6)</Title>
<Binding>Audio CD</Binding>
</AlternateVersion>
<AlternateVersion>
<ASIN>0439785960</ASIN>
<Title>
    Harry Potter And The Half-Blood Prince (Harry Potter, Book 6) (Harry Potter)
</Title>
<Binding>Paperback</Binding>
</AlternateVersion>
<AlternateVersion>
<ASIN>0439786770</ASIN>
<Title>
    Harry Potter and the Half-Blood Prince (Book 6) [LIBRARY EDITION]
</Title>
<Binding>Library Binding</Binding>
</AlternateVersion>
```

BrowseNodeInfo

For a given browse node ID, the BrowseNodeInfo response group returns the browse node name and ID of the child and parent browse nodes.

One application of this information is to use the child and parent browse nodes to traverse the browse node hierarchy to either refine a search (child nodes) or to generalize a search (parent nodes). You do that using [BrowseNodeLookup \(p. 194\)](#) one or more times. Refining a search enables you to return more highly targeted results. Generalizing a response enables you to find what product category a browse node belongs to.

This response group is similar to the [BrowseNodes \(p. 236\)](#) response group. The difference is that the BrowseNodes response group is used with operations, including [ItemSearch \(p. 185\)](#), [ItemLookup \(p. 197\)](#), and [SimilarityLookup \(p. 203\)](#), that are based on item attributes, search indices, lists, and potentially return multiple items. BrowseNodeInfo can only be used with [BrowseNodeLookup \(p. 194\)](#) and the search is always keyed on a browse node ID.

Relevant Operations

Operations that can use this response group include:

- [BrowseNodeLookup \(p. 194\)](#)

Response Elements

The following table describes the elements returned by BrowseNodeInfo.

- [BrowseNodeId \(p. 312\)](#)
- [Name \(p. 320\)](#)

As you can see from this table, BrowseNodeInfo returns the identity of child and parent browse nodes. That information is typically used with [BrowseNodeLookup \(p. 194\)](#) to traverse the browse node hierarchy.

Parent Response Group

The following response groups are parent response groups of BrowseNodeInfo.

- None

Child Response Group

The following response groups are child response groups of BrowseNodeInfo.

- None

Sample REST Use Case

The following request uses the BrowseNodeInfo response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=BrowseNodeLookup&  
BrowseNodeId=11232&  
ResponseGroup=BrowseNodeInfo&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by BrowseNodeInfo.

```
<Item>  
<ASIN>0131856340</ASIN>  
<BrowseNodes>  
  <BrowseNode>  
    <BrowseNodeId>11232</BrowseNodeId>  
    <Name> Social Sciences</Name>  
    <Ancestors>  
      <BrowseNode>  
        <BrowseNodeId>53</BrowseNodeId>  
        <Name>Nonfiction</Name>  
        <Ancestors>  
          <BrowseNode>  
            <BrowseNodeId>1000</BrowseNodeId>  
            <Name>Subjects</Name>  
            <Ancestors>  
              <BrowseNode>  
                <BrowseNodeId>283155</BrowseNodeId>  
                <Name>Books</Name>  
              </BrowseNode>  
            </Ancestors>  
          </BrowseNode>  
        </Ancestors>  
      </BrowseNode>  
    </Ancestors>  
  </BrowseNode>  
</BrowseNodes>
```

```
</BrowseNode>
</Ancestors>
</BrowseNode>
</Ancestors>
<Children>
<BrowseNode>
<BrowseNodeId>11233</BrowseNodeId>
<Name>Anthropology</Name>
</BrowseNode>
<BrowseNode>
<BrowseNodeId>11242</BrowseNodeId>
<Name>Archaeology</Name>
</BrowseNode>
<BrowseNode>
<BrowseNodeId>3048861</BrowseNodeId>
<Name>Children's Studies</Name>
</BrowseNode>
</Children>
</BrowseNodes>
```

Notice in this response snippet that the child nodes are all at the same level in the hierarchical browse node tree, which is the level directly beneath the browse node in the request, 11232, Social Science. The parent nodes, however, show the entire lineage of browse nodes from Social Science all the way up the browse node tree to the root browse node, Books, 283155. Only one lineage, however, is returned. If a single node has multiple parents, only one of those parents is returned.

BrowseNodes

The BrowseNodes response group returns the browse node names and IDs associated with the items returned in the response. The response group also returns the names and IDs of the child and parent browse nodes of the items returned in the response.

It's possible for an item to belong to multiple browse nodes, so it's common to see multiple hierarchies of browse nodes for a single item.

Some products, such as parent ASINs, do not return information in the BrowsesNodes response group.

This response group is similar to the [BrowseNodeInfo \(p. 234\)](#) response group. The difference is that the BrowseNodes response group is used with operations that are based on item attributes, search indices, and lists. These operations typically return multiple items. BrowseNodeInfo can only be used with [BrowseNodeLookup \(p. 194\)](#) and the search is always keyed on a browse node ID.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by BrowseNodes.

- [BrowseNodeID \(p. 312\)](#)

- IsCategoryRoot (p. 317)
- Name (p. 320)
- TotalPages (p. 326)
- TotalResults (p. 326)

Parent Response Group

The following response groups are parent response groups of BrowseNodes.

- None

Child Response Group

The following response groups are child response groups of BrowseNodes.

- None

Sample REST Use Case

The following request uses the BrowseNodes response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=B000002ADT&  
IdType=ASIN&  
ResponseGroup=BrowseNodes&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by BrowseNodes.

```
<Item>  
  <ASIN>B000002ADT</ASIN>  
  <BrowseNodes>  
    <BrowseNode>  
      <BrowseNodeId>598174</BrowseNodeId>  
      <Name>Bebop</Name>  
      <Ancestors>  
        <BrowseNode>  
          <BrowseNodeId>34</BrowseNodeId>  
          <Name>Jazz</Name>  
          <Ancestors>  
            <BrowseNode>  
              <BrowseNodeId>301668</BrowseNodeId>  
              <Name>Styles</Name>  
              <IsCategoryRoot>1</IsCategoryRoot>
```

```
<Ancestors>
  <BrowseNode>
    <BrowseNodeId>5174</BrowseNodeId>
    <Name>CDs & Vinyl</Name>
  </BrowseNode>
  </Ancestors>
</BrowseNode>
<Ancestors>
  <BrowseNode>
    <BrowseNodeId>63929</BrowseNodeId>
    <Name>Modern Postbebop</Name>
  <Ancestors>
    <BrowseNode>
      <BrowseNodeId>34</BrowseNodeId>
      <Name>Jazz</Name>
    <Ancestors>
      <BrowseNode>
        <BrowseNodeId>301668</BrowseNodeId>
        <Name>Styles</Name>
        <IsCategoryRoot>1</IsCategoryRoot>
      <Ancestors>
        <BrowseNode>
          <BrowseNodeId>5174</BrowseNodeId>
          <Name>CDs & Vinyl</Name>
        </BrowseNode>
      </Ancestors>
    </BrowseNode>
  </Ancestors>
</BrowseNode>
</Ancestors>
</BrowseNode>
...

```

Cart

The Cart response group provides information about a specified remote shopping cart and the items in it. The cart information includes:

- CartId
- HMAC
- PurchaseURL

For each item in the cart, including SavedForLaterItems, the response group returns:

- CartItemId
- ProductName
- ASIN
- Quantity
- ListPrice
- OurPrice

Relevant Operations

Operations that can use this response group include:

- [CartAdd \(p. 207\)](#)
- [CartCreate \(p. 214\)](#)
- [CartModify \(p. 224\)](#)
- [CartGet \(p. 220\)](#)
- [CartClear \(p. 212\)](#)

Response Elements

The following table describes the elements returned by Cart.

- [Amount \(p. 311\)](#)
- [ASIN \(p. 311\)](#)
- [CartId \(p. 312\)](#)
- [CartItem \(p. 312\)](#)
- [CartItemID \(p. 312\)](#)
- [CartItems \(p. 312\)](#)
- [CurrencyCode \(p. 314\)](#)
- [FormattedPrice \(p. 316\)](#)
- [HMAC \(p. 316\)](#)
- [ParentASIN \(p. 321\)](#)
- [Price \(p. 322\)](#)
- [ProductGroup \(p. 322\)](#)
- [PurchaseURL \(p. 322\)](#)
- [Quantity \(p. 322\)](#)
- [SavedForLaterItem \(p. 323\)](#)
- [Title \(p. 325\)](#)
- [URLEncodedHMAC \(p. 327\)](#)

Parent Response Group

The following response groups are parent response groups of Cart.

- None

Child Response Group

The following response groups are child response groups of Cart.

- None

Sample REST Use Case

The following request uses the Cart response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=CartCreate&  
Item.1.ASIN=B000062TU1&  
Item.1.Quantity=2&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by Cart.

```
<Cart>  
  <Request>  
    <IsValid>True</IsValid>  
    <CartCreateRequest>  
      <Items>  
        <Item>  
          <ASIN>B000062TU1</ASIN>  
          <Quantity>2</Quantity>  
        </Item>  
      </Items>  
    </CartCreateRequest>  
  </Request>  
  <CartId>002-5281165-2803250</CartId>  
  <HMAC>5ilu00G/PHqkvxZqC8oRkzmCano=</HMAC>  
  <URLEncodedHMAC>5ilu00G%2FPHQkvxZqC8oRkzmCano%3D</URLEncodedHMAC>  
  <PurchaseURL>https://www.amazon.com/gp/cart/aws-merge.html?cart-id=002-5281165-  
2803250%26associate-id=ws%26hmac=5ilu00G/PHqkvxZqC8oRkzmCano=%26Subscrip-  
tionId=AKIAI44QH8DHBEAMPLE</PurchaseURL>  
  <SubTotal>  
    <Amount>1994</Amount>  
    <CurrencyCode>USD</CurrencyCode>  
    <FormattedPrice>$19.94</FormattedPrice>  
  </SubTotal>  
  <CartItems>  
    <SubTotal>  
      <Amount>1994</Amount>  
      <CurrencyCode>USD</CurrencyCode>  
      <FormattedPrice>$19.94</FormattedPrice>  
    </SubTotal>  
    <CartItem>  
      <CartItemId>U2ABORWEFJ0WZP</CartItemId>  
      <ASIN>B000062TU1</ASIN>  
      <Quantity>2</Quantity>  
      <Title>Harry Potter and the Sorcerer's Stone (Full Screen Edition) (Harry  
Potter 1)</Title>  
      <ProductGroup>DVD</ProductGroup>  
        <Price>  
          <Amount>997</Amount>  
          <CurrencyCode>USD</CurrencyCode>  
          <FormattedPrice>$9.97</FormattedPrice>  
        </Price>  
      <ItemTotal>
```

```
<Amount>1994</Amount>
<CurrencyCode>USD</CurrencyCode>
<FormattedPrice>$19.94</FormattedPrice>
</ItemTotal>
</CartItem>
</CartItems>
</Cart>
```

CartNewReleases

The CartNewReleases response group returns the ASINs and titles of the top five new releases in the root category of the item specified in the cart operation. For example, when adding a television to a cart, the top five new releases in the root category, electronics, are returned.

Use this response group to suggest additional items to customers.

Availability

This response group is available in the US locale only.

Relevant Operations

Operations that can use this response group include:

- [CartAdd \(p. 207\)](#)
- [CartCreate \(p. 214\)](#)
- [CartModify \(p. 224\)](#)
- [CartGet \(p. 220\)](#)

Response Elements

The following table describes the elements returned by CartNewReleases.

- [ASIN \(p. 311\)](#)
- [Title \(p. 325\)](#)

Parent Response Group

The following response groups are parent response groups of CartNewReleases.

- None

Child Response Group

The following response groups are child response groups of CartNewReleases.

- None

Sample REST Use Case

The following request uses the CartNewReleases response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=CartCreate&  
Item.1.ASIN=B000062TU1&  
Item.1.Quantity=2&  
ResponseGroup=CartNewReleases&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by CartNewReleases.

```
<NewReleases>  
  <NewRelease>  
    <ASIN>B00005JOFQ</ASIN>  
    <Title>Brokeback Mountain (Widescreen Edition)</Title>  
  </NewRelease>  
  <NewRelease>  
    <ASIN>B000EHRVMY</ASIN>  
    <Title>Memoirs of a Geisha (Widescreen 2-Disc Special Edition)</Title>  
  </NewRelease>  
</NewReleases>
```

CartTopSellers

The CartTopSellers response group returns the ASINs and titles of the top five, best sellers in the root category of the item specified in the cart operation. For example, when adding a television to a cart, the top five sellers in the root category, electronics, are returned such as the top selling computers, MP3 players, or cameras.

Use this response group to specify related items to buy.

Availability

This response group is available in the US locale only.

Relevant Operations

Operations that can use this response group include:

- [CartAdd \(p. 207\)](#)
- [CartCreate \(p. 214\)](#)
- [CartModify \(p. 224\)](#)
- [CartGet \(p. 220\)](#)

Response Elements

The following table describes the elements returned by CartTopSellers.

- ASIN (p. 311)
- Title (p. 325)

CartTopSellers also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of CartTopSellers.

- None

Child Response Group

The following response groups are child response groups of CartTopSellers.

- None

Sample REST Use Case

The following request uses the CartTopSellers response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=CartCreate&  
Item.1.ASIN=B000062TU1&  
Item.1.Quantity=2&  
ResponseGroup=CartTopSellers&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by CartTopSellers.

```
<TopSellers>  
<TopSeller>  
  <ASIN>B00005JOFQ</ASIN>  
  <Title>Brokeback Mountain (Widescreen Edition)</Title>  
  </TopSeller>  
<TopSeller>  
  <ASIN>B000E6EK3S</ASIN>  
  <Title>Harry Potter and the Goblet of Fire (Widescreen Two-Disc Deluxe Edition)  
(Harry Potter 4)</Title>  
  </TopSeller>  
</TopSellers>
```

CartSimilarities

The [CartSimilarities \(p. 244\)](#) response group returns the title and ASINs of items that:

- Are similar to the item specified in the request. These results are tagged in the XML response with <SimilarProducts> and <SimilarProduct> and appear on the retail website under the heading, "Customers who bought this [item] also bought."
- Have been viewed by customers who also viewed the item specified in the request. These results are tagged in the XML response with <SimilarViewedProducts> and <SimilarViewedProduct> and appear on the retail website under the heading, "Customers who viewed this [item] also viewed."
- Can be found in other categories that are similar to the item specified in the request. These results are tagged in the XML response with <OtherCategoriesSimilarProducts> and <OtherCategoriesSimilarProduct> and appear on the retail website under the heading, "Explore Similar Items," which falls under the heading, "Customers who bought this [item] also bought."

This response group returns items similar to the one the customer added to the shopping cart. The response group is used to encourage the customer to purchase additional items.

Relevant Operations

Operations that can use this response group include:

- [CartAdd \(p. 207\)](#)
- [CartCreate \(p. 214\)](#)
- [CartModify \(p. 224\)](#)
- [CartGet \(p. 220\)](#)

Response Elements

The following table describes the elements returned by CartSimilarities.

- [ASIN \(p. 311\)](#)
- [OtherCategoriesSimilarProducts \(p. 321\)](#)
- [SimilarProducts \(p. 324\)](#)
- [SimilarViewedProducts \(p. 324\)](#)
- [Title \(p. 325\)](#)

CartSimilarities also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of CartSimilarities.

- None

Child Response Group

The following response groups are child response groups of CartSimilarities.

- None

Sample REST Use Case

The following request uses the CartSimilarities response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=CartCreate&  
Item.1.ASIN=B000062TU1&  
Item.1.Quantity=2&  
ResponseGroup=CartSimilarities&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by CartSimilarities.

```
<SimilarProducts>  
  <SimilarProduct>  
    <ASIN>B00008DDXC</ASIN>  
    <Title>Harry Potter and the Chamber of Secrets (Widescreen Edition) (Harry  
Potter 2)</Title>  
  </SimilarProduct>  
</SimilarProducts>  
<SimilarViewedProducts>  
  <SimilarViewedProduct>  
    <ASIN>B000E6UZZK</ASIN>  
    <Title>Harry Potter Years 1-4 (Harry Potter and the Sorcerer's Stone /  
Chamber of Secrets / Prisoner of Azkaban / Goblet of Fire) (Widescreen Edi  
tion)</Title>  
  </SimilarViewedProduct>  
</SimilarViewedProducts>  
<OtherCategoriesSimilarProducts>  
  <OtherCategoriesSimilarProduct>  
    <ASIN>0590353403</ASIN>  
    <Title>Harry Potter and the Sorcerer's Stone (Book 1)</Title>  
  </OtherCategoriesSimilarProduct>  
</OtherCategoriesSimilarProducts>
```

EditorialReview

The EditorialReview response group returns Amazon's review of the item, which appears on the Product Detail page for each item in the response.

Note

Copyrighted editorial reviews are not returned. For this reason, the reviews returned may be different than those returned by *Amazon.com*.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)

- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by EditorialReview.

- [EditorialReviewIsLinkSuppressed \(p. 315\)](#)
- [Source \(p. 324\)](#)

EditorialReview also returns the elements that all response groups return, which is described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of EditorialReview.

- None

Child Response Group

The following response groups are child response groups of EditorialReview.

- None

Sample REST Use Case

The following request uses the EditorialReview response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&Keywords=Fable&  
SearchIndex=Blended&  
ResponseGroup=EditorialReview&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by EditorialReview.

```
<EditorialReviews>  
  <EditorialReview>  
    <Source>Product Description</Source>  
    <Content>Considered by many to be the last great musical comedy, then the  
    rest of the review continues here.  
    </Content>  
  </EditorialReview>  
</EditorialReviews>
```

Images

The Images response group returns the URLs to all available images of an item in three sizes: small, medium, and large. For example, if a blender has four images, this response group returns the URLs of 12 images: four images, each in three sizes.

In addition to returning the image URLs, the response groups returns the height and width dimensions of each image. Use these values to display the images correctly.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by the Images response group.

- [Height \(p. 316\)](#)
- [LargeImage \(p. 319\)](#)
- [MediumImage \(p. 320\)](#)
- [SmallImage \(p. 324\)](#)
- [SwatchImage \(p. 325\)](#)
- [ThumbnailImage \(p. 325\)](#)
- [TinyImage \(p. 325\)](#)
- [TotalPages \(p. 326\)](#)
- [TotalResults \(p. 326\)](#)
- [URL \(p. 327\)](#)
- [Width \(p. 328\)](#)

Images also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of the Images response group.

- None

Child Response Group

The following response groups are child response groups of the Images response group.

- None

Sample REST Use Case

The following request uses the Images response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
ItemId=B000Q67800&  
ResponseGroup=Images&  
SearchIndex=Blended&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by the Images response group.

```
<Item>  
  <ASIN>B000Q67800</ASIN>  
  <SmallImage>  
    <URL>http://ecx.images-amazon.com/images/I/51YL4rlI%2B9L._SL75_.jpg</URL>  
    <Height Units="pixels">75</Height>  
    <Width Units="pixels">58</Width>  
  </SmallImage>  
  <MediumImage>  
    <URL>http://ecx.images-amazon.com/images/I/51YL4rlI%2B9L._SL160_.jpg</URL>  
    <Height Units="pixels">160</Height>  
    <Width Units="pixels">124</Width>  
  </MediumImage>  
  <LargeImage>  
    <URL>http://ecx.images-amazon.com/images/I/51YL4rlI%2B9L.jpg</URL>  
    <Height Units="pixels">500</Height>  
    <Width Units="pixels">389</Width>  
  </LargeImage>  
  
<ImageSets>  
  <ImageSet Category="primary">  
    <SwatchImage>  
      <URL>http://ecx.images-amazon.com/images/I/51YL4rlI%2B9L._SL30_.jpg</URL>  
      <Height Units="pixels">30</Height>  
      <Width Units="pixels">23</Width>  
    </SwatchImage>  
    <SmallImage>  
      <URL>http://ecx.images-amazon.com/images/I/51YL4rlI%2B9L._SL75_.jpg</URL>  
      <Height Units="pixels">75</Height>  
      <Width Units="pixels">58</Width>  
    </SmallImage>  
    <ThumbnailImage>  
      <URL>http://ecx.images-amazon.com/images/I/51YL4rlI%2B9L._SL75_.jpg</URL>  
      <Height Units="pixels">75</Height>  
      <Width Units="pixels">58</Width>  
    </ThumbnailImage>  
    <TinyImage>
```

```
<URL>http://ecx.images-amazon.com/images/I/51YL4rlI%2B9L._SL110_.jpg</URL>
<Height Units="pixels">110</Height>
<Width Units="pixels">86</Width>
</TinyImage>
<MediumImage>
<URL>http://ecx.images-amazon.com/images/I/51YL4rlI%2B9L._SL160_.jpg</URL>
<Height Units="pixels">160</Height>
<Width Units="pixels">124</Width>
</MediumImage>
<LargeImage>
<URL>http://ecx.images-amazon.com/images/I/51YL4rlI%2B9L.jpg</URL>
<Height Units="pixels">500</Height>
<Width Units="pixels">389</Width>
</LargeImage>
</ImageSet>
</ImageSets>
```

The images are returned in two ways: under `<Item>` and under `<ImageSets>`. The images under `<Item>` are specified by the `<SmallImage>` , `<MediumImage>` , and `<LargeImage>` elements.

The sizes of the images under `<ImageSets>` are specified by the `_SLXXX_` suffix in the URL, where XXX is the number of pixels on the longest side of the image. For example, a medium size image has 160 pixels on its longest side, so it has the suffix `_SL160_`. This is the preferred way to reference images.

The `ImageSets` element attribute, `Category`, is set to Primary. Primary images are the same images that appear in the `<Item>` section.

ItemAttributes

The ItemAttributes response group returns a potentially large number of attributes that describe an item. For example, an item in the Camera and Photo search index might return the attributes, height, width, weight, title, UPC, price, manufacturer, zoom ratio, number of megapixels, and carrying case.

All search indices can return all item attributes. However, the number of item attributes returned varies by ASIN. Typically, ASINs within the same search index return the same item attributes. For example, the item attributes returned for an item in the "Books" search index will be different from those returned for an item in the "Camera and Photo" search index. However, items within a single search index do not necessarily return the same attributes.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by ItemAttributes.

- [Actor \(p. 310\)](#)
- [Artist \(p. 311\)](#)
- [AspectRatio \(p. 311\)](#)

- AudienceRating (p. 311)
- AudioFormat (p. 311)
- Author (p. 311)
- Binding (p. 312)
- Brand (p. 312)
- Category (p. 313)
- CEROAgeRating (p. 313)
- ClothingSize (p. 313)
- Color (p. 313)
- Creator (p. 314)
 - Role (p. 323)
- Department (p. 314)
- Director (p. 314)
- EAN (p. 314)
- EANList (p. 314)
 - EANListElement (p. 315)
- Edition (p. 315)
- EISBN (p. 315)
- EpisodeSequence (p. 315)
- ESRBAgeRating (p. 315)
- Feature (p. 315)
- Format (p. 316)
- Genre (p. 316)
- HardwarePlatform (p. 316)
- HazardousMaterialType (p. 316)
- IsAdultProduct (p. 317)
- IsAutographed (p. 317)
- ISBN (p. 317)
- IsEligibleForTradeIn (p. 317)
- IsMemorabilia (p. 318)
- IssuesPerYear (p. 318)
- ItemDimensions (p. 318)
 - Height (p. 316)
 - Length (p. 319)
 - Weight (p. 327)
 - Width (p. 328)
- ItemPartNumber (p. 318)
- Label (p. 318)
- Languages (p. 319)
 - Language (p. 318)
 - Name (p. 320)
 - Type (p. 327)
 - AudioFormat (p. 311)
- LegalDisclaimer (p. 319)
- ListPrice (p. 319)
- Manufacturer (p. 319)

- ManufacturerMaximumAge (p. 319)
- ManufacturerMinimumAge (p. 319)
- ManufacturerPartsWarrantyDescription (p. 319)
- MediaType (p. 320)
- Model (p. 320)
- MPN (p. 320)
- NumberOfDiscs (p. 321)
- NumberOfIssues (p. 321)
- NumberOfItems (p. 321)
- NumberOfPages (p. 321)
- NumberOfTracks (p. 321)
- OperatingSystem (p. 321)
- PackageQuantity (p. 321)
- PartNumber (p. 321)
- Platform (p. 322)
- ProductGroup (p. 322)
- ProductTypeSubcategory (p. 322)
- PublicationDate (p. 322)
- Publisher (p. 322)
- RegionCode (p. 322)
- ReleaseDate (p. 323)
- RunningTime (p. 323)
- SeikodoProductCode (p. 324)
- Size (p. 324)
- SKU (p. 324)
- Studio (p. 324)
- SubscriptionLength (p. 324)
- Title (p. 325)
- TradeInValue (p. 326)
- UPC (p. 327)
- UPCList (p. 327)
 - UPCL ElementType (p. 327)
- Warranty (p. 327)
- WEEETaxValue (p. 327)

ItemAttributes also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of ItemAttributes.

- None

Child Response Group

The following response groups are child response groups of ItemAttributes.

- None

Sample REST Use Case

The following request uses the ItemAttributes response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
Condition=All&  
ResponseGroup=ItemAttributes&  
SearchIndex=Blended&  
Keywords=GodSmack&  
Merchant=All&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by ItemAttributes.

```
<Item>  
  <ASIN>B000A2XB9U</ASIN>  
  <ItemAttributes>  
    <AudienceRating>NR (Not Rated)</AudienceRating>  
    <Director>Lawrence Jordan (II)</Director>  
    <Director>Daniel E. Catullo</Director>  
    <EAN>0014381273229</EAN>  
    <Format>Color</Format>  
    <Format>Compilation</Format>  
    <Format>NTSC</Format>  
    <Languages>  
      <Language>  
        <Name>English</Name>  
        <Type>Original Language</Type>  
      </Language>  
    </Languages>  
    <ListPrice>  
      <Amount>1999</Amount>  
      <CurrencyCode>USD</CurrencyCode>  
      <FormattedPrice>$19.99</FormattedPrice>  
    </ListPrice>  
    <NumberOfItems>1</NumberOfItems>  
    <ProductGroup>DVD</ProductGroup>  
    <ReleaseDate>2005-09-06</ReleaseDate>  
    <RunningTime Units="minutes">131</RunningTime>  
    <Studio>Image Entertainment</Studio>  
    <TheatricalReleaseDate>2005-06-25</TheatricalReleaseDate>  
    <Title>Rockin' the Corps</Title>  
    <UPC>014381273229</UPC>  
  </ItemAttributes>  
</Item>
```

ItemIds

The ItemIds response group returns the ASINs for all items returned in a response.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by ItemIds.

- [ASIN \(p. 311\)](#)
- [CorrectedQuery \(p. 313\)](#)
- [Keywords \(p. 318\)](#)
- [Message \(p. 320\)](#)
- [TotalPages \(p. 326\)](#)
- [TotalResults \(p. 326\)](#)

ItemIds also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of ItemIds.

- None

Child Response Group

The following response groups are child response groups of ItemIds.

- None

Sample REST Use Case

The following request uses the ItemIds response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=0976925524&  
ResponseGroup=ItemIds&  
Version=2013-08-01
```

```
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by ItemIds.

```
<TotalResults>1</TotalResults>  
<TotalPages>1</TotalPages>  
<Item>  
  <ASIN>0976925524</ASIN>  
</Item>  
</Items>
```

Large

The Large response group returns a lot of information about items in the response. The Large response group is for demonstration purposes only. It is not intended for production applications. You should use precise required response groups in applications to reduce latency and response size.

Large is a parent response group that returns the results of the following response groups:

- [Accessories \(p. 231\)](#)
- [BrowseNodes \(p. 236\)](#)
- [Medium \(p. 259\)](#)
- [Offers \(p. 271\)](#)
- [Reviews \(p. 283\)](#)
- [Similarities \(p. 291\)](#)
- [Tracks \(p. 296\)](#)

For more information about what is returned by each of these response groups, refer to their sections.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by Large.

- [Actor \(p. 310\)](#)
- [Amount \(p. 311\)](#)
- [Artist \(p. 311\)](#)
- [ASIN \(p. 311\)](#)
- [AspectRatio \(p. 311\)](#)
- [AudienceRating \(p. 311\)](#)

- [AudioFormat \(p. 311\)](#)
- [Author \(p. 311\)](#)
- [Binding \(p. 312\)](#)
- [Brand \(p. 312\)](#)
- [BrowseNodeID \(p. 312\)](#)
- [ClothingSize \(p. 313\)](#)
- [Color \(p. 313\)](#)
- [Creator \(p. 314\)](#)
- [CurrencyCode \(p. 314\)](#)
- [Department \(p. 314\)](#)
- [Director \(p. 314\)](#)
- [EAN \(p. 314\)](#)
- [Edition \(p. 315\)](#)
- [EISBN \(p. 315\)](#)
- [ESRBAgeRating \(p. 315\)](#)
- [Feature \(p. 315\)](#)
- [Format \(p. 316\)](#)
- [FormattedPrice \(p. 316\)](#)
- [Height \(p. 316\)](#)
- [IsAutographed \(p. 317\)](#)
- [ISBN \(p. 317\)](#)
- [IsMemorabilia \(p. 318\)](#)
- [IssuesPerYear \(p. 318\)](#)
- [Keywords \(p. 318\)](#)
- [Label \(p. 318\)](#)
- [LegalDisclaimer \(p. 319\)](#)
- [Length \(p. 319\)](#)
- [Manufacturer \(p. 319\)](#)
- [ManufacturerMaximumAge \(p. 319\)](#)
- [ManufacturerMinimumAge \(p. 319\)](#)
- [ManufacturerPartsWarrantyDescription \(p. 319\)](#)
- [MaterialType \(p. 319\)](#)
- [Message \(p. 320\)](#)
- [MetalType \(p. 320\)](#)
- [Model \(p. 320\)](#)
- [MPN \(p. 320\)](#)
- [NumberOfDiscs \(p. 321\)](#)
- [NumberOfIssues \(p. 321\)](#)
- [NumberOfItems \(p. 321\)](#)
- [NumberOfPages \(p. 321\)](#)
- [NumberOfTracks \(p. 321\)](#)
- [Platform \(p. 322\)](#)
- [ProductGroup \(p. 322\)](#)
- [PublicationDate \(p. 322\)](#)
- [Publisher \(p. 322\)](#)
- [RegionCode \(p. 322\)](#)

- [ReleaseDate](#) (p. 323)
- [Role](#) (p. 323)
- [RunningTime](#) (p. 323)
- [SalesRank](#) (p. 323)
- [Size](#) (p. 324)
- [SKU](#) (p. 324)
- [Source](#) (p. 324)
- [Studio](#) (p. 324)
- [SubscriptionLength](#) (p. 324)
- [Title](#) (p. 325)
- [TotalCollectible](#) (p. 325)
- [TotalUsed](#) (p. 326)
- [TotalNew](#) (p. 325)
- [TotalPages](#) (p. 326)
- [TotalRefurbished](#) (p. 326)
- [TotalResults](#) (p. 326)
- [Type](#) (p. 327)
- [UPC](#) (p. 327)
- [URL](#) (p. 327)
- [Warranty](#) (p. 327)
- [Weight](#) (p. 327)
- [Width](#) (p. 328)

Large also returns the elements that all response groups return, as described in [Elements Common to All Response Groups](#) (p. 309).

Parent Response Group

The following response groups are parent response groups of Large.

- None

Child Response Group

The following response groups are child response groups of Large.

- [Accessories](#) (p. 231)
- [BrowseNodes](#) (p. 236)
- [Medium](#) (p. 259)
- [Offers](#) (p. 271)
- [Reviews](#) (p. 283)
- [Similarities](#) (p. 291)
- [Tracks](#) (p. 296)

Sample REST Use Case

The following request uses the Large response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=B000ESHHXG&  
ResponseGroup=Large&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by Large.

```
<Item>  
  <ASIN>B000A3UB20</ASIN>  
  <SalesRank>47589</SalesRank>  
  <SmallImage>  
    <URL>http://ec1.images-amazon.com/images/P/B000A3UB20.01-  
A3QXF272WQ86AH._SCTHUMBZZZ_.jpg</URL>  
    <Height Units="pixels">75</Height>  
    <Width Units="pixels">75</Width>  
  </SmallImage>  
  <MediumImage>  
    <URL>http://ec1.images-amazon.com/images/P/B000A3UB20.01-  
A3QXF272WQ86AH._SCMZZZZZZ_.jpg</URL>  
    <Height Units="pixels">160</Height>  
    <Width Units="pixels">160</Width>  
  </MediumImage>  
  <LargeImage>  
    <URL>http://ec1.images-amazon.com/images/P/B000A3UB20.01-  
A3QXF272WQ86AH._SCLZZZZZZ_.jpg</URL>  
    <Height Units="pixels">450</Height>  
    <Width Units="pixels">450</Width>  
  </LargeImage>  
  <ImageSets>  
    <ImageSet Category="primary">  
      <SmallImage>  
        <URL>http://ec1.images-amazon.com/images/P/B000A3UB20.01-A3QXF272WQ86A  
H._SCTHUMBZZZ_.jpg</URL>  
        <Height Units="pixels">75</Height>  
        <Width Units="pixels">75</Width>  
      </SmallImage>  
    ...  
    <ItemAttributes>  
      <Brand>Nixon</Brand>  
      <ProductGroup>Apparel</ProductGroup>  
      <Title>Nixon Rotolog Wood</Title>  
    </ItemAttributes>  
    <OfferSummary>  
      <LowestNewPrice>  
        <Amount>19999</Amount>  
        <CurrencyCode>USD</CurrencyCode>  
        <FormattedPrice>$199.99</FormattedPrice>  
      </LowestNewPrice>  
      <TotalNew>1</TotalNew>
```

```
<TotalUsed>0</TotalUsed>
<TotalCollectible>0</TotalCollectible>
<TotalRefurbished>0</TotalRefurbished>
</OfferSummary>
<Offers>
    <TotalOffers>0</TotalOffers>
    <TotalOfferPages>0</TotalOfferPages>
</Offers>
<EditorialReviews>
    <EditorialReview>
        <Source>Product Description</Source>
        <Content>Custom right read direct time. Japanese quartz with led. Custom  
30 Meter stainless steel with hardened mineral crystal, white inlay and double  
gasket crown. Custom solid stainless steel with white inlay and butterfly  
closure.</Content>
    </EditorialReview>
</EditorialReviews>
<BrowseNodes>
    <BrowseNode>
        <BrowseNodeId>1045534</BrowseNodeId>
        <Name>Jewelry</Name>
        <Ancestors>
            <BrowseNode>
                <BrowseNodeId>1044486</BrowseNodeId>
                <Name>Girls</Name>
                <Ancestors>
                    <BrowseNode>
                        <BrowseNodeId>1044484</BrowseNodeId>
                        <Name>Children's Accessories</Name>
                    <Ancestors>
                        <BrowseNode>
                            <BrowseNodeId>1036700</BrowseNodeId>
                            <Name>Accessories</Name>
                        <Ancestors>
                            <BrowseNode>
                                <BrowseNodeId>1036682</BrowseNodeId>
                                <Name>Departments</Name>
                            <Ancestors>
                                <BrowseNode>
                                    <BrowseNodeId>1036592</BrowseNodeId>
                                    <Name>Apparel</Name>
                                </BrowseNode>
                            </Ancestors>
                        </BrowseNode>
                    </Ancestors>
                </BrowseNode>
            </Ancestors>
        <BrowseNode>
            <BrowseNodeId>1036592</BrowseNodeId>
            <Name>Apparel</Name>
        </BrowseNode>
    </Ancestors>
</BrowseNode>
</Ancestors>
</BrowseNode>
</Ancestors>
</BrowseNode>
</Ancestors>
</BrowseNode>
<BrowseNode>
```

Medium

The Medium response group returns a lot of information about the items in a response. The Medium response group is for demonstration purposes only. It is not intended for production applications. You should use precise required response groups in applications to reduce latency and response size.

Medium is a parent response group that returns the results of the following response groups:

- [EditorialReview \(p. 245\)](#)
- [Images \(p. 247\)](#)
- [ItemAttributes \(p. 249\)](#)
- [OfferSummary \(p. 275\)](#)
- [Request \(p. 281\)](#)
- [SalesRank \(p. 285\)](#)
- [Small \(p. 293\)](#)

For more information about what is returned by each of these response groups, refer to their sections.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by Medium.

- [Actor \(p. 310\)](#)
- [Amount \(p. 311\)](#)
- [Artist \(p. 311\)](#)
- [ASIN \(p. 311\)](#)
- [AspectRatio \(p. 311\)](#)
- [AudienceRating \(p. 311\)](#)
- [AudioFormat \(p. 311\)](#)
- [Author \(p. 311\)](#)
- [Binding \(p. 312\)](#)
- [Brand \(p. 312\)](#)
- [ClothingSize \(p. 313\)](#)
- [Color \(p. 313\)](#)
- [Creator \(p. 314\)](#)
- [CurrencyCode \(p. 314\)](#)
- [Department \(p. 314\)](#)
- [Director \(p. 314\)](#)
- [EAN \(p. 314\)](#)
- [Edition \(p. 315\)](#)
- [EISBN \(p. 315\)](#)

- ESRBAgeRating (p. 315)
- Feature (p. 315)
- Format (p. 316)
- FormattedPrice
- Height (p. 316)
- IsAutographed (p. 317)
- ISBN (p. 317)
- IsMemorabilia (p. 318)
- IssuesPerYear (p. 318)
- Keywords (p. 318)
- Label (p. 318)
- LegalDisclaimer (p. 319)
- Length (p. 319)
- Manufacturer (p. 319)
- ManufacturerMaximumAge (p. 319)
- ManufacturerMinimumAge (p. 319)
- ManufacturerPartsWarrantyDescription (p. 319)
- MaterialType (p. 319)
- Message (p. 320)
- MetalType (p. 320)
- Model (p. 320)
- MPN (p. 320)
- NumberOfDiscs (p. 321)
- NumberOfIssues (p. 321)
- NumberOfItems (p. 321)
- NumberOfPages (p. 321)
- NumberOfTracks (p. 321)
- Platform (p. 322)
- ProductGroup (p. 322)
- PublicationDate (p. 322)
- Publisher (p. 322)
- RegionCode (p. 322)
- ReleaseDate (p. 323)
- Role (p. 323)
- RunningTime (p. 323)
- SalesRank (p. 323)
- Size (p. 324)
- SKU (p. 324)
- Source (p. 324)
- Studio (p. 324)
- SubscriptionLength (p. 324)
- Title (p. 325)
- TotalCollectible (p. 325)
- TotalUsed (p. 326)
- TotalNew (p. 325)
- TotalPages (p. 326)

- TotalRefurbished (p. 326)
- TotalResults (p. 326)
- Type (p. 327)
- UPC (p. 327)
- URL (p. 327)
- Warranty (p. 327)
- Weight (p. 327)
- Width (p. 328)

Medium also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of Medium.

- Large (p. 254)

Child Response Group

The following response groups are child response groups of Medium.

- EditorialReview (p. 245)
- Images (p. 247)
- ItemAttributes (p. 249)
- OfferSummary (p. 275)
- Request (p. 281)
- SalesRank (p. 285)
- Small (p. 293)

Sample REST Use Case

The following request uses the Medium response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=1890966533&  
ResponseGroup=Medium&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by Medium.

```
<Item>
    <ASIN>1890966533</ASIN>
    <SalesRank>1738674</SalesRank>
    <SmallImage>
        <URL>http://ecl.images-amazon.com/images/P/1890966533.01._SCTHUM
BZZZ_.jpg</URL>
        <Height Units="pixels">60</Height>
        <Width Units="pixels">40</Width>
    </SmallImage>
    <MediumImage>
        <URL>http://ecl.images-amazon.com/im
ages/P/1890966533.01._SCMZZZZZZ_.jpg</URL>
        <Height Units="pixels">140</Height>
        <Width Units="pixels">94</Width>
    </MediumImage>
    <ItemAttributes>
        <Author>Wayne G. Pardy</Author>
        <Binding>Paperback</Binding>
        <EAN>9781890966539</EAN>
        <Edition>Reprint</Edition>
        <ISBN>1890966533</ISBN>
        <ListPrice>
            <Amount>5995</Amount>
            <CurrencyCode>USD</CurrencyCode>
            <FormattedPrice>$59.95</FormattedPrice>
        </ListPrice>
        <NumberOfPages>185</NumberOfPages>
        <PackageDimensions>
            <Height Units="hundredths-inches">900</Height>
            <Length Units="hundredths-inches">75</Length>
            <Weight Units="hundredths-pounds">130</Weight>
            <Width Units="hundredths-inches">600</Width>
        </PackageDimensions>
        <ProductGroup>Book</ProductGroup>
        <PublicationDate>1999-05-19</PublicationDate>
        <Publisher>Safetycertified.Com Inc</Publisher>
        <Title>Safety Incentives: The Pros and Cons of Award and Recognition
Programs (Osha Compliance)</Title>
    </ItemAttributes>
    <OfferSummary>
        <TotalNew>0</TotalNew>
        <TotalUsed>0</TotalUsed>
        <TotalCollectible>0</TotalCollectible>
        <TotalRefurbished>0</TotalRefurbished>
    </OfferSummary>
    <EditorialReviews>
        <EditorialReview>
            <Source>Book Description</Source>
            <Content>Whether you are a small business or a safety manager trying
to decide whether or not awards and incentives are right for your operation,
the Safety Incentives Answer Book is must reading.
            </Content>
        </EditorialReview>
    </EditorialReviews>
</Item>
```

MostGifted

The MostGifted response group returns the ASINs and titles of the 10 items given as gifts within a specified browse node.

Availability

This response group is available in all locales.

Relevant Operations

Operations that can use this response group include:

- [BrowseNodeLookup \(p. 194\)](#)

Response Elements

The following table describes the elements returned by MostGifted. In the Ancestry column, the elements on the left side of a slash mark are the parents of the elements on the right side of the slash mark.

- [Actors \(p. 310\)](#)
- [Artist \(p. 311\)](#)
- [ASIN \(p. 311\)](#)
- [Authors \(p. 311\)](#)
- [ProductGroup \(p. 322\)](#)
- [Title \(p. 325\)](#)

MostGifted also returns the elements that all response groups return, which is described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of MostGifted.

- None

Child Response Group

The following response groups are child response groups of MostGifted.

- None

Sample REST Use Case

The following request uses the MostGifted response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&
```

```
Operation=BrowseNodeLookup&
BrowseNodeId=20&
ResponseGroup=MostGifted&
Version=2013-08-01
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by MostGifted.

```
<TopItemSet>
  <Type>MostGifted</Type>
  <TopItem>
    <ASIN>0553576399</ASIN>
    <Title>Distraction</Title>
    <ProductGroup>Book</ProductGroup>
    <Author>Bruce Sterling</Author>
  </TopItem>
  <TopItem>
    ....
    ....
  </TopItemSet>
```

MostWishedFor

The MostWishedFor response group returns the ASINs and titles of the 10 items that are the most popular on wishlists within a specified browse node.

Availability

This response group is available in all locales.

Relevant Operations

Operations that can use this response group include:

- [BrowseNodeLookup \(p. 194\)](#)

Response Elements

The following table describes the elements returned by MostWishedFor.

- [Actors \(p. 310\)](#)
- [Artist \(p. 311\)](#)
- [ASIN \(p. 311\)](#)
- [Authors \(p. 311\)](#)
- [ProductGroup \(p. 322\)](#)
- [Title \(p. 325\)](#)

MostWishedFor also returns the elements that all response groups return, which is described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of MostWishedFor.

- None

Child Response Group

The following response groups are child response groups of MostWishedFor.

- None

Sample REST Use Case

The following request uses the MostWishedFor response element.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=BrowseNodeLookup&  
BrowseNodeId=20&  
ResponseGroup=MostWishedFor&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by MostWishedFor.

```
<TopItemSet>  
    <Type>MostWishedFor</Type>  
    <TopItem>  
        <ASIN>0553576399</ASIN>  
        <Title>Distraction</Title>  
        <ProductGroup>Book</ProductGroup>  
        <Author>Bruce Sterling</Author>  
    </TopItem>  
    <TopItem>  
        ....  
        ....  
    </TopItem>  
</TopItemSet>
```

NewReleases

The NewReleases response group returns the ASIN and title of newly released items in a specified browse node.

Availability

This response group is available in the US locale only.

Relevant Operations

Operations that can use this response group include:

- [BrowseNodeLookup \(p. 194\)](#)

Response Elements

The following table describes the elements returned by NewReleases.

- [Actors \(p. 310\)](#) for the Video search index
- [Artist \(p. 311\)](#) for the Music search index
- [ASIN \(p. 311\)](#)
- [Authors \(p. 311\)](#) for the Books search index
- [ProductGroup \(p. 322\)](#)
- [Title \(p. 325\)](#)
- [TopItemSet \(p. 325\)](#)

NewReleases also returns the elements that all response groups return, which is described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of NewReleases.

- None

Child Response Group

The following response groups are child response groups of NewReleases.

- None

Sample REST Use Case

The following request uses the NewReleases response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=BrowseNodeLookup&  
BrowseNodeId=4229&  
ResponseGroup=NewReleases&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by NewReleases.

```
<TopItemSet>
  <Type>NewReleases</Type>
  <TopItem>
    <ASIN>0553576399</ASIN>
    <Title>Distraction</Title>
    <ProductGroup>Book</ProductGroup>
    <Author>Bruce Sterling</Author>
  </TopItem>
  <TopItem>
    ....
    ....
  </TopItem>
</TopItemSet>
```

OfferFull

The OfferFull response group returns comprehensive information about an offer. OfferFull is a parent response group that returns the results of the [Offers \(p. 271\)](#) response group.

Note

This response group is not returned for Amazon Kindle digital books. An Amazon Kindle ASIN can be verified through the Binding, Format, and ProductTypeName response elements.

The OfferFull response group returns a single offer for each item condition. For example, if a book is available in the conditions New, Used, and Collectible, an OfferFull request will return a total of three offers, one for New, one for Used, and one for Collectible.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by OfferFull.

- [Amount \(p. 311\)](#)
- [Availability \(p. 311\)](#)
- [Condition \(p. 313\)](#)
- [CurrencyCode \(p. 314\)](#)
- [FormattedPrice \(p. 316\)](#)
- [IsEligibleForPrime \(p. 317\)](#)
- [IsEligibleForSuperSaverShipping \(p. 317\)](#)
- [MoreOffersUrl \(p. 320\)](#)
- [Name \(p. 320\)](#)
- [OfferListingId \(p. 321\)](#)
- [TotalCollectible \(p. 325\)](#)
- [TotalNew \(p. 325\)](#)
- [TotalOfferPages \(p. 325\)](#)
- [TotalOffers \(p. 325\)](#)

- [TotalRefurbished \(p. 326\)](#)
- [TotalUsed \(p. 326\)](#)

OfferFull also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of OfferFull.

- None

Child Response Group

The following response groups are child response groups of OfferFull.

- [Offers \(p. 271\)](#)

Sample REST Use Case

The following request uses the OfferFull response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
Condition=All&  
SearchIndex=Books&  
Title=Harry%20Potter&  
ResponseGroup=OfferFull&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by OfferFull.

```
<Item>  
  <ASIN>0439682584</ASIN>  
  <OfferSummary>  
    <LowestNewPrice>  
      <Amount>2580</Amount>  
      <CurrencyCode>USD</CurrencyCode>  
      <FormattedPrice>$25.80</FormattedPrice>  
    </LowestNewPrice>  
    <LowestUsedPrice>  
      <Amount>1599</Amount>  
      <CurrencyCode>USD</CurrencyCode>  
      <FormattedPrice>$15.99</FormattedPrice>  
    </LowestUsedPrice>  
    <LowestCollectiblePrice>
```

```
<Amount>2580</Amount>
<CurrencyCode>USD</CurrencyCode>
<FormattedPrice>$25.80</FormattedPrice>
</LowestCollectiblePrice>
<TotalNew>40</TotalNew>
<TotalUsed>16</TotalUsed>
<TotalCollectible>3</TotalCollectible>
<TotalRefurbished>0</TotalRefurbished>
</OfferSummary>
<Offers>
<TotalOffers>0</TotalOffers>
<TotalOffers>1</TotalOffers>
<TotalOfferPages>0</TotalOfferPages>
<TotalOfferPages>1</TotalOfferPages>
<Offer>
<Merchant>
<Name>Amazon.com</Name>
</Merchant>
<OfferAttributes>
<Condition>New</Condition>
</OfferAttributes>
<OfferListing>
<OfferListingId>cMVuIFx8kiYSgRIJXiCzKZZyy1
wztVSAYV8vCo2OxHS8L9SB7lwho8fK6CxYkmdDPy8thFzm30Y%3D</OfferListingId>
<Price>
<Amount>2580</Amount>
<CurrencyCode>USD</CurrencyCode>
<FormattedPrice>$25.80</FormattedPrice>
</Price>
<Availability>Usually ships in 24 hours</Availability>
<IsEligibleForSuperSaverShipping>1</IsEligibleForSuperSaverShipping>
<IsEligibleForPrime>1</IsEligibleForPrime>
</OfferListing>
</Offer>
</Offers>
</Item>
```

OfferListings

The OfferListings response group returns the OfferListings for items returned in the response. The values returned are similar to those returned by the Offers response group minus the values returned by the OfferSummary response group. OfferListings returns shipping options, including IsEligibleForSuperSavingShipping which specifies if the item qualifies for super saver shipping.

Note

This response group is not returned for Amazon Kindle digital books. An Amazon Kindle ASIN can be verified through the Binding, Format, and ProductTypeName response elements.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes and shows the parentage of the elements returned by OfferListings. In the Ancestry column, the elements on the left side of a slash mark are the parents of the elements on the right side of the slash mark.

- [Amount \(p. 311\)](#)
- [Availability \(p. 311\)](#)
- [Code \(p. 313\)](#)
- [Condition \(p. 313\)](#)
- [CurrencyCode \(p. 314\)](#)
- [FormattedPrice \(p. 316\)](#)
- [IsEligibleForPrime \(p. 317\)](#)
- [IsEligibleForPrimeFreeDigitalVideo \(p. 317\)](#)
- [IsEligibleForSuperSaverShipping \(p. 317\)](#)
- [MoreOffersUrl \(p. 320\)](#)
- [Name \(p. 320\)](#)
- [OfferListingId \(p. 321\)](#)
- [TotalOfferPages \(p. 325\)](#)
- [TotalOffers \(p. 325\)](#)

OfferListings also returns the elements that all response groups return, which described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of OfferListings.

- None

Child Response Group

The following response groups are children response groups of OfferListings.

- None

Sample REST Use Case

The following request uses the OfferListings response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
Keywords=sports&  
ResponseGroup=OfferListings&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by OfferListings.

```

<Item>
  <ASIN>B000AYGDIO</ASIN>
  <Offers>
    <TotalOffers>3</TotalOffers>
    <TotalOfferPages>1</TotalOfferPages>
    <Offer>
      <OfferAttributes>
        <Condition>New</Condition>
      </OfferAttributes>
      <OfferListing>
        <OfferListingId>
          pBmLD7%2F4J7zqIMjOLpQI5pkO774zf%2Bsvrzbg4JR92xLmC%
        </OfferListingId>
        <Price>
          <Amount>24700</Amount>
          <CurrencyCode>USD</CurrencyCode>
          <FormattedPrice>$247.00</FormattedPrice>
        </Price>
        <Availability>Usually ships in 1-2 business days</Availability>
        <IsEligibleForSuperSaverShipping>0</IsEligibleForSuperSaverShipping>

        <IsEligibleForPrime>0</IsEligibleForPrime>
      </OfferListing>
    </Offer>
  </Offers>
</Item>
```

Offers

The Offers response group is a parent response group that returns the contents of the [OfferSummary \(p. 275\)](#) response group. It also returns, by default, offer listing information. The Offers response group can take Condition and MerchantID as an optional input parameter. By default, the offer for Buy Box winner is returned, which is an item listed in the box on an item's detail page that enables the customer to add the item to a shopping cart.

Note

The Offers response group returns a single offer for each item condition. For example, if a book is available in the conditions New, Used, and Collectible, an Offers request will return a total of three offers, one for New, one for Used, and one for Collectible.

The are two valid values for MerchantId: Amazon or All. Use Amazon to get only the Amazon offers for an item. Use All (the default value) to return all offers. The Offers response group ignores a MerchantID value of Featured or Featured Buy Box Winner and returns the default value, All.

The following table shows the expected behavior of the Offers response group for various input parameters:

Optional Input Parameters		Offer(s) Data Returned by the API
(Default)	(Default)	Buy Box Winner if it exists [OR] Lowest priced new offer.
(Default)	Amazon	Amazon offer

Optional Input Parameters		Offer(s) Data Returned by the API
(Default)	Any other value	Error: invalid value
Used, New, Refurbished, or Collectible	(Default) or All	Lowest priced offer for given condition
Used, New, Refurbished, or Collectible	Amazon	Amazon offer
Used, New, Refurbished, or Collectible	Any other value	Error: invalid value
All	(Default)	Lowest priced offer for each condition
All	Amazon	Amazon offer
(Default)	Any other value	Error: invalid value

Note

This response group is not returned for Amazon Kindle digital books. An Amazon Kindle ASIN can be verified through the `Binding`, `Format`, and `ProductTypeName` response elements.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by Offers.

- [Amount \(p. 311\)](#)
- [Availability \(p. 311\)](#)
- [Condition \(p. 313\)](#)
- [CurrencyCode \(p. 314\)](#)
- [FormattedPrice \(p. 316\)](#)
- [IsEligibleForPrime \(p. 317\)](#)
- [IsEligibleForPrimeFreeDigitalVideo \(p. 317\)](#)
- [IsEligibleForSuperSaverShipping \(p. 317\)](#)
- [LoyaltyPoints \(p. 319\)](#)
- [MoreOffersUrl \(p. 320\)](#)
- [Name \(p. 320\)](#)
- [OfferListingId \(p. 321\)](#)
- [TotalCollectible \(p. 325\)](#)
- [TotalNew \(p. 325\)](#)
- [TotalOfferPages \(p. 325\)](#)
- [TotalOffers \(p. 325\)](#)
- [TotalRefurbished \(p. 326\)](#)

- [TotalUsed \(p. 326\)](#)

`Offers` also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Loyalty Points

In the JP locale only, loyalty points are returned. Loyalty points are used to encourage patronage and stimulate sales.

Parent Response Group

The following response groups are parent response groups of `Offers`.

- [OfferFull \(p. 267\)](#)

Child Response Group

The following response groups are child response groups of `Offers`.

- [OfferSummary \(p. 275\)](#)

Sample REST Use Case

The following request uses the `Offers` response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
Condition=All&  
SearchIndex=Books&  
Title=Harry%20Potter&  
ResponseGroup=Offers&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by `Offers`. This example returns two offers, one in New condition, and one in Used condition.

```
<Item>  
  <ASIN>047061529X</ASIN>  
  <OfferSummary>  
    <LowestNewPrice>  
      <Amount>1025</Amount>  
      <CurrencyCode>USD</CurrencyCode>  
      <FormattedPrice>$10.25</FormattedPrice>  
    </LowestNewPrice>  
    <LowestUsedPrice>
```

```
<Amount>1110</Amount>
<CurrencyCode>USD</CurrencyCode>
<FormattedPrice>$11.10</FormattedPrice>
</LowestUsedPrice>
<TotalNew>37</TotalNew>
<TotalUsed>12</TotalUsed>
<TotalCollectible>0</TotalCollectible>
<TotalRefurbished>0</TotalRefurbished>
</OfferSummary>
<Offers>
    <TotalOffers>2</TotalOffers>
    <TotalOfferPages>1</TotalOfferPages>
    <MoreOffersUrl>http://www.amazon.com/gp/offer-listing/047061529X/?SubscriptionId=AKIAI44QH8DHBEXAMPLE&ie=UTF8&tag=adrpik-20&creative=386001&camp=2025&linkCode=xm2 </MoreOffersUrl>
    <Offer>
        <OfferAttributes>
            <Condition>New</Condition>
        </OfferAttributes>
        <OfferListing>
            <OfferListingId>6vZH%2FR4dOoabV7sTSv3vC0Np5xK1c8MKOhA15HYbCIJhx
            OLLJw1O2AM6mLYyVhjnI8s2gMkx7yq%2F%2BEC7yKPWkQTqhVhFBeUDm71EdpaYwrXpppf
            A1lyPzyQFkOuK6MsK8aLPSNSWVs%3D </OfferListingId>
            <Price>
                <Amount>1025</Amount>
                <CurrencyCode>USD</CurrencyCode>
                <FormattedPrice>$10.25</FormattedPrice>
            </Price>
            <AmountSaved>
                <Amount>974</Amount>
                <CurrencyCode>USD</CurrencyCode>
                <FormattedPrice>$9.74</FormattedPrice>
            </AmountSaved>
            <PercentageSaved>49</PercentageSaved>
            <Availability>Usually ships in 1-2 business days</Availability>
            <AvailabilityAttributes>
                <AvailabilityType>now</AvailabilityType>
                <MinimumHours>24</MinimumHours>
                <MaximumHours>48</MaximumHours>
            </AvailabilityAttributes>
            <IsEligibleForSuperSaverShipping>0</IsEligibleForSuperSaverShipping>
            <IsEligibleForPrime>1</IsEligibleForPrime>
        </OfferListing>
    </Offer>

    <Offer>
        <OfferAttributes>
            <Condition>Used</Condition>
        </OfferAttributes>
        <OfferListing>
            <OfferListingId>uXULLeu7rH5t3ogkZJ%2Bd11tWCsdsj5kHhjoscRF1D1GuBuDw
            Cyrz0XyR%2BTEOJO7PgpfwLjtX4ojhbXeHZgM0Br4DiWsPhNZTduzvYC8zLgG0zle%2FgYii
            uuR0wTyKqssY6ncHyvjZK1A%3D </OfferListingId>
            <Price>
                <Amount>1110</Amount>
                <CurrencyCode>USD</CurrencyCode>
                <FormattedPrice>$11.10</FormattedPrice>
            </Price>
        </OfferListing>
    </Offer>

```

```
<AmountSaved>
  <Amount>889</Amount>
  <CurrencyCode>USD</CurrencyCode>
  <FormattedPrice>$8.89</FormattedPrice>
</AmountSaved>
<PercentageSaved>44</PercentageSaved>
<Availability>Usually ships in 1-2 business days</Availability>
<AvailabilityAttributes>
  <AvailabilityType>now</AvailabilityType>
  <MinimumHours>24</MinimumHours>
  <MaximumHours>48</MaximumHours>
</AvailabilityAttributes>
<IsEligibleForSuperSaverShipping>0</IsEligibleForSuperSaverShipping>
<IsEligibleForPrime>1</IsEligibleForPrime>
</OfferListing>
</Offer>
</Offers>
</Item>
```

OfferSummary

The `OfferSummary` response group returns the number of offer listings and the lowest price for each condition type for each item in the response. Condition types are New, Used, Collectible, and Refurbished. For example, this response group returns the lowest price for each Condition:

- New item
- Used item
- Collectible item
- Refurbished item

Individual offer listings are not returned. The `OfferSummary` is dependent only on the ASIN parameter and is not affected by the [MerchantId \(p. 320\)](#) or [Condition \(p. 313\)](#) parameters (i.e. the `OfferSummary` will always be the same for a given ASIN independent of other parameters).

Note

This response group is not returned for Amazon Kindle digital books. An Amazon Kindle ASIN can be verified through the `Binding`, `Format`, and `ProductTypeName` response elements.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by `OfferSummary`.

- [Amount \(p. 311\)](#)
- [CurrencyCode \(p. 314\)](#)
- [FormattedPrice \(p. 316\)](#)

- TotalCollectible (p. 325)
- TotalNew (p. 325)
- TotalRefurbished (p. 326)
- TotalUsed (p. 326)

OfferSummary also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of OfferSummary.

- Offers (p. 271)

Child Response Group

The following response groups are child response groups of OfferSummary.

- None

Sample REST Use Case

The following request uses the OfferSummary response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=B000A3UB20&  
ResponseGroup=OfferSummary&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by OfferSummary.

```
<OfferSummary>  
  <LowestNewPrice>  
    <Amount>801</Amount>  
    <CurrencyCode>USD</CurrencyCode>  
    <FormattedPrice>$8.01</FormattedPrice>  
  </LowestNewPrice>  
  <LowestUsedPrice>  
    <Amount>799</Amount>  
    <CurrencyCode>USD</CurrencyCode>  
    <FormattedPrice>$7.99</FormattedPrice>  
  </LowestUsedPrice>  
  <TotalNew>45</TotalNew>  
  <TotalUsed>20</TotalUsed>
```

```
<TotalCollectible>0</TotalCollectible>
<TotalRefurbished>0</TotalRefurbished>
</OfferSummary>
```

PromotionSummary

The PromotionSummary response group returns summary information about a promotion, including the type of promotion, the promotion ID, eligibility requirements, and text that describes the specifics of the promotion.

PromotionSummary must be used with one of the following response groups:

- Large
- OfferFull
- Offers

An error is returned if ProductDetails is not accompanied by one of these response groups.

Promotion Types

Promotion types are returned by the Category element and include:

- **ForEachQuantityXGetQuantityFreeX**—For a specified number of items, you receive additional items for free. For example, buy six dozen eggs and get a dozen eggs free.
- **BuyAmountXGetSimpleShippingFreeX**—For a specified dollar amount, you receive free shipping. For example, spend \$25 and your item is shipped free of charge.
- **BuyAmountXGetAmountOffX**—For a specified dollar amount, you receive a discounted price. For example, spend \$25 and get a \$5 discount.

Relevant Operations

Operations that can use this response group include:

- ItemLookup
- ItemSearch
- SimilarityLookup

Response Elements

The following table describes and shows the parentage of the elements returned by PromotionSummary.

With the exception of Promotions, Promotion, and Summary, the ancestry of all elements in the table are Offers/Offer/Promotions/Promotion/Summary.

- [BenefitDescription \(p. 312\)](#)
- [Category \(p. 313\)](#)
- [EligibilityRequirementDescription \(p. 315\)](#)
- [Promotion \(p. 322\)](#)
- [PromotionId \(p. 322\)](#)
- [Promotions \(p. 322\)](#)
- [Summary \(p. 325\)](#)

- [TermsAndConditions \(p. 325\)](#)

PromotionSummary also returns the elements that all response groups return, which described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of PromotionSummary.

- None

Child Response Group

The following response groups are children response groups of PromotionSummary.

- None

Sample REST Use Case

The following request uses the PromotionSummary response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=B000AQSMPO&  
IdType=ASIN&  
ResponseGroup=Offers,PromotionSummary&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by PromotionSummary.

```
<Promotions>  
  <Promotion>  
    <Summary>  
      <PromotionId>A2QIQTNOFYRK5N</PromotionId>  
      <Category>BuyAmountXGetAmountOffX</Category>  
      <EligibilityRequirementDescription>Save $25.00 when you spend $125.00 or  
more on Kitchen & Housewares or Bed & Bath products offered by Amazon.com. Enter  
code AUGSAVER at checkout.</EligibilityRequirementDescription>  
      <BenefitDescription>Save $25.00 when you spend $125.00 or more on Kitchen  
& Housewares or Bed & Bath products offered by Amazon.com. Enter code AUGSAVER  
at checkout.</BenefitDescription>  
      <TermsAndConditions><STRONG>To receive the Best Value discount:</STRONG>  
<OL> <LI>Add $125 of qualifying Kitchen & Housewares or Bed & Bath products  
to your Shopping Cart via the <STRONG>Add to Shopping Cart</STRONG> button on  
each respective product information page. <LI>At checkout, enter the promotional  
code and click the <STRONG>Apply</STRONG> button. <LI>The amount of the Best
```

Value savings (\$25) will be reflected on the final order checkout page. If you remove any of the participating promotion items from your Shopping Cart or violate any of the terms and conditions listed below, the promotion will be invalid, and the discount will be removed from the order. If you return any of the items involved in the promotion, the discount previously applied to the order will be subtracted from the return credit.
Terms and Conditions: Promotional offer valid for a limited time only. Items placed in Shopping Cart overnight may not be eligible for promotion at time of checkout. Promotion applies only to qualifying items displaying the offer message on their product information pages. Items that do not display the offer message do not qualify, regardless of the nature of the item. All All Clad, Calphalon, Capresso, Emerilware, Le Creuset, Orrefors, Kosta Boda, J.A. Henckels, Wusthof, Riedel, Marquis, Shun, Komachi, Waterford, Weber, Tassimo, Margaritaville, DKNY, Tommy Hilfiger, Nautica, Oscar by Oscar de Larenta, Cath Kidston, Raymond Waites, and KitchenAid Proline products are excluded from this offer. All Furniture & Decor and Outdoor Living products are excluded from this offer. Applies only to products sold by the merchant indicated in the promotional offer message. Does not apply to items sold by other merchants on the Amazon.com Web site. For example, where the promotional offer applies to items offered by Amazon.com, items offered by other merchants on the Amazon.com Web site (e.g. Land's End) do not qualify. Does not apply to any products purchased in Amazon.com's other sites, including Amazon.co.uk, Amazon.de, Amazon.fr, or Amazon.co.jp, or in Marketplace, zShops, or Auctions. Unless the offer message indicates otherwise, the promotional offer applies to the lowest priced qualifying item(s) Applies only when all qualifying and benefit products in the promotion are purchased in one order. Applies only to complete orders shipping to a single address that meet all other promotional requirements. Offer good while supplies last. No substitutions or rain checks. Offer must be redeemed through the Shopping Cart. Does not apply to orders placed with 1 Click. Shipping and handling charges apply to all products, including bonus items. Offer may not be combined with other offers, including promotional certificates. Void where prohibited. </TermsAndConditions>

</Summary>
</Promotion>
</Promotions>

RelatedItems

The *RelatedItems* response group returns items related to an item specified in an [ItemLookup \(p. 197\)](#) request. For example, related items could be all of the Unbox episodes in a TV season that are sold separately, or all of the MP3Download tracks on an MP3 album.

The data returned for *RelatedItems* is limited to ASINs and ItemAttributes. This remains true even if you add additional response groups, such as Large, that would otherwise return additional data.

The relationship between items is unidirectional. One item is the parent and one item is the child. Items, however, can have multiple children or multiple parents for a given relationship type.

The way in which the items are related is specified by the *RelationshipType* parameter. This parameter is required when you use the *RelatedItems* response group. Some values include Episode, Season, Tracks, and Variation. For a list of all relationship types, go to the [ItemLookup \(p. 197\)](#) page.

The relationship type is usually named after the child item in the relationship. For example, an MP3 Track is related to an MP3 album and the type of relationship is Tracks. In this relationship, the album is the parent. If you did an *ItemLookup* for an MP3 Track and requested *RelatedItems* using Tracks as the

RelationshipType, you would receive the parent album (or albums) for that Track. Conversely, looking up an album using Tracks as the *RelationshipType* returns the list of Tracks on that album.

Each `ItemLookup` request can return, at most, 10 related items. To return additional items, use the `RelatedItemsPage` parameter. For example, a value of 2 returns the second set of 10 related items.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)

Response Elements

The following elements are returned by `RelatedItems`:

- [ItemAttributes \(p. 318\)](#)
- [RelatedItems \(p. 323\)](#)

`RelatedItems` also returns the elements that all response groups return. For more information, go to [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of `RelatedItems`.

- None

Child Response Group

The following response groups are child response groups of `RelatedItems`.

- None

Sample REST Use Case

The following request iluses the `RelatedItems` response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
RelationshipType=Tracks&  
ItemId=B0013D8EQK&  
ItemType=ASIN&  
ResponseGroup=RelatedItems,Small&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by RelatedItems.

```
<Item>
<ASIN>B0013D8EQK</ASIN>
<ItemAttributes>
<Creator Role="Primary Contributor">Johnny Cash</Creator>
<Manufacturer>Columbia/Legacy</Manufacturer>
<ProductGroup>Digital Music Album</ProductGroup>
<Title>At San Quentin</Title>
</ItemAttributes>
<RelatedItems>
<Relationship>Children</Relationship>
<RelationshipType>Tracks</RelationshipType>
<RelatedItemCount>31</RelatedItemCount>
<RelatedItemPageCount>4</RelatedItemPageCount>
<RelatedItemPage>1</RelatedItemPage>
<RelatedItem>
<Item>
<ASIN>B0013D4KJK</ASIN>
<ItemAttributes>
<Creator Role="Primary Contributor">Johnny Cash</Creator>
<Manufacturer>Columbia/Legacy</Manufacturer>
<ProductGroup>Digital Music Track</ProductGroup>
<Title>The Long Black Veil/Give My Love To Rose (Live)</Title>
</ItemAttributes>
</Item>
</RelatedItem>
<RelatedItem>
<Item>
<ASIN>B0013D7VG4</ASIN>
<ItemAttributes>
<Creator Role="Primary Contributor">Johnny Cash</Creator>
<Manufacturer>Columbia/Legacy</Manufacturer>
<ProductGroup>Digital Music Track</ProductGroup>
<Title>Folsom Prison Blues (Live)</Title>
</ItemAttributes>
</Item>
</RelatedItem>
...
</RelatedItems>
</Item>
```

Request

The Request response group returns all of the parameters and their values that were submitted in a request. Use this information to debug requests.

All Product Advertising API operations return this response group by default. There can be up to 10 parameters in each request.

Relevant Operations

Operations that can use this response group include:

- All Product Advertising API operations use the Request response group by default.

Response Elements

The following table describes the elements returned by Request. These elements are returned in all Product Advertising API responses because the Request response group is a default response group for all Product Advertising API operations.

- [Code \(p. 313\)](#)
- [IsValid \(p. 318\)](#)
- [Message \(p. 320\)](#)
- [Name \(p. 320\)](#)
- [RequestId \(p. 323\)](#)
- [UserAgent \(p. 327\)](#)

Parent Response Group

The following response groups are parent response groups of Request.

- None

Child Response Group

The following response groups are child response groups of Request.

- None

Sample REST Use Case

The following shows the request parameters and values.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
SearchIndex=Books&  
Title=Harry%20Potter&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by Request.

```
<OperationRequest>  
  <HTTPHeaders>  
    <Header Name="UserAgent" Value="Mozilla/4.0 (compatible; MSIE 6.0; Windows  
NT 5.1; SV1; FunWebProducts; .NET CLR 1.1.4322; .NET CLR 2.0.50727)" />  
  </HTTPHeaders>  
  <RequestId>00KE1E5MWR4KXX0V1WYD</RequestId>  
  <Arguments>
```

```
<Argument Name="SearchIndex" Value="Books" />
<Argument Name="Service" Value="AWSECommerceService" />
<Argument Name="Title" Value="Harry Potter" />
<Argument Name="Operation" Value="ItemSearch" />
<Argument Name="AWSAccessKeyId" Value="[VALUE]" />
<Argument Name="AssociateTag" Value="[VALUE]" />
<Argument Name="Version" Value="2013-08-01" />
</Arguments>
<RequestProcessingTime>
  0.3419508934021
</RequestProcessingTime>
</OperationRequest>
<Items>
  <Request>
    <IsValid>True</IsValid>
    <ItemSearchRequest>
      <SearchIndex>Books</SearchIndex>
      <Title>Harry Potter</Title>
    </ItemSearchRequest>
  </Request>
</Items>
```

Reviews

The reviews response group returns the URL to an iframe that contains customer reviews. You can embed the iframe on any web page to display the response content. Only the iframe URL is returned in the request.

Important

Each iframe URL is valid for 24 hours. If the iframe URL expires, you will receive a 403 Forbidden error code.

For more information about reviews, see [Getting Customer Reviews \(p. 132\)](#).

Each customer review contains the following:

- Rating
- Summary
- Date of review
- Text of the review

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following elements are returned by Reviews:

- [IFrameURL \(p. 317\)](#)

Reviews also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of Reviews.

- None

Child Response Group

The following response groups are child response groups of Reviews.

- None

Sample REST Use Case

The following request uses the Reviews response parameter.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=0316067938&  
ResponseGroup=Reviews&  
TruncateReviewsAt="256"&  
IncludeReviewsSummary="False"&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by Reviews.

```
<ItemLookupResponse>  
  <OperationRequest>  
    <RequestId>00KE1E5MWR4KXX0V1WYD</RequestId>  
    <Arguments>  
      <Argument Name="Service" Value="AWSECommerceService" />  
      <Argument Name="AWSAccessKeyId" Value="[VALUE]" />  
      <Argument Name="AssociateTag" Value="[VALUE]" />  
      <Argument Name="Operation" Value="ItemLookup" />  
      <Argument Name="SearchIndex" Value="Books" />  
      <Argument Name="ItemId" Value="0316067938" />  
      <Argument Name="IncludeReviewsSummary" Value="False" />  
      <Argument Name="TruncateReviewsAt" Value="256" />  
      <Argument Name="ResponseGroup" Value="Reviews" />  
      <Argument Name="Version" Value="2013-08-01" />  
    </Arguments>  
    <RequestProcessingTime>  
      0.3419508934021  
    </RequestProcessingTime>
```

```
</OperationRequest>
<Items>
  <Item>
    <Request>
      <ASIN>0316067938</ASIN>
      <Customer Reviews>
        <IFrameURL>http://www.amazon.com/reviews/iframe?akid=[AWS Access Key ID]&asin=0316067938&exp=2011-08-01T17%3A54%3A07Z&linkCode=xm2&summary=0&tag=ws&truncate=256&v=2&sig=[Signature]</IFrameURL>
      </CustomerReviews>
    </Request>
  </Item>
</Items>
</ItemLookupResponse>
```

SalesRank

The [SalesRank \(p. 285\)](#) response group returns the sales rank for each item in the response. One is the highest rating, while a large number means the item is not as popular. Sales rank is per Search Index, so it is possible to have several items in one response ranked the same.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by SalesRank.

- [ASIN \(p. 311\)](#)
- [SalesRank \(p. 323\)](#)
- [TotalPages \(p. 326\)](#)
- [TotalResults \(p. 326\)](#)

SalesRank also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of SalesRank.

- None

Child Response Group

The following response groups are child response groups of SalesRank.

- None

Sample REST Use Case

The following request uses the SalesRank response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=0976925524&  
ResponseGroup=SalesRank&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by SalesRank.

```
<Item>  
  <ASIN>0976925524</ASIN>  
  <SalesRank>68</SalesRank>  
</Item>
```

SearchBins

The SearchBins response group groups the items returned by [ItemSearch \(p. 185\)](#) into bins. For example, a set of bins can be a set of price ranges for a product. In the case of women's shoes, you might have a bin that returns ASINs for shoes that cost between \$0 and \$50, a second bin for shoes that cost \$50 to \$100, and a third bin for shoes that cost more than \$100. The basis on which the items are split into bins is specified by the [NarrowBy \(p. 286\)](#) attribute in the SearchBinSet tag. To refine the search, you make repeated requests using the [NarrowBy \(p. 286\)](#) values.

`ItemSearch` returns the first 25 bins of results.

For more information about search bins, see [Using Search Bins to Find Items \(p. 85\)](#).

Availability

US locale only.

NarrowBy

The basis on which the items are split into bins is specified by the [NarrowBy \(p. 286\)](#) attribute in the [SearchBinSet \(p. 323\)](#) tag. In the following example, the [NarrowBy \(p. 286\)](#) attribute shows that the bins are based on price range:

```
<SearchBinSet NarrowBy="PriceRange">
```

For another product category, the [NarrowBy \(p. 286\)](#) attribute might be different, for example:

```
<SearchBinSet NarrowBy="BrandName">
```

You cannot specify [NarrowBy \(p. 286\)](#) values nor can you specify the values they encompass. When SearchBins is included as a Response Group in a request, ItemSearch automatically divides the ItemSearch results into bins.

NarrowBy Values

NarrowBy values include:

- Subject—BrowseNode IDs of all topics related to items returned by ItemSearch. For example, searching for books about dogs returns, in the Subject bins, BrowseNodes for "Home & Garden," "Animal Care & Pets," "Dogs," and "Educational."
- BrandName—Brands, such as Levi's, Reebok, and Nike, that create the item. Use the name of a brand to filter out similar items made by other companies.
- PriceRange—Minimum and maximum prices for a bin of items. Use the minimum and maximum price values in each bin to filter out items outside of the price range you want.
- PercentageOff—Percentage off required on items returned. Use the values returned with MinPercentageOff key to refine the results

All Search Indices return the Subject bin. Most return all of the bins.

NarrowBy Values by Search Index

The following list shows which NarrowBy values are returned by each Search Index:

- Apparel—Subject, BrandName, PriceRange, PercentageOff
- Appliances—Subject, BrandName, PriceRange, PercentageOff
- ArtsAndCrafts—Subject, BrandName, PriceRange, PercentageOff
- Automotive—Subject, BrandName, PriceRange, PercentageOff
- Baby—Subject, BrandName, PriceRange, PercentageOff
- Beauty—Subject, BrandName, PriceRange, PercentageOff
- Blended—Not supported
- Books—Subject
- Classical—Subject
- DigitalMusic—Subject
- DVD—Subject
- Electronics—Subject, BrandName, PriceRange
- GourmetFood—Subject, BrandName, PriceRange
- Grocery—Subject, BrandName, PriceRange
- HealthPersonalCare—Subject, BrandName, PriceRange, PercentageOff
- HomeGarden—Subject, BrandName, PriceRange, PercentageOff
- Industrial—Subject, BrandName
- Jewelry—Subject, PriceRange, PercentageOff
- KindleStore—Subject
- Kitchen—Subject, BrandName, PriceRange, PercentageOff
- Magazines—Subject
- Miscellaneous—BrandName, PriceRange, PercentageOff
- MobileApps—Subject, PriceRange

- MP3Downloads—Subject
- Music—Subject
- MusicalInstruments—Subject, BrandName, PriceRange, PercentageOff
- MusicTracks—Subject
- OfficeProducts—Subject, BrandName, PriceRange, PercentageOff
- OutdoorLiving—Subject, BrandName, PriceRange, PercentageOff
- PCHardware—Subject, BrandName, PriceRange
- PetSupplies—Subject, BrandName, PriceRange, PercentageOff
- Photo—Subject, BrandName, PriceRange
- Shoes—Subject, BrandName, PriceRange, PercentageOff
- Software—Subject, PriceRange, PercentageOff
- SportingGoods—Subject, BrandName, PriceRange, PercentageOff
- Tools—Subject, BrandName, PriceRange, PercentageOff
- Toys—Subject, BrandName, PriceRange, PercentageOff
- UnboxVideo—Subject, PriceRange
- VHS—Subject
- Video—Subject
- VideoGames—Subject, BrandName, PriceRange
- Watches—Subject, BrandName, PriceRange, PercentageOff
- Wireless—Subject, PriceRange, PercentageOff
- WirelessAccessories—Subject, PriceRange, PercentageOff

Element Tags in a Bin

The element tags in a bin vary according to the bin. For example, in bins based on price, the elements and Name values are:

```
<BinName>
<BinItemCount>
<BinParameter>
  <Name>MinimumPrice</Name>
  <Value>
<BinParameter>
  <Name>MaximumPrice</Name>
  <Value>
```

The elements show the minimum and maximum price for items in that bin, and the number of items in that bin.

Other NarrowBy values use other element tags in their bins. For example, when NarrowBy is “Brand,” the element tags and Name values are:

```
<Bin>
<BinName>
<BinItemCount>
<BinParameter>
  <Name>Brand</Name>
  <Value>
```

You cannot specify the element tags returned in a bin.

Drilling Down

You can take the values in a bin and add them to the `ItemSearch` query to filter out of the response items that are outside of that bin. To narrow the search results to shirts that cost between \$0 and \$25, for example, you would add the following Name and its corresponding value as additional parameters in the original `ItemSearch` request:

```
&MinimumPrice=0  
&MaximumPrice=2499
```

You could narrow the search results even further by adding an additional parameter to the query. For example, you could specify

```
&Brand=Levi's
```

The response would then only include shirts by Levi's that cost under \$25. You could continue to drill down by adding additional parameters to the request.

Notice that in each case the results were narrowed down by adding the value tagged with Name and its corresponding value to the original `ItemSearch` request.

Relevant Operations

Operations that can use this response group include:

- [ItemSearch \(p. 185\)](#)

Response Elements

The following table describes the elements returned by SearchBins.

- [BinItemCount \(p. 312\)](#)
- [BinName \(p. 312\)](#)
- [Name \(p. 320\)](#)
- [SearchBinSets \(p. 323\)](#)

SearchBins also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of SearchBins.

- None

Child Response Group

The following response groups are child response groups of SearchBins.

- None

Sample REST Use Case

The following request uses the SearchBins response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
Condition=All&  
SearchIndex=SportingGoods&  
Keywords=Glove&  
ResponseGroup=SearchBins&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by SearchBins. Notice that the NarrowBy value is PriceRange.

```
<Item>  
  <ASIN>B00005R2GR</ASIN>  
</Item>  
<Item>  
  <ASIN>B00076ZDV8</ASIN>  
</Item>  
<Item>  
  <ASIN>B00092FEEG</ASIN>  
</Item>  
<Item>  
  <ASIN>B000ADTP1W</ASIN>  
</Item>  
<SearchBinSets>  
  <SearchBinSet NarrowBy="PriceRange">  
    <Bin>  
      <BinName>$25-$49</BinName>  
      <BinItemCount>316</BinItemCount>  
      <BinParameter>  
        <Name>MinimumPrice</Name>  
        <Value>2500</Value>  
      </BinParameter>  
      <BinParameter>  
        <Name>MaximumPrice</Name>  
        <Value>4999</Value>  
      </BinParameter>  
    </Bin>  
  </SearchBinSet>  
  <Bin>  
    <BinName>$0-$24</BinName>  
    <BinItemCount>280</BinItemCount>  
    <BinParameter>  
      <Name>MinimumPrice</Name>  
      <Value>0</Value>  
    </BinParameter>
```

```
<BinParameter>
<Name>MaximumPrice</Name>
<Value>2499</Value>
</BinParameter>
</Bin>
</SearchBinSet>

<SearchBinSet NarrowBy="BrandName">
<Bin>
<BinName>Rawlings</BinName>
<BinItemCount>71</BinItemCount>
<BinParameter>
<Name>Brand</Name>
<Value>Rawlings</Value>
</BinParameter>
</Bin>
<Bin>
<BinName>Body Glove</BinName>
<BinItemCount>53</BinItemCount>
<BinParameter>
<Name>Brand</Name>
<Value>Body Glove</Value>
</BinParameter>
</Bin>
</SearchBinSet>

<SearchBinSet NarrowBy="Subject">
<Bin>
<BinName>Categories</BinName>
<BinItemCount>860</BinItemCount>
<BinParameter>
<Name>BrowseNode</Name>
<Value>3375301</Value>
</BinParameter>
</Bin>
<Bin>
<BinName>Baseball</BinName>
<BinItemCount>311</BinItemCount>
<BinParameter>
<Name>BrowseNode</Name>
<Value>3395731</Value>
</BinParameter>
</Bin>
</SearchBinSet>
```

Similarities

The Similarities response group returns titles and ASINs of five items that are similar to the item specified in the request. This response group is often used with [ItemLookup \(p. 197\)](#).

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)

- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by Similarities.

- [ASIN \(p. 311\)](#)
- [Title \(p. 325\)](#)
- [TotalPages \(p. 326\)](#)
- [TotalResults \(p. 326\)](#)

Similarities also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of Similarities.

- None

Child Response Group

The following response groups are child response groups of Similarities.

- None

Sample REST Use Case

The following request uses the Similarities response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKey=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
Condition=All&  
SearchIndex=Blended&  
Keywords=Mustang&  
Merchant=All&  
ResponseGroup=Similarities  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by Similarities.

```
<SimilarProduct>  
  <ASIN>B00004GJVO</ASIN>  
  <Title>Minor Move</Title>  
</SimilarProduct>
```

Small

The Small response group returns basic information about each item in a response. The information includes the item's ASIN, title, product group, and author.

This response group is often used with [ItemLookup \(p. 197\)](#), [ItemSearch \(p. 185\)](#), or [SimilarityLookup \(p. 203\)](#) to return basic information about the items in the response.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by Small.

- [Actor \(p. 310\)](#)
- [Artist \(p. 311\)](#)
- [ASIN \(p. 311\)](#)
- [Author \(p. 311\)](#)
- [CorrectedQuery \(p. 313\)](#)
- [Creator \(p. 314\)](#)
- [Director \(p. 314\)](#)
- [Keywords \(p. 318\)](#)
- [Manufacturer \(p. 319\)](#)
- [Message \(p. 320\)](#)
- [ProductGroup \(p. 322\)](#)
- [Role \(p. 323\)](#)
- [Title \(p. 325\)](#)
- [TotalPages \(p. 326\)](#)
- [TotalResults \(p. 326\)](#)

Small also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of Small.

- [Large \(p. 254\)](#)
- [Medium \(p. 259\)](#)

Child Response Group

The following response groups are child response groups of Small.

- None

Sample REST Use Case

The following request uses the Small response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKey=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
Condition=All&  
SearchIndex=Blended&  
Keywords=Mustan&  
Merchant=All&  
ResponseGroup=Small&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by Small.

```
<CorrectedQuery>  
  <Keywords>mustang</Keywords>  
  <Message>We found no matches for (keywords=Mustan). Below are results for  
(keywords=mustang).</Message>  
</CorrectedQuery>  
<Item>  
  <ASIN>B0002Y69UO</ASIN>  
<ItemAttributes>  
  <Actor>America's Favorite Cars</Actor>  
  <ProductGroup>DVD</ProductGroup>  
  <Title>America's Favorite Cars - The Complete Mustang 40th An  
niversary</Title>  
</ItemAttributes>  
</Item>
```

Notice that the [Keywords](#) (p. 318) value, Mustan, was considered misspelled. The search results were for the word, Mustang, instead.

TopSellers

The TopSellers response group returns the ASINs and titles of the 10 best sellers within a specified browse node.

Availability

This response group is available in all locales.

Relevant Operations

Operations that can use this response group include:

- [BrowseNodeLookup \(p. 194\)](#)

Response Elements

The following table describes the elements returned by TopSellers.

- [Actor \(p. 310\)](#) for the Video search index
- [Artist \(p. 311\)](#) for the Music search index
- [ASIN \(p. 311\)](#)
- [Authors \(p. 311\)](#) for the Books search index
- [ProductGroup \(p. 322\)](#)
- [Title \(p. 325\)](#)
- [TopItemSet \(p. 325\)](#)

TopSellers also returns the elements that all response groups return, which is described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of TopSellers.

- None

Child Response Group

The following response groups are child response groups of TopSellers.

- None

Sample REST Use Case

The following request uses the TopSellers response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=BrowseNodeLookup&  
BrowseNodeId=20&  
ResponseGroup=TopSellers&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by TopSellers.

```
<TopItemSet>  
  <Type>TopSellers</Type>  
  <TopItem>
```

```
<ASIN>0553576399</ASIN>
<Title>Distraction</Title>
<ProductGroup>Book</ProductGroup>
<Author>Bruce Sterling</Author>
</TopItem>
<TopItem>
    ....
    ....
</TopItemSet>
```

Tracks

The Tracks response group returns the title and number of each track on each CD in the response. For example, you could use [ItemLookup \(p. 197\)](#) to return Tracks information about a specified CD.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by Tracks.

- [Number \(p. 321\)](#)
- [TotalPages \(p. 326\)](#)
- [TotalResults \(p. 326\)](#)
- [Track \(p. 326\)](#)

Tracks also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of Tracks.

- None

Child Response Group

The following response groups are child response groups of Tracks.

- None

Sample REST Use Case

The following request uses the Tracks response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
Condition=All&  
SearchIndex=Blended&  
Keywords=GodSmack&  
Merchant=All&  
ResponseGroup=Tracks&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by Tracks.

```
<Item>  
  <ASIN>B000EXOAAO</ASIN>  
  <Tracks>  
    <Disc Number="1">  
      <Track Number="1">Livin In Sin</Track>  
      <Track Number="2">Speak</Track>  
      <Track Number="3">The Enemy</Track>  
      <Track Number="4">Shine Down</Track>  
      <Track Number="5">Hollow</Track>  
      <Track Number="6">No Rest For The Wicked</Track>  
      <Track Number="7">Bleeding Me</Track>  
      <Track Number="8">Voodoo Too</Track>  
      <Track Number="9">Temptation</Track>  
      <Track Number="10">Mama</Track>  
      <Track Number="11">One Rainy Day</Track>  
    </Disc>  
  </Tracks>  
</Item>
```

Variations

The Variations response group is a parent response group that returns the contents of the VariationSummary response group plus other variation details, such as item attributes, offers, and offer listings for each variation in the response.

Variation Dimensions

A variation is a child ASIN. The parent ASIN is an abstraction of the children items. For example, a shirt is a parent ASIN and parent ASINs cannot be sold. A child ASIN would be a blue shirt, size 16, sold by MyApparelStore. This child ASIN is one of potentially many variations. The ways in which variations differ are called dimensions. In the preceding example, size and color are the dimensions. Parent ASINs therefore return two related elements:

- VariationDimensions
- VariationDimension

For example:

```
<VariationDimensions>
  <VariationDimension>ClothingSize</VariationDimension>
  <VariationDimension>Color</VariationDimension>
</VariationDimensions>
```

The values returned by these elements are the dimensions listed in the child ASIN's response, for example:

```
<Item>
  ...
  <ItemAttributes>
    ...
  </ItemAttributes>
  <VariationAttributes>
    <VariationAttribute>
      <Name>Color</Name>
      <Value>Black</Value>
    </VariationAttribute>
    <VariationAttribute>
      <Name>ClothingSize</Name>
      <Value>Large</Value>
    </VariationAttribute>
  </VariationAttributes>

  ...
</Item>
```

The following lists some of the Product Advertising API variation dimensions:

- GemType (string)
- HandOrientation (string)
- HardwarePlatform (string)
- PackageQuantity (nonNegativeInteger)
- ItemDimensions/Length (DecimalWithUnits)
- ItemDimensions/Width (DecimalWithUnits)
- LensColor (string)
- MetalType (string)
- Model (string)
- NumberOfLicenses (string)
- OperatingSystem (string)
- ProductTypeSubcategory (string)
- ScentName (string) StyleName (string)
- TotalDiamondWeight (DecimalWithUnits)
- TotalGemWeight (DecimalWithUnits)

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)

- [SimilarityLookup \(p. 203\)](#)

Response Elements

The following table describes the elements returned by Variations.

- [Amount \(p. 311\)](#)
- [ASIN \(p. 311\)](#)
- [CurrencyCode \(p. 314\)](#)
- [FormattedPrice \(p. 316\)](#)

Variations also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of Variations.

- None

Child Response Group

The following response groups are child response groups of Variations.

- [VariationSummary \(p. 307\)](#)

Sample REST Use Case

The following request uses the Variations response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
SearchIndex=Apparel&  
Keywords=Hooded%20Short%20Down%20Jacket&  
ResponseGroup=Variations&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by Variations. This response shows that the ASIN in the response has 10 variations.

```
<ItemSearchRequest>  
  <Keywords>Hooded Short Down Jacket</Keywords>  
  <ResponseGroup>Variations</ResponseGroup>  
  <SearchIndex>Apparel</SearchIndex>
```

```
</ItemSearchRequest>
</Request>
<Item>
  <ASIN>B000CCIIT6</ASIN>
</Item>
<Item>
  <ASIN>B000CCIITQ</ASIN>
</Item>
<Item>
  <ASIN>B000CCONPO</ASIN>
</Item>
<Item>
  <ASIN>B000CCMQ1W</ASIN>
</Item>
<Item>
  <ASIN>B000CDDRJ6</ASIN>
</Item>
<Item>
  <ASIN>B000CCIISC</ASIN>
</Item>
<Item>
  <ASIN>B000CCIIRI</ASIN>
</Item>
<Item>
  <ASIN>B000BVA9AE</ASIN>
</Item>
<Item>
  <ASIN>B000CCMQFS</ASIN>
</Item>
<Item>
  <ASIN>B000CDDRJG</ASIN>
</Item>
</Items>
```

VariationImages

The VariationImages response group displays different image variations of the same item in four sizes: small, medium, large, and swatch. Swatch images are the smallest size. VariationImages is used when there are variations of an item for sale. For example, a shirt for sale might come in four colors. If VariationImages is used as the response group, the shirt will be displayed in 16 images: four image sizes of each of the four colors.

Parent ASINs do not always have images associated with them. Child ASINs should have images.

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)

Response Elements

The following table describes the elements returned by VariationImages. The elements on the left side of a slash mark are the parents of the elements on the right side of the slash mark.

- [Height \(p. 316\)](#)

- [LargeImage \(p. 319\)](#)
- [MediumImage \(p. 320\)](#)
- [SmallImage \(p. 324\)](#)
- [SwatchImage \(p. 325\)](#)
- [ThumbnailImage \(p. 325\)](#)
- [TinyImage \(p. 325\)](#)
- [URL \(p. 327\)](#)
- [Width \(p. 328\)](#)

VariationImages also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of VariationImages.

- None

Child Response Group

The following response groups are child response groups of VariationImages.

- None

Sample REST Use Case

The following request uses the VariationImages response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=0239409223&  
ResponseGroup=VariationImages  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by VariationImages.

```
<ImageSets>  
  <ImageSet Category="primary">  
  
    <SmallImage>  
      <URL>  
        http://images.amazon.com/images/P/B9999999A.01._SCTHUMBZZZ_.jpg  
      </URL>  
      <Height Units="pixels">60</Height>
```

```
<Width Units="pixels">60</Width>
</SmallImage>

<MediumImage>
<URL>
http://images.amazon.com/images/P/B9999999A.01._SCMZZZZZZ_.jpg
</URL>
<Height Units="pixels">140</Height>
<Width Units="pixels">140</Width>
</MediumImage>

<LargeImage>
<URL>
http://images.amazon.com/images/P/B9999999A.01._SCLZZZZZZ_.jpg
</URL>
<Height Units="pixels">500</Height>
<Width Units="pixels">500</Width>
</LargeImage>
</ImageSet>
<ImageSet Category="variant">

<SmallImage>
<URL>
http://images.amazon.com/images/P/B9999999A.01.PT01._SCTHUMBZZZ_.jpg
</URL>
<Height Units="pixels">48</Height>
<Width Units="pixels">60</Width>
</SmallImage>

<MediumImage>
<URL>
http://images.amazon.com/images/P/B9999999A.01.PT01._SCMZZZZZZ_.jpg
</URL>
<Height Units="pixels">120</Height>
<Width Units="pixels">150</Width>
</MediumImage>

<LargeImage>
<URL>
http://images.amazon.com/images/P/B9999999A.01.PT01._SCLZZZZZZ_.jpg
</URL>
<Height Units="pixels">400</Height>
<Width Units="pixels">500</Width>
</LargeImage>

</ImageSet>
```

VariationMatrix

The VariationMatrix response group returns, for a given parent ASIN, the variation dimension name and value of each child ASIN. If a returned item does not have variations, the VariationMatrix response group will not return any data. For example, rings vary by ring size. RingSize, therefore, is the dimension related to this item. The VariationMatrix response group would only return, in this example, <RingSize> elements. The VariationMatrix response group returns nothing if the returned items do not have variations. For more information, see, [Returning Variation Dimension Information Only \(p. 37\)](#).

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)

Response Elements

The following table describes the elements returned by VariationMatrix.

- [ClothingSize \(p. 313\)](#)
- [Color \(p. 313\)](#)
- [HardwarePlatform \(p. 316\)](#)
- [ItemDimensions/Length \(p. 319\)](#)
- [ItemDimensions/Width \(p. 328\)](#)
- [MaterialType \(p. 319\)](#)
- [MetalType \(p. 320\)](#)
- [Model \(p. 320\)](#)
- [OperatingSystem \(p. 321\)](#)
- [PackageQuantity \(p. 321\)](#)
- [ProductTypeSubcategory \(p. 322\)](#)
- [Size \(p. 324\)](#)
- [VariationDimension \(p. 327\)](#)

VariationMatrix also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of VariationMatrix.

- None

Child Response Group

The following response groups are child response groups of VariationMatrix.

- None

Sample REST Use Case

The following request uses the VariationMatrix response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&
```

```
ItemId=B0008G23PQ&
Condition=All&
ResponseGroup=VariationMatrix&
Version=2013-08-01
```

Sample Response Snippet

The following response snippet shows the elements returned by VariationMatrix.

```
<Item>
  <ASIN>B0008G23PQ</ASIN>
  <Variations>
    <VariationDimensions>
      <VariationDimension>ClothingSize</VariationDimension>
      <VariationDimension>Color</VariationDimension>
    </VariationDimensions>
  <Item>
    <ASIN>B0008EOA9U</ASIN>
    <VariationAttributes>
      <VariationAttribute>
        <Name>Color</Name>
        <Value>Grey</Value>
      </VariationAttribute>
      <VariationAttribute>
        <Name>ClothingSize</Name>
        <Value>29W x 30L</Value>
      </VariationAttribute>
    </VariationAttributes>
  </Item>
  <Item>
    <ASIN>B0008EO9J6</ASIN>
    <VariationAttributes>
      <VariationAttribute>
        <Name>Color</Name>
        <Value>Navy</Value>
      </VariationAttribute>
      <VariationAttribute>
        <Name>ClothingSize</Name>
        <Value>29W x 30L</Value>
      </VariationAttribute>
    </VariationAttributes>
  </Item>
</Item>
```

VariationOffers

The VariationOffers response group retrieves the offers for the children of a parent ASIN. VariationOffers is similar to the Variations response group. However, item attributes are not returned for the individual variations. Variations is the parent response group of VariationOffers.

The only valid values for MerchantId are Amazon and All, which is the default value. For reverse compatibility, the Offers response group treats a MerchantID value of Featured or Featured Buy Box Winner as the default value (All).

The following table shows the expected behavior of the parent and child variation offers for the corresponding MerchantId input parameter:

Merchant ID	Semantics
(Default)	All variations with offers
Amazon	Variations with Amazon offers only
Featured, Featured Buy Box Winner, or All	All variations with offers. The specified MerchantId parameter will not have any effect on the response.
Any other value	Error: invalid value

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)

Response Elements

The following table describes the elements returned by VariationOffers. In the Ancestry column, In the Ancestry column, the elements on the left side of a slash mark are the parents of the elements on the right side of the slash mark.

- [Amount \(p. 311\)](#)
- [ASIN \(p. 311\)](#)
- [Availability \(p. 311\)](#)
- [AvailabilityAttributes \(p. 311\)](#)
- [Condition \(p. 313\)](#)
- [CurrencyCode \(p. 314\)](#)
- [FormattedPrice \(p. 316\)](#)
- [IsEligibleForSuperSaverShipping \(p. 317\)](#)
- [LoyaltyPoints \(p. 319\)](#)
- [MaximumHours \(p. 319\)](#)
- [MinimumHours \(p. 320\)](#)
- [Name \(p. 320\)](#)
- [OfferListingId \(p. 321\)](#)
- [TotalCollectible \(p. 325\)](#)
- [TotalNew \(p. 325\)](#)
- [TotalOfferPages \(p. 325\)](#)
- [TotalOffers \(p. 325\)](#)

VariationOffers also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of VariationOffers.

- [Variations \(p. 307\)](#)

Child Response Group

The following response groups are child response groups of VariationOffers.

- None

Sample REST Use Case

The following request uses the VariationOffers response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemSearch&  
ItemId=B000P4VW1M&  
Condition=All&  
ResponseGroup=VariationOffers&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by VariationOffers.

```
<Item>  
  <ASIN>B000P4VW1M</ASIN>  
  <VariationSummary>  
    <LowestPrice>  
      <Amount>12900</Amount>  
      <CurrencyCode>USD</CurrencyCode>  
      <FormattedPrice>$129.00</FormattedPrice>  
    </LowestPrice>  
    <HighestPrice>  
      <Amount>12900</Amount>  
      <CurrencyCode>USD</CurrencyCode>  
      <FormattedPrice>$129.00</FormattedPrice>  
    </HighestPrice>  
  </VariationSummary>  
  <Variations>  
    <TotalVariations>5</TotalVariations>  
    <TotalVariationPages>1</TotalVariationPages>  
    <Item>  
      <ASIN>B000N637Y2</ASIN>  
      <Offers>  
        <Offer>  
          <Merchant>  
            <Name>Amazon.com</Name>  
          </Merchant>  
          <OfferAttributes>  
            <Condition>New</Condition>  
          </OfferAttributes>  
          <OfferListing>  
            <OfferListingId>ct%2FbvUK%2F9aRAxMyfX4EnsglyWuwchEGTLsR7xZwnr
```

```
Rt9AwU%2Fe4yKOeVDEIdVSBUhrBFoR4VZWR1aUBZX%2FhzSDQ%3D%3D</OfferListingId>
    <Price>
        <Amount>12900</Amount>
        <CurrencyCode>USD</CurrencyCode>
        <FormattedPrice>$129.00</FormattedPrice>
    </Price>
    <AmountSaved>
        <Amount>28428</Amount>
        <CurrencyCode>USD</CurrencyCode>
        <FormattedPrice>$284.28</FormattedPrice>
    </AmountSaved>
    <PercentageSaved>69</PercentageSaved>
    <Availability>In stock soon. Order now to get in line. First come,
first served.</Availability>
    <AvailabilityAttributes>
        <AvailabilityType>unknown</AvailabilityType>
        <MinimumHours>672</MinimumHours>
        <MaximumHours>1008</MaximumHours>
    </AvailabilityAttributes>
    <IsEligibleForSuperSaverShipping>1</IsEligibleForSuperSaverShipping>
</OfferListing>
</Offer>
</Offers>
</Item>
```

VariationSummary

The VariationSummary response group provides the lowest price, highest price, lowest sale price, and highest sale price for all child ASINs in a response.

Parent ASINs do not have offers, but their children do. For example, you cannot buy a shirt (the parent ASIN) but you can buy a shirt that is a specific color and size (the child ASIN).

Relevant Operations

Operations that can use this response group include:

- [ItemLookup \(p. 197\)](#)
- [ItemSearch \(p. 185\)](#)
- [SimilarityLookup \(p. 203\)](#)

Variation Dimensions

A variation is a child ASIN. The parent ASIN is an abstraction of the children items. For example, a shirt is a parent ASIN and parent ASINs cannot be sold. A child ASIN of the parent would be a blue shirt, size 16, sold by MyApparelStore. This child ASIN is one of potentially many variations. The ways in which variations differ are called dimensions. In the preceding example, size and color are the dimensions. The parent ASIN therefore returns two related elements:

- [VariationDimensions](#)
- [VariationDimension](#)

For example:

```
<VariationDimensions>
  <VariationDimension>ClothingSize</VariationDimension>
  <VariationDimension>Color</VariationDimension>
</VariationDimensions>
```

The values returned by these elements are the dimensions listed in the child ASIN's response, for example:

```
<Item>
  <ItemAttributes>
    ...
  </ItemAttributes>
  <VariationAttributes>
    <VariationAttribute>
      <Name>Color</Name>
      <Value>Black</Value>
    </VariationAttribute>
    <VariationAttribute>
      <Name>ClothingSize</Name>
      <Value>Large</Value>
    </VariationAttribute>
  </VariationAttributes>
  ...
</Item>
```

Response Elements

The following table describes the elements returned by VariationSummary.

- [Amount \(p. 311\)](#)
- [CurrencyCode \(p. 314\)](#)
- [FormattedPrice \(p. 316\)](#)

VariationSummary also returns the elements that all response groups return, as described in [Elements Common to All Response Groups \(p. 309\)](#).

Parent Response Group

The following response groups are parent response groups of VariationSummary.

- [Variations \(p. 297\)](#)

Child Response Group

The following response groups are child response groups of VariationSummary.

- None

Sample REST Use Case

The following request uses the VariationSummary response group.

```
http://webservices.amazon.com/onca/xml?  
Service=AWSECommerceService&  
AWSAccessKeyId=[AWS Access Key ID]&  
AssociateTag=[Associate ID]&  
Operation=ItemLookup&  
ItemId=B00006XYAA&  
ResponseGroup=VariationSummary&  
Version=2013-08-01  
&Timestamp=[YYYY-MM-DDThh:mm:ssZ]  
&Signature=[Request Signature]
```

Sample Response Snippet

The following response snippet shows the elements returned by VariationSummary.

```
<Item>  
  <ASIN>B00006XYAA</ASIN>  
  <VariationSummary>  
    <LowestPrice>  
      <Amount>1450</Amount>  
      <CurrencyCode>USD</CurrencyCode>  
      <FormattedPrice>$14.50</FormattedPrice>  
    </LowestPrice>  
    <HighestPrice>  
      <Amount>1750</Amount>  
      <CurrencyCode>USD</CurrencyCode>  
      <FormattedPrice>$17.50</FormattedPrice>  
    </HighestPrice>  
  </VariationSummary>  
</Item>
```

Response Elements Common to All Response Groups

The following table describes and shows the parentage of the elements returned by all response groups.

Response Element	Definition
ASIN	An token distributed by Amazon that uniquely identifies an item. Type: String Ancestry: Item/ASIN
Code	Error code if there is an error Type: Integer Ancestry: Errors/Error/Code
IsValid	Is True if the request is valid Type: String Ancestry: Request/IsValid

Response Element	Definition
Message	Error message that corresponds with error code Type: String Ancestry: Errors/Error/Message
Name	Name of a parameter in the request Type: String Ancestry: Arguments/Argument/Name
RequestId	Unique number that identifies the request Type: String Ancestry: OperationRequest/RequestId
TotalPages	Number of pages found. There are up to ten items per page. Type: String Ancestry: List/TotalPages
TotalResults	The total number of items found. Up to ten are returned per request. . By default, the first ten items are returned. Type: String Ancestry: List/TotalResults
UserAgent	The name and version of the Web browser Type: String Ancestry: OperationRequest/UserAgent
Value	A value of a parameter in the request Type: String Ancestry: Arguments/Argument/Value

Response Elements

This chapter provides a description of all response elements. In the Ancestry paragraphs, the elements on the left side of a slash mark are the parents of the elements on the right side of the slash mark.

Response Element	Definition
About	Describes the seller. Ancestry: Seller/About
AboutMe	Information a customer supplies about themselves. Ancestry: Seller/About
Actor	Actor associated with the item. Ancestry: ItemAttributes
AdditionalName	For baby registries, this field is used for the name of the other parent. Ancestry: List/AdditionalName

Product Advertising API Developer Guide
Response Elements

Response Element	Definition
AlternateVersion	Container for AlternateVersion information, including ASIN, Title, and Binding. Ancestry: None
Amount	Price in terms of the lowest currency denomination, for example, pennies. The price, in terms of the lowest currency denomination, for example, pennies, of an item in the cart. The total price, in terms of the lowest currency denomination, of one or more of the same item in the Saved For Later area. If, for example, Saved For Later contains two orders of the same book, this amount would be the sum total of those two books. Ancestry: OfferSummary/LowestCollectiblePrice OfferSummary/LowestNewPrice OfferSummary/LowestRefurbished-Price/Amount Offers/Offer/OfferListing/Price/Amount Offers/Offer/OfferListing/SalePrice/Amount Transaction/TransactionItems/TransactionItem/UnitPrice VariationSummary/LowestPrice VariationSummary/HighestSalePrice
Artist	Artist associated with the item. Ancestry: ItemAttributes
ASIN	A positive integer distributed by Amazon that uniquely identifies an item. ASIN of the new release, item, or similar item Ancestry: Item/ASIN CartItem SavedForLaterItem Cart/NewReleases/NewRelease NewReleases/NewRelease/ASIN OtherCategoriesSimilarProduct/ASIN SimilarProduct SimilarViewedProduct
AspectRatio	The ratio of an item's length to its width. Ancestry: ItemAttributes
AudienceRating	Audience rating for a movie. The rating suggests the age for which the movie is appropriate. The rating format varies by locale. Ancestry: ItemAttributes
AudioFormat	Format, such as MP3, of the audio media. Ancestry: ItemAttributes/Languages/Language
Author	Author associated with the item. Ancestry: ItemAttributes
Availability	How soon the item can be shipped. Ancestry: Offers/Offer/OfferListing/Availability
AvailabilityAttributes	Container for availability information, including AvailabilityType, MaximumHours, and MinimumHours. Ancestry: Offers/Offer/OfferListing
Benefit	Container for information relating to a promotional benefit. Ancestry: Benefits

Product Advertising API Developer Guide
Response Elements

Response Element	Definition
Benefits	Container for one or more Benefit elements. Ancestry: None
BenefitType	Specifies the type of promotion, for example, Free, FixedAmountOff, PercentOff, and TieredFixedAmountOff. Ancestry: Benefits/Benefit/BenefitType
BenefitDescription	Text that describes the promotion. Ancestry: Offers/Offer/Promotions/Promotion/Summary
Bin	Container for Bin elements. Children: BinItemCount, BinName, BinParameter
Binding	Typically but not always similar to the product category. Ancestry: ItemAttributes
BinItemCount	Number of items in a bin. Ancestry: SearchBinSets/SearchBinSet/Bin/BinItemCount
BinName	Name of the bin. Ancestry: SearchBinSets/SearchBinSet/Bin/BinName
BinParameter	Container for the BrowseNodeId and it's value. Ancestry: Bin
Brand	An item's brand. Ancestry: ItemAttributes, Large, Medium
BrowseNodeId	A positive integer that uniquely identifies a parent product category. Ancestry: BrowseNode/Ancestors/BrowseNode BrowseNode/Children/BrowseNode
CartId	A positive integer that uniquely identifies a cart. All operations on a cart must include this value, which is generated by CartCreate. Ancestry: Cart
CartItem	A parent element for many child elements, including CartItemId, Quantity, Title, ProductGroup, Price, and ItemTotal. Ancestry: Cart/CartItems
CartItemID	A positive integer that uniquely identifies an item in a cart or in the Saved For Later area. You must use this value to modify cart items or Saved For Later items. Other identifiers, such as ASINs, do not work. Ancestry: CartItem SavedForLaterItem
CartItems	A parent element for many child elements, including SubTotal, and CartItem. Ancestry: Cart

Product Advertising API Developer Guide
Response Elements

Response Element	Definition
Category	Specifies the kind of promotion. Valid values include ForEachQuantityXGetQuantityFreeX, BuyAmountXGetSimpleShippingFreeX, and BuyAmountXGetAmountOffX. For more information, see Promotion Types (p. 277) . Ancestry: Offers/Offer/Promotions/Promotion/Summary
CEROAgeRating	The Computer Entertainment Rating Organization (CERO) sets ratings, advertising guidelines, and online privacy principles for video and computer games in the United States and Canada. The value of CEROAgeRating specifies whether a video or software game (search indices Software and VideoGames) is suitable for everyone to play. Ancestry: ItemAttributes
ClothingSize	Size of clothes. Ancestry: ItemAttributes
Code	Number that uniquely identifies an error. Ancestry: Errors/Error
Collection	Container for items that are part of a collection. Ancestry: None
CollectionItem	An item that is part of a collection of items. Ancestry: None
CollectionParent	The parent item has an ASIN but it cannot be purchase. It names the collection. CollectionItems are children of the Collection Parent and can be purchased. Ancestry: None
Color	Color Ancestry: ItemAttributes
Comment	Comment typically about the purpose of the list. Ancestry: List
ComponentType	Specifies what the promotion applies to, for example, Shipping, ItemPrice, Subtotal. Ancestry: Benefits/Benefit
Condition	Specifies the condition of the item, such as new, used, collectible, or refurbished. Ancestry: SellerListing/Condition Offers/Offer/OfferAttributes/Condition
CorrectedQuery	A parent element that contains the elements related to a corrected keyword. See Keywords . Ancestry: Items
CouponCombinationType	Specifies the kinds of promotional coupons that can be combined, for example, Unrestricted, Preferential, and Exclusive. Ancestry: None

Product Advertising API Developer Guide
Response Elements

Response Element	Definition
Creator	Creator associated with the item. Ancestry: ItemAttributes
CurrencyAmount	Price in terms of the lowest currency denomination, for example, pennies. Ancestry: EligibilityRequirements/EligibilityRequirement/
CurrencyCode	An abbreviation that specifies the format of the price for the associated locale. Ancestry: Cart/SavedForLaterItems/SubTotal Cart/CartItem/SubTotal SavedForLaterItem/ItemTotal SavedForLaterItem/Price CartItem/Price Cart/SubTotal EligibilityRequirements/EligibilityRequirement/CurrencyAmount Benefits/Benefit/FixedAmount OfferSummary/LowestCollectiblePrice/CurrencyCode OfferSummary/LowestNewPrice/CurrencyCode OfferSummary/LowestRefurbishedPrice/CurrencyCode OfferSummary/LowestUsedPrice/CurrencyCode Offers/Offer/OfferListing/Price/CurrencyCode Offers/Offer/OfferListing/SalePrice/CurrencyCode Transaction/Totals/Total VariationSummary/HighestPrice VariationSummary/LowestSalePrice VariationSummary/HighestSalePrice VariationSummary/LowestPrice
DateAdded	Date the item was added to the list. Ancestry: ListItem/DateAdded
DateCreated	Date the list was created, in the form yyyy-mm-dd. Ancestry: List/DateCreated
Department	Department Ancestry: ItemAttributes
Details	Container for all of the elements that describe a promotion. Ancestry:
Director	Director Ancestry: ItemAttributes
EAN	European Article Number, which is a number that uniquely identifies an item. Ancestry: ItemAttributes
EANList	The container for one or more EANListElement (p. 315) attributes. Ancestry: ItemAttributes/EANList

Product Advertising API Developer Guide
Response Elements

Response Element	Definition
EANListElement	A possible EAN for the ASIN. Ancestry: ItemAttributes/EANList
Edition	Edition Ancestry: ItemAttributes
EditorialReviewIsLinkSuppressed	Boolean value that specifies whether or not the link to the editorial review is displayed. Ancestry: EditorialReview
EISBN	Electronic ISBN number for digital books Ancestry: ItemAttributes
EligibilityRequirement	Container for one or more EligibilityRequirementType elements. Ancestry: EligibilityRequirements
EligibilityRequirementDescription	Specifies the conditions necessary to qualify for the promotion. Ancestry: Offers/Offer/Promotions/Promotion/Summary
EligibilityRequirements	Container for one or more EligibilityRequirement elements. Ancestry: None
EligibilityRequirementType	Specifies the eligibility requirements to qualify for the promotion, for example, MinPurchase, MinQuantity, or ForEachQuantity. Ancestry: EligibilityRequirements/EligibilityRequirement
EndDate	Specifies a date when the item will stop being sold, or the last day and ending time (GMT) of the promotion. Ancestry: SellerListing/EndDate, Offers/Offer/Promotions/Promotion/Summary
EpisodeSequence	Television shows are broadcast in sequence. Each show is called an episode. EpisodeSequence specifies the number of the show in the series of episodes.
ESRBAgeRating	The Entertainment Software Rating Board (ESRB) sets ratings, advertising guidelines, and online privacy principles for video and computer games in the United States and Canada. The value of ESRBAgeRating specifies whether a video or software game (search indices Software and VideoGames) is suitable for everyone to play. Ancestry: ItemAttributes
Feature	An item's feature Ancestry: ItemAttributes
Feedback	Provides customer feedback about the seller. Ancestry:
Fitment	A part that works in (fits into/onto) a car. Children: Bed, BodyStyle, Brakes, DriveType, Engine, Make, MfrBodyCode, Model, Notes, Position, SpringTypes, Steering, Transmission, Trim, Wheelbase, Year Ancestor: FitmentAttributes: FitmentAttribute

Product Advertising API Developer Guide
Response Elements

Response Element	Definition
FitmentAttribute	Container for one or more Fitment elements. Child: Fitment Ancestry: FitmentAttributes
FitmentAttributes	Container for one or more FitmentAttribute elements. Child: FitmentAttribute
FixedAmount	Specifies the price in terms of the lowest currency denomination. Ancestry: Benefits/Benefit
Format	An item's format Ancestry: ItemAttributes
FormattedPrice	The price formatted as it should be displayed to the user. The formatting is specified by CurrencyCode and varies by country. Or, the discounted price formatted for display. Ancestry: EligibilityRequirements/EligibilityRequirement/CurrencyAmount Benefits/Benefit/FixedAmount OfferSummary/LowestUsedPrice/FormattedPrice OfferSummary/LowestCollectiblePrice/FormattedPrice OfferSummary/LowestNewPrice/FormattedPrice Transaction/TransactionItems/TransactionItem/TotalPrice
Genre	Specifies the genre, such as romance, of a digital item. Ancestry: ItemAttributes
GroupClaimCode	An alphanumeric token to use to claim the promotional benefit. Ancestry: None
HardwarePlatform	Hardware Ancestry: Variations/Item/ItemAttributes
HasReviews	Boolean that specifies whether the product has one or more customer reviews. Ancestry: CustomerReviews
HazardousMaterialType	Type of hazardous material that is present in the product. Ancestry: ItemAttributes
Height	Height of an item, package, or image. Ancestry: ImageSets/ImageSet/LargeImage ImageSets/ImageSet/MediumImage ImageSets/ImageSet/SmallImage ImageSets/ImageSet/TinyImage ImageSets/ImageSet/SwatchImage/Height
HMAC	Hash-Based Message Authentication Code (HMAC) is a keyed hash function that is used with a cryptographic hash function such as SHA256 or MD5 that cannot be computed without the key. The HMAC is used to identify and secure a cart. Each cart operation must include this value. Ancestry: Cart

Product Advertising API Developer Guide
Response Elements

Response Element	Definition
IFrameURL	<p>URL to an iframe that contains customer reviews.</p> <p>To embed the iframe on a web page, add the following to your HTML:</p> <pre><iframe src="reviews_iframe_url" /></pre> <p>Note The URL expires in 24 hours. Ancestry: CustomerReviews</p>
Image	An image associated with a seller's item. Ancestry: SellerListing/Image
IsAdultProduct	Indicates if the product is considered to be for adults only. Ancestry: ItemAttributes
IsAutographed	Specifies whether or not the item is autographed. Ancestry: ItemAttributes
ISBN	ISBN number Ancestry: ItemAttributes
IsCategoryRoot	Boolean value that specifies if the browse node is at the top of the browse node tree. Ancestry: BrowseNodes
IsEligibleForPrime	Specifies if an item is eligible for Amazon Prime. If the flag is 1, the item is eligible. If the flag is 0 or missing, the item is not eligible. Ancestry: Offers/Offer/OfferListing Version: 2011-08-01 or greater
IsEligibleForPrimeFreeDigitalVideo	Specifies whether or not an Amazon Instant Video is eligible to view for free with a Prime Subscription. If the flag is 1, the item is eligible. If the flag is 0 or missing, the item is not eligible. Ancestry: Offers/Offer/OfferListing Version: 2013-08-01 or greater
IsEligibleForSuperSaver-Shipping	Specifies whether the item is eligible for super saver shipping. Ancestry: Offers/Offer/OfferListing
IsEligibleForTradeIn	Specifies whether or not the item is eligible for trade-in. Ancestry: ItemAttributes
IsEmailNotifyAvailable	Boolean value, where 1 specifies that an email can be sent when the item is in stock, and 0 specifies when an email cannot be sent. This functionality is not available for all offers.
IsFit	YES, NO, or MAYBE values specify whether or not a part will work in a specified vehicle.

Product Advertising API Developer Guide
Response Elements

Response Element	Definition
IsInBenefitSet	A boolean value. When True, the associated ASIN is what the customer receives as part of the promotion. Ancestry: ItemApplicability
IsInEligibilityRequirementSet	A boolean value. When True, the associated ASIN is what the customer must purchase to qualify for the promotion. Ancestry: ItemApplicability
IsLinkSuppressed	"1" (true) if there were any HTML links in editorial reviews that have been suppressed. Ancestry: EditorialReview
IsMemorabilia	Specifies whether the item is considered memorabilia. Ancestry: ItemAttributes
IsNext	Indicates that there are more vehicle parts to return with higher ASIN values than those already returned.
IsPrevious	Indicates that there are more vehicle parts to return with lower ASIN values than those already returned.
ItemApplicability	Container for ASIN, IsInBenefitSet, and IsInEligibilityRequirementSet. This group of values specifies whether the ASIN is what the customer receives as part of the promotion, whether the ASIN is what the customer needs to purchase to qualify for the promotion, or both. Ancestry: ItemApplicability/ItemApplicability
ItemDimensions	Container for Height, Length, Weight, and Width. Ancestry: ItemAttributes
IssuesPerYear	Number of issues per year, usually, in a subscription. Ancestry: ItemAttributes
IsValid	Boolean value that specifies whether the request syntax was correct. Ancestry: Request
ItemAttributes	Container for many attributes that describe an item. Children: 88 elements.
ItemPartNumber	The item part number. Ancestry: ItemAttributes
Keywords	The (corrected) words used in a search. Ancestry: Items/CorrectedQuery
Label	Label Ancestry: ItemAttributes
Language	Container for the Name, Type, and AudioFormat response elements, which together describe a language. Ancestry: ItemAttributes/Languages

Product Advertising API Developer Guide
Response Elements

Response Element	Definition
Languages	Container for one or more Language response elements. Ancestry: ItemAttributes
LargeImage	Container for a large image Ancestry: ImageSets/ImageSet
LastModified	Date when the list was last modified Ancestry: List/LastModified
LegalDisclaimer	Legal disclaimer Ancestry: ItemAttributes
Length	Length Ancestry: ItemAttributes/ItemDimensions ItemAttributes/Length ItemAttributes
ListItemId	A number that uniquely identifies an item on a list. Ancestry: ListItem/ListItemId
ListPrice	The manufacturer's suggested retail price for a product. Ancestry: ItemAttributes
LoyaltyPoints	In the JP locale only, loyalty points are returned. Loyalty points are used to generate sales. Ancestry: Offers/Offer
LoyaltyPoints	The number of points awarded as part of a purchase. Points translate into rewards. Ancestry: Offers, VariationOffers
Manufacturer	Manufacturer Ancestry: ItemAttributes
ManufacturerMaximumAge	Defines the maximum age in months the user should be to enjoy the use of the item. For example, for a toy targeted at kids from ages 2 to 4, 4 would be the value for the ManufacturerMaximumAge. Ancestry: ItemAttributes
ManufacturerMinimumAge	Defines the minimum age in months the user should be to enjoy the use of the item. For example, for a toy targeted at kids from ages 2 to 4, 2 would be the value for the ManufacturerMinimumAge. Ancestry: ItemAttributes
ManufacturerPartsWarranty-Description	Describes the manufacturer's parts warranty Ancestry: ItemAttributes
MaterialType	The type of material used in the item. Ancestry: ItemAttributes
MaximumHours	The maximum number of hours for which the item might be available. Ancestry: Offers/Offer/OfferListing/AvailabilityAttributes

Product Advertising API Developer Guide
Response Elements

Response Element	Definition
MediaType	The type of media used by a recording device. Ancestry: ItemAttributes
MediumImage	Container for a medium image Ancestry: ImageSets/ImageSet
MerchandisingMessage	MerchandisingMessage contains text entered by a merchant that describes a promotion. Ancestry: None
MerchantId	A positive integer distributed by Amazon that uniquely identifies a merchant. Ancestry: CartItem SavedForLaterItem
Message	Message, or, a description of an error. Ancestry: Items/CorrectedQuery, Errors/Error
MetalType	The type of metal used in the item. Ancestry: ItemAttributes
MinimumHours	The minimum number of hours for which the item is available. Ancestry: Offers/Offer/OfferListing/AvailabilityAttributes
Model	Model Ancestry: ItemAttributes
MoreOffersUrl	The URL where all offers for an item are displayed. The URLs provided on the page are the exact ones that you should use when you link back to Amazon.com. They are tagged with your Associate tag and contain other tracking information to increase your hourly request limit as the sales that you generate increase. Ancestry: OfferFull, OfferListing, Offers
MPN	Manufacturer's part number. Ancestry: ItemAttributes
Name	Name of Search BinParameter, BrowseNode, Language, Variation-Attribute, or Merchant. Ancestry: SearchBinSets/SearchBinSet/Bin/BinParameter, BrowseNode, BrowseNode/Ancestors/BrowseNode, BrowseNode/Children/BrowseNode, ItemAttributes/Languages/Language, VariationAttributes/VariationAttribute, Offers/Offer/Merchant
Nickname	Seller's nickname Ancestry: SellerListing/Seller

Product Advertising API Developer Guide
Response Elements

Number	An attribute in the Disc tag that denotes the CD number in the item, for example, <Disc Number="1">. Items with multiple CDs would have multiple values for Disc, or an attribute in the Track tag that denotes the track number on the CD, for example, <Track Number="9">Now Or Never</Track>. Ancestry: Tracks/Disc Tracks/Disc/Track
NumberOfDiscs	Number of discs an item can hold or use. Ancestry: ItemAttributes
NumberOfIssues	Number of issues in a subscription. Ancestry: ItemAttributes
NumberOfItems	Number of items. Ancestry: ItemAttributes
NumberOfPages	Number of pages. Ancestry: ItemAttributes
NumberOfTracks	Number of recorded track on a CD. Ancestry: ItemAttributes
OccasionDate	Specifies the date of an occasion, such as a wedding or birthday. Ancestry: List/OccasionDate
OfferListingId	A number that uniquely identifies an offer listing. This number represents a sales offer from a specific merchant. Ancestry: Offers/Offer/OfferListing
OperatingSystem	Specifies the name of the operating system on a computer. Ancestry: ItemAttributes, VariationMatrix
OtherCategoriesSimilar-Products	Parent element for Title and ASIN of similar products in other product groups Ancestry: Cart
PackageQuantity	Quantity of items in a package Ancestry: Variations/Item/ItemAttributes/
ParentASIN	A parent ASIN for an item in Saved For Later or the Active areas of a cart.. Ancestry: CartItem SavedForLaterItem
PartBrandBins	Container for one or more Bin elements.
PartBrowseNodeBins	Container for one or more Bin elements.
PartNumber	The part number. Ancestry: ItemAttributes
PartnerName	For wedding registries, this is the name of the bride or groom. Ancestry: List/PartnerName

Product Advertising API Developer Guide
Response Elements

Platform	Operating system. Ancestry: ItemAttributes
Price	A parent element for Amount, CurrencyCode, and FormattedPrice. In this case, the price is for an item in the Active or Saved For Later areas, respectively. Ancestry: CartItem SavedForLaterItem
ProductGroup	The product category an item belongs to. The name of a category, such as sporting goods, to which an item in the cart belongs. The name of a category, such as sporting goods, to which an item in Saved For Later belongs. Ancestry: ItemAttributes/ProductGroup CartItem/ProductGroup SavedForLaterItem
ProductTypeSubcategory	Subcategory of product type Ancestry: Variations/Item/ItemAttributes
Promotion	Container for one or more Details elements. Ancestry: Offers/OfferPromotions
PromotionId	An alphanumeric token that uniquely identifies a promotion. Ancestry: Offers/Offer/Promotions/Promotion/Summary
Promotions	A container for one or more Promotion elements. Ancestry: Offers/Offer
PublicationDate	Publication date. Ancestry: ItemAttributes
Publisher	Publisher Ancestry: ItemAttributes
PurchaseURL	A URL that the customer must use to purchase the items in their cart. Ancestry: Cart
Quantity	The number of a particular item in a cart. Or, the number of items in stock.Or, the number of a particular item in Saved For Later. Ancestry: CartItem, SellerListing SavedForLaterItem
RegionCode	A code that specifies a region. Ancestry: ItemAttributes
RegistryName	Name of a wedding or baby registry. Ancestry: List/RegistryName
RelatedItem	Container for an item that is related to the one specified in the ItemLookup request. Ancestry: RelatedItems Children: Item, ASIN , ItemAttributes

Product Advertising API Developer Guide
Response Elements

RelatedItems	A container for one or more items that are related to the one specified in an ItemLookup request. This value is only returned with the Relateditems response group. Children: Relationship, Relationshiptype, RelatedItemCount, RelatedItemPageCount, RelatedItemPage, RelatedItem.
RelatedItemCount	Specifies the number of related items found. This number, however, can be larger than the actual number returned because not all related items found are available.
RelatedItemPage	Specifies which page of related items has been returned. Ancestry: RelatedItems
RelatedItemPageCount	Specifies the number of pages of related items found. There are up to ten items per page. Ancestry: RelatedItems
Relationship	Child or parent. Relationships are uni-directional: either parent to child, or child to parent. Ancestry: RelatedItems
RelationshipType	Specifies how the related item relates to the specified item in the ItemLookup request, for example, Tracks. For more information, go to the ItemLookup page. Ancestry: RelatedItems
ReleaseDate	Date on which the item was latest released. Items that have been released multiple times have both release dates and original release dates. See ReleaseDate. Ancestry: ItemAttributes
RequestId	A number that uniquely identifies a request Ancestry: OperationRequest
Role	Role Ancestry: ItemAttributes/Creator
RunningTime	The duration of a presentation. Ancestry: ItemAttributes
SalesRank	Indicates how well an item is selling within its product category. The lower the number, the better the item has sold. Ancestry: Item
SavedForLaterItem	A positive integer that uniquely identifies an item in Saved For Later. Ancestry: Cart/SavedForLaterItems Cart
SearchBinSet	A means by which to categorize results, such as price range. Ancestry: SearchBinSets
SearchBinSets	Parent element for SearchBins element. Ancestry: SearchBinSets

Product Advertising API Developer Guide
Response Elements

SeikodoProductCode	The Seikodo catalog number (JP only). Ancestry: ItemAttributes
ShipmentItems	The items in a specific shipment. Ancestry: Transaction/Shipment/Shipmetn
Shipments	The shipments Ancestry: Transaction
SimilarProducts	Parent element for Title and ASIN of similar products in the same product group Ancestry: Cart
SimilarViewedProducts	Parent element for Title and ASIN of similar products in the same product group that have been viewed Ancestry: Cart
Size	Size Ancestry: ItemAttributes
SKU	Stock Keeping Unit (SKU) is a number that uniquely identifies an item. Ancestry: ItemAttributes SellerListing
SmallImage	Container for a small image. Thumbnail and Small images are the same size. Ancestry: ImageSets/ImageSet
Source	Where the review was entered. Typically, this is Amazon.com. It is possible, however, for customers to enter reviews from other sites. Ancestry: EditorialReviews/EditorialReview
StartDate	The date an item goes on sale, or the first day and beginning time (GMT) of the promotion. Ancestry: SellerListing, Offers/Offer/Promotions/Promotion/Summary
StoreId	A number that uniquely identifies the seller's store. Ancestry: SellerListing/Seller
StoreName	The name of the seller's store. Ancestry: SellerListing/Seller
Studio	The name of the studio, such as Warner Brothers, that produced a digital item. Ancestry: ItemAttributes
SubscriptionLength	The duration of a subscription. Ancestry: ItemAttributes

Product Advertising API Developer Guide
Response Elements

Summary	A short summary of the content of the review. Or, a container for all promotion information, including the elements BenefitDescription, Category, EndDate, EligibilityRequirementsDescription, PromotionId, StartDate, and TermsAndConditions. Ancestry: Offers/Offer/Promotions/Promotion
SwatchImage	Container for a swatch image, which is smaller than a Small image. Ancestry: ImageSets/ImageSet
TermsAndConditions	Specifies the terms and conditions of the promotion. Ancestry: Offers/Offer/Promotions/Promotion/Summary
ThumbnailImage	Container for a Thumbnail image. Thumbnail and Small images are the same size. Ancestry: ImageSets/ImageSet
TinyImage	Container for a tiny image Ancestry: ImageSets/ImageSet
Title	Title or the name of the accessory, item, new release, similar products in other product groups, similar products in the same product group, similar products in the same product group that have been viewed Ancestry: ItemAttributes Accessories/Accessory SavedForLaterItem Cart/NewReleases/NewRelease/Title CartItem OtherCategoriesSimilarProduct Similar-Product SimilarViewedProduct TopSellers/TopSeller
TopItem	Container object for information related to ranked responses, including MostGifted, MostWishedFor, TopSellers, and NewReleases. Information contained includes ASIN, Title, ProductGroup, Author, Artist, and Actor. Ancestry: MostGifted, MostWishedFor, TopSellers, NewReleases Children: ASIN, Title, ProductGroup, Actor, Artist, Author
TopItemSet	Container for one or more TopItem elements.
TotalCollectible	The total number of collectible items for sale. Ancestry: OfferSummary
TotalItems	The total number of items found on the list. Each page holds up to ten items. Ancestry: Items/TotalItems
TotalNew	The total number of new items for sale. Ancestry: OfferSummary
TotalOfferPages	Number of pages of offers. By default, the first ten offers are returned in the response. Ancestry: Offers
TotalOffers	Total number of offers. Ancestry: Offers

Product Advertising API Developer Guide
Response Elements

TotalPages	The total number of pages found in a response. Each page holds up to ten items. Type: nonNegativeInteger Ancestry: Items, Fitment
TotalRefurbished	The total number of refurbished items for sale. Ancestry: OfferSummary
TotalResults	Total number of items found by the request. Only ten are returned at a time. Ancestry: Items, List
Totals	Container object for all other Total* elements, including Total, Sub-Total, Tax, ShippingCharges, Promotion, and so forth. Ancestry: Transaction
TotalTimesRead	The number of times a list has been viewed. Ancestry: List
TotalUsed	The total number of used items for sale. Ancestry: OfferSummary
TotalVotes	The total number of review votes cast. Ancestry: CustomerReviews/Review/TotalVotes
Track	Refers to each track on a CD. On a music CD, each track corresponds to a song. Ancestry: Tracks/Disc
TradeInValue	The trade-in value of this item. Ancestry: ItemAttributes
TransactionDate	The date on which the transaction began of the form yyyy-mm-ddThh.mm.ss, for example, 2014-09-24T19:19:27. Ancestry: Transaction
TransactionDateEpoch	The date on which the transaction began in the form of epoch seconds. Ancestry: Transaction
TransactionId	A series of dash-separated integers, such as 111-222-333, that uniquely identify a transaction. Ancestry: Transaction
TransactionItem	Container for all of the information related to a specified transaction. Ancestry: Transaction
TransactionItemId	A string that uniquely identifies a transaction item. Ancestry: Transaction/TransactionItems/TransactionItem
TransactionItems	Container for TransactionItem Ancestry: Transaction

Product Advertising API Developer Guide
Response Elements

Type	Type, depending on the context. If the ancestor is TopItemSet, the Type is the ranking criteria, such as MostGifted. Ancestry: ItemAttributes/Languages/Language, TopItemSet
UPC	Universal Product Code, which is a 12 digit number, 6 of which represents an item's manufacturer. These numbers are translated into a bar code that is printed on an item or its packaging. Ancestry: ItemAttributes SellerListing
UPCList	The container for one or more UPCListElement (p. 327) attributes. Ancestry: ItemAttributes
UPCListElement	A possible UPC for the ASIN. Ancestry: ItemAttributes/UPCList
URL	URL of an image. Ancestry: Variations/Item/ImageSets/ImageSet/LargeImage Variations/Item/ImageSets/ImageSet/SwatchImage Variations/Item/ImageSets/ImageSet/MediumImage Variations/Item/ImageSets/ImageSet/SmallImage
URLEncodedHMAC	A URL-encoded version of the HMAC that can be used directly in a request. Ancestry: Cart
UserAgent	The client application, for example, web browsers, search engine crawlers, mobile phones, screen readers and braille browsers. The text associated with UserAgent contains the application name, version, host operating system, and language. Ancestry: OperationRequest
UserId	An alphanumeric token that uniquely identifies the customer that tagged the entity. Ancestry:
VariationAttribute	Container for a variation name and value. Ancestry: Item/VariationAttributes/
VariationDimension	Container for dimensions Ancestry: Variations/VariationDimensions/
Warranty	Warranty terms Ancestry: ItemAttributes
WEEETaxValue	The Waste Electrical and Electronic Equipment tax for the item. Ancestry: ItemAttributes
Weight	Weight Ancestry: ItemAttributes/ItemDimensions

Width	Width Ancestry: ItemAttributes/ItemDimensions Variations/Item/ImageSets/ImageSet/SwatchImage Variations/Item/ImageSets/ImageSet/LargeImage Variations/Item/ImageSets/ImageSet/SmallImage Variations/Item/ImageSets/ImageSet/MediumImage
Year	Year, for example, 2015.

Locale Reference

Operations are the same for all locales but valid search indices, browse node IDs, sort values, and ItemSearch parameters vary from one locale to another. For more information about general differences, see [Locale Considerations \(p. 180\)](#).

- Search index

Some search operations require a search index to limit the scope of the search. Specifying a browse node in addition to a search index returns more targeted search results. The search index values that can be used in an ItemSearch request vary by locale. For example, the Baby and Beauty search indices are available in the US but not in the UK locale. An error is returned if you use a search index value that is not supported in a locale. The limitations on the use of search indices by locale therefore also limits the use of ItemSearch parameters.

- Browse node ID

These IDs represent the top level browse nodes only, and do not attempt to cover the thousands of possible nodes. For more targeted results, you can use these IDs in a BrowseNodeLookup request to get additional browse node IDs. For more information, see [BrowseNodeLookup \(p. 194\)](#).

- Sort values

The Sort parameter enables you to choose the order of the items in a response. There are many sort values, and availability varies by locale and search index. The majority are not applied unless the Sort parameter is included in the request. There are two sort values, however, that are used by default:

- For ItemSearch requests that do not use the BrowseNode parameter, results are sorted by Relevance.
- For ItemSearch requests that do use the BrowseNode parameter, results are sorted by BestSeller ranking.

- ItemSearch parameters

For a given search index, only some ItemSearch parameters are valid. Also, each locale supports only a subset of all search index values. For example, in the US locale, when SearchIndex is Blended, the only parameter that can be used in an ItemSearch request is Keywords.

Topics

- [Locale Information for the BR Marketplace \(p. 329\)](#)
- [Locale Information for the CA Marketplace \(p. 331\)](#)
- [Locale Information for the CN Marketplace \(p. 345\)](#)
- [Locale Information for the DE Marketplace \(p. 358\)](#)
- [Locale Information for the ES Marketplace \(p. 378\)](#)
- [Locale Information for the FR Marketplace \(p. 394\)](#)
- [Locale Information for the IN Marketplace \(p. 410\)](#)

- [Locale Information for the IT Marketplace \(p. 423\)](#)
- [Locale Information for the JP Marketplace \(p. 437\)](#)
- [Locale Information for the MX Marketplace \(p. 455\)](#)
- [Locale Information for the UK Marketplace \(p. 462\)](#)
- [Locale Information for the US Marketplace \(p. 480\)](#)

Locale Information for the BR Marketplace

BR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Todos os departamentos	All			Availability ItemPage Keywords MaximumPrice MerchantId MinimumPrice
Apps e Jogos	MobileApps	6563510011	-price (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the BR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Livros	Books	7841278011	-price (p. 503) daterank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Condition ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title
Loja Kindle	KindleStore	5308308011	-price (p. 503) daterank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Locale Information for the CA Marketplace

CA Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
All Departments	All			Availability ItemPage Keywords MaximumPrice MerchantId MinimumPrice
Apps & Games	MobileApps	6386372011	-price (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the CA Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Automotive	Automotive	6948389011	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Baby	Baby	3561347011	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the CA Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Beauty	Beauty	6205125011	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Books	Books	927726	daterank (p. 503) inverse-pricerank (p. 503) pricerank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Clothing & Accessories	Apparel	8604904011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Electronics	Electronics	677211011	-price (p. 503) -titlerank (p. 503) price (p. 503) relevancerank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the CA Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Gift Cards	GiftCards	9230167011	-price (p. 503) -release-date (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Grocery & Gourmet Food	Grocery	6967216011	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Health & Personal Care	HealthPersonal-Care	6205178011	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Home & Kitchen	Kitchen	2206276011	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Jewelry	Jewelry	9674384011	-price (p. 503) -release-date (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Kindle Store	KindleStore	2972706011	-price (p. 503) daterank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the CA Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Luggage & Bags	Luggage	6205506011	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Movies & TV	DVD	14113311	salesrank (p. 503) titlerank (p. 503)	Actor AudienceRating Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the CA Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Music	Music	962454	-orig-rel-date (p. 503) -releasedate (p. 503) orig-rel-date (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Artist Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Musical Instruments, Stage & Studio	MusicallInstruments	6916845011	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the CA Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Office Products	OfficeProducts	6205512011	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Patio, Lawn & Garden	LawnAndGarden	6299024011	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the CA Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Pet Supplies	PetSupplies	6291628011	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Shoes & Handbags	Shoes	8604916011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the CA Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Software	Software	3234171	-daterank (p. 503) inverse-pricerank (p. 503) pricerank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Sports & Outdoors	SportingGoods	2242990011	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the CA Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Tools & Home Improvement	Tools	3006903011	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Toys & Games	Toys	6205517011	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the CA Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Video Games	VideoGames	110218011	-titlerank (p. 503) inverse-pricerank (p. 503) pricerank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Watches	Watches	2235621011	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	Blended			Availability ItemPage Keywords

Locale Information for the CN Marketplace

CN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	All			Availability ItemPage Keywords MerchantId
	Appliances	80208071	-launch-date (p. 503) -pct-off (p. 503) -price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Kindle	KindleStore	116088071	-price (p. 503) daterank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the CN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
TBD	GiftCards	311868071	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
TBD	Kitchen	2016126051	-price (p. 503) -release-date (p. 503) pct-off (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the CN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
TBD	MobileApps	146629071	-price (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
TBD	PCHardware	42690071	-price (p. 503) -release-date (p. 503) pct-off (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the CN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	HealthPersonal-Care	852804051	-price (p. 503) -release-date (p. 503) -releasedate (p. 503) -titlerank (p. 503) price (p. 503) release-date (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	MusicalInstruments	2127219051	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the CN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	OfficeProducts	2127222051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	Books	658391051	-price (p. 503) -publica- tion_date (p. 503) -titlerank (p. 503) -unit-sales (p. 503) daterank (p. 503) inverse-pricerank (p. 503) price (p. 503) pricerank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the CN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	PetSupplies	118864071	-launch-date (p. 503) -pct-off (p. 503) -price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	HomeImprovement	1952921051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the CN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Home	2016127051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Neighborhood Sort Title
/	Photo	755653051	-launch-date (p. 503) -pct-off (p. 503) -price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the CN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Apparel	2016157051	-launch-date (p. 503) -pct-off (p. 503) -price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	Baby	42693071	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the CN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Automotive	1947900051	-launch-date (p. 503) -pct-off (p. 503) -price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
/	VideoGames	897416051	-price (p. 503) -release-date (p. 503) -releasedate (p. 503) -titlerank (p. 503) price (p. 503) release-date (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice ReleaseDate Sort Title

Product Advertising API Developer Guide
Locale Information for the CN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Toys	647071051	-price (p. 503) -release-date (p. 503) -releasedate (p. 503) -titlerank (p. 503) price (p. 503) release-date (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	Jewelry	816483051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title
	Electronics	2016117051	-price (p. 503) -release-date (p. 503) -releasedate (p. 503) -titlerank (p. 503) price (p. 503) release-date (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Beauty	746777051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	Software	863873051	-price (p. 503) -release-date (p. 503) -releasedate (p. 503) -titlerank (p. 503) price (p. 503) release-date (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the CN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	SportingGoods	836313051	-price (p. 503) -release-date (p. 503) -titlerank (p. 503) price (p. 503) release-date (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	Watches	1953165051	-price (p. 503) -titlerank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title
	Shoes	2029190051	-launch-date (p. 503) -price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the CN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Music	754387051	-orig-rel-date (p. 503) -price (p. 503) -pricerank (p. 503) -releasedate (p. 503) -titlerank (p. 503) orig-rel-date (p. 503) price (p. 503) pricerank (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Artist Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice ReleaseDate Sort Title
	Video	2016137051	-orig-rel-date (p. 503) -price (p. 503) -pricerank (p. 503) -releasedate (p. 503) -titlerank (p. 503) orig-rel-date (p. 503) price (p. 503) pricerank (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Actor Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Grocery	2127216051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Locale Information for the DE Marketplace

DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Alle Kategorien	All			Availability FutureLaunchDate ItemPage Keywords MerchantId

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Amazon Instant Video	UnboxVideo	3010076031	date-desc-rank (p. 503) popularity-rank (p. 503) price-asc-rank (p. 503) price-desc-rank (p. 503) relevancerank (p. 503) review-rank (p. 503)	Actor AudienceRating Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher SearchIndex Sort Title
Amazon Pantry	Pantry	N/A	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Apps & Spiele	MobileApps	1661650031	-price (p. 503) pmrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Auto & Motorrad	Automotive	78193031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Baby	Baby	357577011	-price (p. 503) price (p. 503) psrank (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Baumarkt	Tools	80085031	-price (p. 503) featured (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Actor Artist AudienceRating Author Availability Brand Composer Conductor Director ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Neighborhood Orchestra Power Publisher ReleaseDate Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Beauty	Beauty	64257031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Bekleidung	Apparel	78689031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Beleuchtung	Lighting	213084031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Bücher	Books	541686	-price (p. 503) -pubdate (p. 503) -publica- tion_date (p. 503) -titlerank (p. 503) -unit-sales (p. 503) inverse-pricerank (p. 503) price (p. 503) pricerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Bürobedarf & Schreibwaren	OfficeProducts	192417031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Computer & Zubehör	PCHardware	569604	-price (p. 503) launch_date (p. 503) price (p. 503) psrank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
DVD & Blu-ray	DVD	547664	-price (p. 503) -titlerank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Actor AudienceRating Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Drogerie & Körperpflege	HealthPersonal-Care	64257031	-price (p. 503) -titlerank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Elektro-Großgeräte	Appliances	931573031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Elektronik & Foto	Electronics	569604	-price (p. 503) -titlerank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Fremdsprachige Bücher	ForeignBooks	54071011	-price (p. 503) -pubdate (p. 503) -publica- tion_date (p. 503) -titlerank (p. 503) -unit-sales (p. 503) inverse-pricer- ank (p. 503) price (p. 503) pricerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title
Games	VideoGames	541708	-date (p. 503) -price (p. 503) -titlerank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Garten	HomeGarden	10925241	-price (p. 503) -titlerank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Geschenkgutscheine	GiftCards	1571257031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Artist Availability Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Haustier	PetSupplies	427727031	-price (p. 503) -price-new-bin (p. 503) price (p. 503) price-new-bin (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Kamera & Foto	Photo	571860	-price (p. 503) -titlerank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Kindle-Shop	KindleStore	530485031	-edition-sales-velocity (p. 503) -price (p. 503) daterank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Klassik	Classical	542676	-price (p. 503) -pubdate (p. 503) -publica-tion_date (p. 503) -released-date (p. 503) -titlerank (p. 503) price (p. 503) pubdate (p. 503) publica-tion_date (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Artist Availability Composer Conductor ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Orchestra Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Koffer, Rucksäcke & Taschen	Luggage	2454119031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Küche & Haushalt	Kitchen	3169011	-price (p. 503) -titlerank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Lebensmittel & Getränke	Grocery	344162031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Musik-CDs & Vinyl	Music	542676	-price (p. 503) -pubdate (p. 503) -publica- tion_date (p. 503) -released- ate (p. 503) -titlerank (p. 503) price (p. 503) pubdate (p. 503) publica- tion_date (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Artist Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Musik-Downloads	MP3Downloads	180529031	-albumrank (p. 503) -artistalbumrank (p. 503) -price (p. 503) -releasedate (p. 503) -runtime (p. 503) -titlerank (p. 503) albumrank (p. 503) artistalbumrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) runtime (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Musikinstrumente & DJ-Equipment	MusicallInstruments	340850031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Schmuck	Jewelry	327473011	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title
Schuhe & Handtaschen	Shoes	362995011	-launch-date (p. 503) -price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Software	Software	542064	-date (p. 503) -price (p. 503) -titlerank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Spielzeug	Toys	12950661	-date (p. 503) -price (p. 503) -titlerank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title

Product Advertising API Developer Guide
Locale Information for the DE Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Sport & Freizeit	SportingGoods	16435121	-price (p. 503) -release-date (p. 503) -titlerank (p. 503) price (p. 503) release-date (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Technik & Wissenschaft	Industrial	5866099031	-price (p. 503) featured (p. 503) price (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Uhren	Watches	193708031	-price (p. 503) -titlerank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Zeitschriften	Magazines	1161660	-price (p. 503) -titlerank (p. 503) -unit-sales (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
	Blended			Availability ItemPage Keywords

Locale Information for the ES Marketplace

ES Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Todos los departamentos	All			Availability ItemPage Keywords MerchantId

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Apps y Juegos	MobileApps	1661651031	-price (p. 503) pmrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Bebé	Baby	1703496031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Belleza	Beauty	6198055031	-price (p. 503) -release-date (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Bricolaje y herramientas	Tools	2454134031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Actor Artist AudienceRating Author Availability Brand Composer Conductor Director ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Neighborhood Orchestra Power Publisher ReleaseDate Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Cheques regalo	GiftCards	3564280031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Artist Availability Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice
Coche y moto	Automotive	1951052031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Deportes y aire libre	SportingGoods	2665403031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Electrónica	Electronics	667050031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Equipaje	Luggage	2454130031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Hogar	Kitchen	599392031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Iluminación	Lighting	3564290031	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Industria y ciencia	Industrial	5866089031	-price (p. 503) featured (p. 503) price (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Informática	PCHardware	667050031	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Instrumentos musicales	Musicallnstruments	3628867031	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Jardín	LawnAndGarden	1571260031	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Joyería	Jewelry	2454127031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title
Juguetes y juegos	Toys	599386031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title
Libros	Books	599365031	-price (p. 503) -pubdate (p. 503) -publica-tion_date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Libros en idiomas extranjeros	ForeignBooks	599368031	-price (p. 503) -pubdate (p. 503) -publica-tion_date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title
Música Digital	MP3Downloads	1748201031	-albumrank (p. 503) -artistalbum-rank (p. 503) -price (p. 503) -releasedate (p. 503) -runtime (p. 503) -titlerank (p. 503) albumrank (p. 503) artistalbum-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) runtime (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the ES Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Música: CDs y vinilos	Music	599374031	-price (p. 503) -released-date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Artist Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Oficina y papelería	OfficeProducts	3628729031	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the ES Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Películas y TV	DVD	599380031	-price (p. 503) -released-date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Actor AudienceRating Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Relojes	Watches	599389031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Ropa y accesorios	Apparel	2846221031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Salud y cuidado personal	HealthPersonal-Care	3677431031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand Director ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the ES Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Software	Software	599377031	-price (p. 503) -released-date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Supermercado	Grocery	6198073031	-price (p. 503) -release-date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Tienda Kindle	KindleStore	818938031	-edition-sales-velocity (p. 503) -price (p. 503) daterank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Videojuegos	VideoGames	599383031	-price (p. 503) -released-date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Zapatos y complementos	Shoes	1571263031	-launch-date (p. 503) -price (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Locale Information for the FR Marketplace

FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Toutes nos boutiques	All			Availability ItemPage Keywords MerchantId

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Animalerie	PetSupplies	1571269031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Applis & Jeux	MobileApps	1661655031	-price (p. 503) pmrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Bagages	Luggage	2454146031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Beauté et Parfum	Beauty	197859031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Bijoux	Jewelry	193711031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Boutique Kindle	KindleStore	672109031	-edition-sales-velocity (p. 503) -price (p. 503) daterank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Boutique chèques-cadeaux	GiftCards	2524128031	-price (p. 503) -reviewrank_authority (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability Brand Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Bricolage	HomeImprovement	590749031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Bébés & Puériculture	Baby	206618031	-price (p. 503) price (p. 503) relevancerank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Chaussures et Sacs	Shoes	248812031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Cuisine & Maison	Kitchen	57686031	-price (p. 503) price (p. 503) relevancerank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
DVD & Blu-ray	DVD	578608	-titlerank (p. 503) amzrank (p. 503) availability (p. 503) salesrank (p. 503) titlerank (p. 503)	Actor AudienceRating Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Epicerie	Grocery	3635789031	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Fournitures de bureau	OfficeProducts	192420031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Gros électroménager	Appliances	908827031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
High-Tech	Electronics	14011561	-price (p. 503) -titlerank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Hygiène et Santé	HealthPersonal-Care	197862031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Informatique	PCHardware	340859031	-price (p. 503) launch_date (p.503) price (p. 503) psrank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Instruments de musique & Sono	MusicallInstruments	340862031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Jardin	LawnAndGarden	3557028031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Jeux et Jouets	Toys	548014	-price (p. 503) -titlerank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title
Jeux vidéo	VideoGames	548014	-date (p. 503) -price (p. 503) -titlerank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Livres anglais et étrangers	ForeignBooks	69633011	-daterank (p. 503) -price (p. 503) -titlerank (p. 503) -unit-sales (p. 503) inverse-pricerank (p. 503) price (p. 503) pricerank (p. 503) publica-tion_date (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title
Livres en français	Books	468256	-daterank (p. 503) -price (p. 503) -titlerank (p. 503) -unit-sales (p. 503) inverse-pricerank (p. 503) price (p. 503) pricerank (p. 503) publica-tion_date (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Logiciels	Software	548012	-date (p. 503) -pricerank (p. 503) -titlerank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Luminaires et Eclairage	Lighting	213081031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Montres	Watches	60937031	-price (p. 503) -titlerank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Musique : CD & Vinyles	Music	537366	-price (p. 503) -pricerank (p. 503) -releasedate (p. 503) -titlerank (p. 503) availability (p. 503) price (p. 503) pricerank (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Artist Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Musique classique	Classical	537366	-price (p. 503) -titlerank (p. 503) inverse-pricerank (p. 503) price (p. 503) pricerank (p. 503) salesrank (p. 503) titlerank (p. 503)	Artist Availability Composer Conductor ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Orchestra Sort Title

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Secteur industriel & scientifique	Industrial	5866110031	-price (p. 503) featured (p. 503) price (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Sports et Loisirs	SportingGoods	325615031	-launch-date (p. 503) -price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title

Product Advertising API Developer Guide
Locale Information for the FR Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Téléchargement de musique	MP3Downloads	206442031	-albumrank (p. 503) -artistalbumrank (p. 503) -price (p. 503) -releasedate (p. 503) -runtime (p. 503) -titlerank (p. 503) albumrank (p. 503) artistalbumrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) runtime (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Vêtements et accessoires	Apparel	340856031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Blended			Availability ItemPage Keywords

Locale Information for the IN Marketplace

IN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
All Departments	All			Availability ItemPage Keywords MerchantId
Baby	Baby	1571275031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the IN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Beauty	Beauty	1355017031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Books	Books	976390031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the IN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Car & Motorbike	Automotive	4772061031	-price (p. 503) -release-date (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Clothing & Accessories	Apparel	1571272031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank_authority (p. 503)	Author Availability ItemPage Keywords MerchantId
Computers & Accessories	PCHardware	976393031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the IN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Electronics	Electronics	976420031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Gift Cards	GiftCards	3704983031	-price (p. 503) -reviewrank_authority (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability Brand Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice

Product Advertising API Developer Guide
Locale Information for the IN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Gourmet & Specialty Foods	Grocery	2454179031	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Health & Personal Care	HealthPersonal-Care	1350385031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MerchantId Sort Title

Product Advertising API Developer Guide
Locale Information for the IN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Home & Kitchen	HomeGarden	2454176031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Industrial & Scientific	Industrial	5866079031	-price (p. 503) featured (p. 503) price (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the IN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Jewellery	Jewelry	1951049031	-price (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Kindle Store	KindleStore	1571278031	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the IN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Luggage & Bags	Luggage	2454170031	-price (p. 503) date-desc-rank (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Movies & TV Shows	DVD	976417031	-price (p. 503) daterank (p. 503) inverse-pricerank (p. 503) price (p. 503) releasedate (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Actor AudienceRating Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the IN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Music	Music	976446031	-price (p. 503) -released-date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Musical Instruments	MusicInstruments	3677698031	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the IN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Office Products	OfficeProducts	2454173031	-price (p. 503) -release-date (p. 503) popularity-ank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Pet Supplies	PetSupplies	4740420031	-price (p. 503) -titlerank (p. 503) price (p. 503) relevance (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the IN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Shoes & Handbags	Shoes	1571284031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Software	Software	976452031	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the IN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Sports, Fitness & Outdoors	SportingGoods	1984444031	-price (p. 503) date-desc-rank (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank_authority (p. 503)	Author Availability ItemPage Keywords MerchantId
Toys & Games	Toys	1350381031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the IN Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Video Games	VideoGames	976461031	-price (p. 503) -released-date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Watches	Watches	1350388031	-price (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Locale Information for the IT Marketplace

IT Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Tutte le categorie	All			Availability ItemPage Keywords MerchantId
Abbigliamento	Apparel	2844434031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Alimentari e cura della casa	Grocery	6198093031	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
App e Giochi	MobileApps	1661661031	-price (p. 503) pmrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Auto e Moto	Automotive	1571281031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the IT Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Bellezza	Beauty	6198083031	-price (p. 503) -release-date (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Buoni Regalo	GiftCards	3557018031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Artist Availability Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice
CD e Vinili	Music	412601031	-price (p. 503) -releasedate (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Artist Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Cancelleria e prodotti per ufficio	OfficeProducts	3606311031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Casa e cucina	Kitchen	524016031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Cura della Persona	HealthPersonal-Care	1571290031	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Elettronica	Electronics	412610031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Fai da te	Tools	2454161031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Actor Artist AudienceRating Author Availability Brand Composer Conductor Director ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Neighborhood Orchestra Power Publisher ReleaseDate Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Film e TV	DVD	412607031	-price (p. 503) -released-date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Actor AudienceRating Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Giardino e giardinaggio	Garden	635017031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Neighborhood Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Giochi e giocattoli	Toys	523998031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title
Gioielli	Jewelry	2454164031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title
Illuminazione	Lighting	1571293031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the IT Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Industria e Scienza	Industrial	5866069031	-price (p. 503) featured (p. 503) price (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Informatica	PCHardware	425917031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the IT Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Kindle Store	KindleStore	1331141031	-edition-sales-velocity (p. 503) -price (p. 503) daterank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Libri	Books	411664031	-price (p. 503) -pubdate (p. 503) -publica-tion_date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the IT Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Libri in altre lingue	ForeignBooks	433843031	-price (p. 503) -pubdate (p. 503) -publica- tion_date (p. 503) price (p. 503) relevancer- ank (p. 503) reviewrank (p. 503) reviewrank_author- ity (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title
Musica Digitale	MP3Downloads	1748204031	-albumrank (p. 503) -artistalbum- rank (p. 503) -price (p. 503) -released- ate (p. 503) -runtime (p. 503) -titlerank (p. 503) albumrank (p. 503) artistalbum- rank (p. 503) price (p. 503) relevancer- ank (p. 503) reviewrank (p. 503) reviewrank_author- ity (p. 503) runtime (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Orologi	Watches	524010031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title
Prima infanzia	Baby	1571287031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Scarpe e borse	Shoes	524007031	-launch-date (p. 503) -price (p. 503) inverse-pricerank (p. 503) price (p. 503) pricerank (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Software	Software	412613031	-price (p. 503) -released-date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Sport e tempo libero	SportingGoods	524013031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Strumenti musicali e DJ	Musicallnstruments	3628630031	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Valigeria	Luggage	2454149031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Videogiochi	VideoGames	412604031	-price (p. 503) -released-date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Locale Information for the JP Marketplace

JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	All			Availability ItemPage Keywords MerchantId

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Amazon	VideoDownload	2351650051	date-desc-rank (p. 503) popularity-rank (p. 503) price-asc-rank (p. 503) price-desc-rank (p. 503) relevancerank (p. 503) review-rank (p. 503)	Actor Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Android	MobileApps	2381131051	-price (p. 503) pmrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
DIY	HomeImprovement	2016930051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
DVD	Video	561972	-orig-rel-date (p. 503) -price (p. 503) -pricerank (p. 503) -releasedate (p. 503) -titlerank (p. 503) orig-rel-date (p. 503) price (p. 503) pricerank (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Actor Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Kindle	KindleStore	2250739051	-price (p. 503) daterank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
PC	Software	637630	-price (p. 503) -releasedate (p. 503) -releasedate (p. 503) -titlerank (p. 503) price (p. 503) release-date (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
TBD	Industrial	3445394051	-price (p. 503) featured (p. 503) price (p. 503) relevancerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
TBD	GiftCards	2351653051	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice
TBD	Kitchen	3839151	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff Sort Title

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
TBD	CreditCards	2320456051	-price (p. 503) -release-date (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
TV	VideoGames	637872	-price (p. 503) -releasedate (p. 503) -titlerank (p. 503) price (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice ReleaseDate Sort Title

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Toys	13331821	-price (p. 503) -release-date (p. 503) -releasedate (p. 503) -titlerank (p. 503) price (p. 503) release-date (p. 503) releasedate (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	Automotive	2017305051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Classical	562032	-orig-rel-date (p. 503) -price (p. 503) -pricerank (p. 503) -releasedate (p. 503) -titlerank (p. 503) orig-rel-date (p. 503) price (p. 503) pricerank (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Artist Availability Composer Conductor ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Orchestra Sort Title
	Beauty	52391051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Shoes	2016926051	-launch-date (p. 503) -price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	Jewelry	85896051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title
&	SportingGoods	14315361	-price (p. 503) -release-date (p. 503) -titlerank (p. 503) price (p. 503) release-date (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	MP3Downloads	2129039051	-albumrank (p. 503) -artistalbumrank (p. 503) -price (p. 503) -price-new-bin (p. 503) -runtime (p. 503) -titlerank (p. 503) albumrank (p. 503) artistalbumrank (p. 503) price (p. 503) price-new-bin (p. 503) releasedate (p. 503) relevancerank (p. 503) reviewrank_authority (p. 503) runtime (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	PCHardware	2127210051	-price (p. 503) -price-new-bin (p. 503) price (p. 503) price-new-bin (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
&	HealthPersonal-Care	161669011	-price (p. 503) -release-date (p. 503) -releasedate (p. 503) -titlerank (p. 503) price (p. 503) release-date (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
&	Baby	13331821	-price (p. 503) price (p. 503) psrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	PetSupplies	2127213051	-price (p. 503) -price-new-bin (p. 503) price (p. 503) price-new-bin (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Hobbies	13331821	-mfg-age-min (p. 503) -price (p. 503) -release-date (p. 503) -releasedate (p. 503) -titlerank (p. 503) mfg-age-min (p. 503) price (p. 503) releasedate (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Music	562032	-orig-rel-date (p. 503) -price (p. 503) -pricerank (p. 503) -releasedate (p. 503) -titlerank (p. 503) orig-rel-date (p. 503) price (p. 503) pricerank (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Artist Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice ReleaseDate Sort Title
	Appliances	2277725051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
&	Electronics	3210991	-price (p. 503) -release-date (p. 503) -releasedate (p. 503) -titlerank (p. 503) price (p. 503) release-date (p. 503) releasedate (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	OfficeProducts	86732051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Apparel	361299011	-price (p. 503) price (p. 503) relevancerank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	Books	465610	-price (p. 503) -publica-tion_date (p. 503) -titlerank (p. 503) -unit-sales (p. 503) daterank (p. 503) inverse-pricerank (p. 503) price (p. 503) pricerank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Musicallnstruments	2123630051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	ForeignBooks	388316011	-price (p. 503) -publica-tion_date (p. 503) -titlerank (p. 503) -unit-sales (p. 503) daterank (p. 503) inverse-pricerank (p. 503) price (p. 503) pricerank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the JP Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Watches	331952011	-price (p. 503) -titlerank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title
	Grocery	57240051	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
	DVD	562002	-orig-rel-date (p. 503) -price (p. 503) -pricerank (p. 503) -releasedate (p. 503) -titlerank (p. 503) orig-rel-date (p. 503) price (p. 503) pricerank (p. 503) releasedate (p. 503) salesrank (p. 503) titlerank (p. 503)	Actor Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher ReleaseDate Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
	Blended			Availability ItemPage Keywords

Locale Information for the MX Marketplace

MX Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Todos los departamentos	All			Availability ItemPage Keywords MaximumPrice MerchantId MinimumPrice
Bebé	Baby	9482651011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the MX Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Deportes y Aire Libre	SportingGoods	9482661011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Electrónicos	Electronics	9482559011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Herramientas y Mejoras del Hogar	HomeImprovement	9482671011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Hogar y Cocina	Kitchen	9482594011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the MX Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Libros	Books	9298577011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Música	Music	9482621011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the MX Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Películas y Series de TV	DVD	9482631011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Relojes	Watches	9482681011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the MX Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Salud, Belleza y Cuidado Personal	HealthPersonal-Care	9482611011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Software	Software	9482691011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the MX Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Tienda Kindle	KindleStore	6446440011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Videojuegos	VideoGames	9482641011	-price (p. 503) -release-date (p. 503) popularityrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Locale Information for the UK Marketplace

UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
All Departments	All			Availability ItemPage Keywords MerchantId
Amazon Instant Video	UnboxVideo	3010086031	date-desc-rank (p. 503) popularity-rank (p. 503) price-asc-rank (p. 503) price-desc-rank (p. 503) relevancerank (p. 503) review-rank (p. 503)	Actor AudienceRating Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher SearchIndex Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Amazon Pantry	Pantry	N/A	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Apps & Games	MobileApps	1661658031	-price (p. 503) pmrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Baby	Baby	60032031	-price (p. 503) price (p. 503) relevancerank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Beauty	Beauty	66280031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Books	Books	1025612	-price (p. 503) -publica-tion_date (p. 503) -titlerank (p. 503) -unit-sales (p. 503) daterank (p. 503) inverse-pricerank (p. 503) price (p. 503) pricerank (p. 503) pubdate (p. 503) publica-tion_date (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title
CDs & Vinyl	Music	520920	-price (p. 503) -released-ate (p. 503) -titlerank (p. 503) inverse-pricerank (p. 503) price (p. 503) releasedate (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Artist Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Car & Motorbike	Automotive	248878031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Classical	Classical	505510	-price (p. 503) -titlerank (p. 503) inverse-pricerank (p. 503) price (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Artist Availability Composer Conductor ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Orchestra Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Clothing	Apparel	83451031	-launch-date (p. 503) -price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Computers	PCHardware	340832031	-price (p. 503) launch_date (p.503) price (p. 503) psrank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
DIY & Tools	Tools	11052591	-price (p. 503) -titlerank (p. 503) daterank (p. 503) price (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Actor Artist AudienceRating Author Availability Brand Composer Conductor Director ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Neighborhood Orchestra Power Publisher ReleaseDate Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
DVD & Blu-ray	DVD	573406	-price (p. 503) -titlerank (p. 503) daterank (p. 503) inverse-pricerank (p. 503) price (p. 503) releasedate (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Actor AudienceRating Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Digital Music	MP3Downloads	77925031	-price (p. 503) -releasedate (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Electronics & Photo	Electronics	560800	-titlerank (p. 503) daterank (p. 503) inverse-pricerank (p. 503) price (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Garden & Outdoors	HomeGarden	11052591	-price (p. 503) -titlerank (p. 503) daterank (p. 503) price (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Gift Cards	GiftCards	1571305031	-price (p. 503) -release-date (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Grocery	Grocery	344155031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Health & Personal Care	HealthPersonal-Care	66280031	-price (p. 503) -titlerank (p. 503) daterank (p. 503) price (p. 503) releasedate (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Industrial & Scientific	Industrial	5866055031	-price (p. 503) featured (p. 503) price (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Jewellery	Jewelry	193717031	-launch-date (p. 503) -price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Kindle Store	KindleStore	341677031	-edition-sales-velocity (p. 503) -price (p. 503) daterank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Kitchen & Home	Kitchen	11052591	-price (p. 503) -titlerank (p. 503) daterank (p. 503) price (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Large Appliances	Appliances	908799031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Lighting	Lighting	213078031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Luggage	Luggage	2454167031	-price (p. 503) date-desc-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Musical Instruments & DJ	MusicallInstruments	340837031	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
PC & Video Games	VideoGames	1025616	-titlerank (p. 503) daterank (p. 503) inverse-pricerank (p. 503) price (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Pet Supplies	PetSupplies	340841031	-price (p. 503) -price-new-bin (p. 503) price (p. 503) price-new-bin (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Shoes & Bags	Shoes	362350011	-launch-date (p. 503) -price (p. 503) pmrank (p. 503) price (p. 503) reviewrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Software	Software	1025614	-titlerank (p. 503) daterank (p. 503) inverse-pricerank (p. 503) price (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Sports & Outdoors	SportingGoods	319530011	-price (p. 503) -titlerank (p. 503) price (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title
Stationery & Office Supplies	OfficeProducts	560800	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Toys & Games	Toys	712832	-mfg-age-min (p. 503) -price (p. 503) mfg-age-min (p. 503) price (p. 503) salesrank (p. 503)	Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the UK Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
VHS	VHS	125556011	-price (p. 503) -titlerank (p. 503) daterank (p. 503) inverse-pricerank (p. 503) price (p. 503) releasedate (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Actor Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Watches	Watches	328229011	-launch-date (p. 503) -price (p. 503) -titlerank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords MerchantId MinPercentageOff Sort Title
	Blended			Availability ItemPage Keywords

Locale Information for the US Marketplace

US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
All Departments	All			Availability ItemPage Keywords MaximumPrice MerchantId MinimumPrice
Amazon Instant Video	UnboxVideo	2858778011	-launch-date (p. 503) -price (p. 503) -video-release-date (p. 503) price (p. 503) relevancerank (p. 503) salesrank (p. 503) titlerank (p. 503)	Actor AudienceRating Availability Director ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Appliances	Appliances	2619526011	-price (p. 503) pmrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Apps & Games	MobileApps	2350150011	-price (p. 503) pmrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Arts, Crafts & Sewing	ArtsAndCrafts	2617942011	-price (p. 503) pmrank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Automotive	Automotive	15690151	-price (p. 503) -titlerank (p. 503) price (p. 503) relevancerank (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Baby	Baby	165797011	-price (p. 503) price (p. 503) psrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Beauty	Beauty	11055981	-launch-date (p. 503) -price (p. 503) pmrank (p. 503) price (p. 503) sale-flag (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Books	Books	1000	-price (p. 503) -publica-tion_date (p. 503) -titlerank (p. 503) -unit-sales (p. 503) daterank (p. 503) inverse-pricerank (p. 503) price (p. 503) pricerank (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Power Publisher Sort Title
CDs & Vinyl	Music	301668	-orig-rel-date (p. 503) -price (p. 503) -releasedate (p. 503) -titlerank (p. 503) artistrank (p. 503) orig-rel-date (p. 503) price (p. 503) psrank (p. 503) release-date (p. 503) releasedate (p. 503) relevancerank (p. 503) salesrank (p. 503) titlerank (p. 503)	Artist Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Cell Phones & Accessories	Wireless	2335753011	-titlerank (p. 503) daterank (p. 503) inverse-pricerank (p. 503) pricerank (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Clothing, Shoes & Jewelry	Fashion	7141124011	-price (p. 503) launch-date (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Clothing, Shoes & Jewelry - Baby	FashionBaby	7147444011	-price (p. 503) launch-date (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Clothing, Shoes & Jewelry - Boys	FashionBoys	7147443011	-price (p. 503) launch-date (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Clothing, Shoes & Jewelry - Girls	FashionGirls	7147442011	-price (p. 503) launch-date (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Clothing, Shoes & Jewelry - Men	FashionMen	7147441011	-price (p. 503) launch-date (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Clothing, Shoes & Jewelry - Women	FashionWomen	7147440011	-price (p. 503) launch-date (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Collectibles & Fine Arts	Collectibles	4991426011	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Artist Author Availability Composer Conductor ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Orchestra Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Computers	PCHardware	541966	-price (p. 503) price (p. 503) psrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Digital Music	MP3Downloads	624868011	-price (p. 503) -released-date (p. 503) price (p. 503) relevancerank (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Electronics	Electronics	493964	-price (p. 503) pmrank (p. 503) price (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Gift Cards	GiftCards	2864120011	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords MaximumPrice MerchantId MinimumPrice Neighborhood Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Grocery & Gourmet Food	Grocery	16310211	in- verseprice (p. 503) launch-date (p. 503) pricerank (p. 503) relevancerank (p. 503) sale-flag (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Health & Personal Care	HealthPersonal-Care	3760931	in- verseprice (p. 503) launch-date (p. 503) pmrank (p. 503) pricerank (p. 503) sale-flag (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Home & Kitchen	HomeGarden	1063498	-price (p. 503) -titlerank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Neighborhood Sort Title
Industrial & Scientific	Industrial	16310161	-price (p. 503) -titlerank (p. 503) pmrank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Kindle Store	KindleStore	133141011	-edition-sales-velocity (p. 503) -price (p. 503) daterank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) salesrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Luggage & Travel Gear	Luggage	9479199011	-price (p. 503) launch-date (p. 503) popularity-rank (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Availability Brand Condition ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Magazine Subscriptions	Magazines	599872	-price (p. 503) -publica-tion_date (p. 503) -titlerank (p. 503) -unit-sales (p. 503) daterank (p. 503) price (p. 503) reviewrank (p. 503) subslot-sales-rank (p. 503) titlerank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
Movies & TV	Movies	2625374011	-price (p. 503) -release-date (p. 503) featured (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Musical Instruments	Musicallnstruments	11965861	-launch-date (p. 503) -price (p. 503) pmrank (p. 503) price (p. 503) sale-flag (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Office Products	OfficeProducts	1084128	-price (p. 503) pmrank (p. 503) price (p. 503) reviewrank (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Patio, Lawn & Garden	LawnAndGarden	3238155011	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Pet Supplies	PetSupplies	2619534011	-price (p. 503) -titlerank (p. 503) price (p. 503) relevance (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewrank_authority (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Prime Pantry	Pantry	N/A	-price (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503)	Availability ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Software	Software	409488	-price (p. 503) pmrank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Sports & Outdoors	SportingGoods	3375301	-price (p. 503) in- verseprice (p. 503) launch-date (p. 503) price (p. 503) pricerank (p. 503) relevance-fs- rank (p. 503) relevancer- ank (p. 503) reviewrank_author- ity (p. 503) sale-flag (p. 503) salesrank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Tools & Home Improvement	Tools	468240	-price (p. 503) -titlerank (p. 503) pmrank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Product Advertising API Developer Guide
Locale Information for the US Marketplace

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Toys & Games	Toys	165795011	-age-min (p. 503) -price (p. 503) pmrank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title
Video Games	VideoGames	11846801	-price (p. 503) pmrank (p. 503) price (p. 503) salesrank (p. 503) titlerank (p. 503)	Author Availability Brand ItemPage Keywords Manufacturer MaximumPrice MerchantId MinPercentageOff MinimumPrice Sort Title

Department	Search index	Root Browse node	Sort values	ItemSearch parameters
Wine	Wine	2983386011	-price (p. 503) featured (p. 503) price (p. 503) relevancerank (p. 503) reviewrank (p. 503) reviewscore (p. 503)	Author Availability ItemPage Keywords MaximumPrice MerchantId MinPercentageOff MinimumPrice Publisher Sort Title
	Blended			Availability ItemPage Keywords

ItemSearch Sort Values

The `ItemSearch Sort` parameter enables you to choose the order of the items in a response. Available sort values vary by locale and search index.

There are many sort values. The majority are not applied unless the `Sort` parameter is included in the request. There are two sort values, however, that are used by default:

- For ItemSearch requests that do not use the `BrowseNode` parameter, results are sorted by relevance ([relevancerank \(p. 503\)](#)).
- For ItemSearch requests that do use the `BrowseNode` parameter, results are sorted by bestseller ranking ([psrank \(p. 503\)](#)).

Sort Values

Value	Description
-age-min	Age: high to low
albumrank	Album: A to Z
-albumrank	Album: Z to A
amzrank	Alphabetical: A to Z

Product Advertising API Developer Guide
Sort Values

Value	Description
artistalbumrank	Artist: A to Z
-artistalbumrank	Artist: Z to A
artistrank	Artist name: A to Z
availability	Most to least available
-date	Publication date: old to new
daterank	Publication date: new to old
-daterank	Publication date: old to new
date-desc-rank	Publication date: new to old
-edition-sales-velocity	Quickest to slowest selling products.
featured	Featured items
inverseprice	Price: high to low
inverse-price	Price: high to low
inverse-pricerank	Price: high to low
launchdate	Launch date: newer to older
launch-date	Launch date: newer to older
-launch-date	Launch date: older to newer
mfg-age-min	Age: low to high
-mfg-age-min	Age: high to low
orig-rel-date	Original release date: earliest to latest
-orig-rel-date	Original release date: latest to earliest
paidsalesrank	Bestseller ranking taking into consideration projected sales. The lower the value, the better the sales.
pct-off	Discount: high to low
-pct-off	Discount: low to high
pmrank	Featured items
popularityrank	Items ranked by popularity
popularity-rank	Items ranked by popularity
price	Price: low to high
-price	Price: high to low
price-asc-rank	Price: low to high
price-desc-rank	Price: high to low
price-new-bin	Price: low to high

Product Advertising API Developer Guide
Sort Values

Value	Description
-price-new-bin	Price: high to low
pricerank	Price: low to high
-pricerank	Price: high to low
psrank	Bestseller ranking taking into consideration projected sales. The lower the value, the better the sales.
pubdate	Publication date: newest to oldest
-pubdate	Publication date: oldest to most recent
publicationdate	Publication date: newest to oldest
publication_date	Publication date: newest to oldest
-publicationdate	Publication date: oldest to most recent
-publication_date	Publication date: oldest to most recent
releasedate	Release date: older to newer
release-date	Release date: older to newer
-releasedate	Release date: newer to older
-release-date	Release date: newer to older
relevance	Items ranked according to the following criteria: how often the keyword appears in the description, where the keyword appears (the ranking is higher when keywords are found in titles and—if there are multiple keywords—how closely they occur in descriptions), and how often customers purchased the products they found using the keyword.
relevance-fs-rank	
relevancerank	Items ranked according to the following criteria: how often the keyword appears in the description, where the keyword appears (the ranking is higher when keywords are found in titles and—if there are multiple keywords—how closely they occur in descriptions), and how often customers purchased the products they found using the keyword.
reviewrank	Highest to lowest ratings in customer reviews.
review-rank	Highest to lowest ratings in customer reviews.
reviewrank_authority	Review rank: high to low
-reviewrank_authority	Review rank: low to high
reviewscore	Review score
runtime	Track length: high to low
-runtime	Track length: low to high
sale-flag	On sale
salesrank	Bestselling
songtitlerank	Most popular

Product Advertising API Developer Guide
Sort Values

Value	Description
subslot-salesrank	Bestselling
titlerank	Alphabetical: A to Z
-titlerank	Alphabetical: Z to A
-unit-sales	
uploaddaterank	Date added
-video-release-date	Release date: newer to older
xsrelevancerank	

Resources

Use the following resources when working with the Product Advertising API.

Resource	Description
Product Advertising API Scratchpad	Use this tool to send requests and view sample code responses.
Getting Started Guide	Read this documentation for a quick introduction to the Product Advertising API.
Product Advertising API Best Practices	Follow this checklist of best practices.
Discussion Forums	Join the community of developers who are using our service.
Conditions of Use	Read detailed information about the copyright and trademark usage at Amazon.com and other topics.
Contact Us	Contact us for inquiries concerning billing, accounts, events, abuse, and more.

See the Product Advertising API website for your locale:

Locale	URL
Canada	https://associates.amazon.ca/gp/advertising/api/detail/main.html
China	https://associates.amazon.cn/gp/advertising/api/detail/main.html
France	http://partenaires.amazon.fr/gp/advertising/api/main.html
Germany	http://partner.net.amazon.de/gp/advertising/api/main.html
India	https://affiliate-program.amazon.in/gp/advertising/api/detail/main.html
Italy	https://programma-affiliazione.amazon.it/gp/advertising/api/detail/main.html
Japan	https://affiliate.amazon.co.jp/gp/advertising/api/detail/main.html
Mexico	https://afiliados.amazon.com.mx/gp/advertising/api/detail/main.html

Locale	URL
Spain	https://afiliados.amazon.es/gp/advertising/api/detail/main.html
United Kingdom	https://affiliate-program.amazon.co.uk/gp/advertising/api/detail/main.html
United States	https://affiliate-program.amazon.com/gp/advertising/api/detail/main.html

Document History

API version: 2013-08-01

Latest documentation update: 01 December 2015

Change	Description	Release Date-New
New value for RelationshipType parameter	The NewerVersion value returns the latest version of an item. For more information, see Discover Newer Versions (p. 98) .	25 November 2015
Updated Scratchpad tool	Scratchpad now supports all operations and returns sample code responses. Use this tool to generate sample code and help debug your requests. For more information, see Using the Product Advertising API Scratchpad (p. 8) .	8 September 2015
Change to batch requests	Batch requests are no longer supported for the cart operations: CartAdd, CartClear, CartCreate, CartGet, and CartModify. Batch requests are still supported for all other operations.	4 September 2015
New Marketplace	The MX (Mexico) marketplace was added.	25 August 2015
Updated locale information	We added the Department name associated with the search index and root browse node. For more information, see Locale Reference (p. 328) .	
New OfferListing flag	The IsEligibleForPrime flag indicates if an item is eligible for Amazon Prime.	31 July 2015
New OfferListing flag	The IsEligibleForPrimeFreeDigitalVideo flag indicates if an Amazon Instant Video is eligible to view for free with a Prime Subscription.	24 June 2014
New Marketplace	The IN (India) marketplace was added.	6 June 2013
Added MarketplaceDomain support for www.amazonsupply.com	We added MarketplaceDomain support for a new URL, www.amazonsupply.com.	25 July 2012

Change	Description	Release Date-New
New SearchIndex values	We added support for two SearchIndex values: HomeImprovement (FR) and Collectibles (US)	16 July 2012
Support for music in AlternateVersions	We've added support for additional product categories. SearchIndex values for AlternateVersions now include Music, KindleStore, and MP3Downloads.	20 June 2012
New SearchIndex values	We added two new SearchIndex values: LawnAndGarden (US) and Appliances (JP)	4 April 2012
New Marketplace	The ES (Spain) marketplace was added.	20 September 2011
New Marketplace	The IT (Italy) and CN (China) marketplaces were added.	1 August 2011
Changes to Operations and Response Groups	As part of our efforts to streamline the Product Advertising API, we will be making changes to the currently supported operations and response groups. This guide will be updated to reflect these changes in the next few weeks. For more information about these changes, see Changes to the Product Advertising API .	26 July 2011

Index

A

About response element, 310
AboutMe response element, 310
accessories, 39
Active, 155
Actor response element, 310
Add to Cart form, 166
AdditionalName response element, 310
after, 82
All, 34, 34, 79
AlternateVersion response element, 311
AlternateVersions, 232
Amount response element, 311
Artist response element, 311
ASIN, 20, 82
ASIN response element, 311
AspectRatio response element, 311
Associates, 182, 230
AssociateTag, 64, 64
attribute item, 70
AudienceRating response element, 311
AudioFormat response element, 311
authentication, 48, 161
 REST, 51
 signature version 2, 51
 SOAP, 59
author, 82
Author response element, 311
author-begins, 82
author-exact, 82
availability, 75, 77, 154
 Availability, 151
 determining, 151
 element, 151
 limited, 152
 returning, 145, 145
 values, 152
Availability, 77
Availability response element, 311
AvailabilityAttributes response element, 311
AWSAccessKeyId, 64

B

benefit items, 129
Benefit response element, 311
BenefitDescription, 130
BenefitDescription response element, 312
Benefits response element, 312
BenefitType response element, 312
best programming practices, 178
bin, 288
Bin response element, 312
binding, 16, 82

Binding response element, 312
BinItemCount response element, 312
BinName response element, 312
BinParameter response element, 312
blended, 34, 80
Brand response element, 312
browse node, 27
 drilling down, 94
 finding, 31
 finding items using, 91
 ID, 30, 328
 properties, 28
 root category, 28
 search indices, 30
 traversing hierarchy, 92
BrowseNode, 77
BrowseNodeId, 195
BrowseNodeId response element, 312
BrowseNodeInfo, 234
BrowseNodeLookup, 73, 194
BrowseNodes, 236
BuyAmountXGetAmountOffX, 128, 277
BuyAmountXGetSimpleShippingFreeX, 128, 277
BuyQuantityXGetAmountOffX, 128
BuyQuantityXGetPercentOffX, 128

C

caching, 180
cart , 155, 157
 Active, 155
 Active area, 156
 Add to Cart form, 166
 adding items to, 162
 create, 158
 expiration, 158
 identifier, 155
 lifespan, 157
 local cart, 158
 modify, 157
 new releases, 143
 purchasing items in, 167
 remote cart, 158
 removing items, 164
 returning items in, 165
 similar items, 135
 tasks, 158
 working with, 154
Cart, 238
CartAdd, 162, 207
CartClear, 162, 164, 212
CartCreate, 159, 214
CartGet, 220
CartId, 160
CartId response element, 312
CartItem response element, 312
CartItemId, 160, 162
CartItem response element, 312

CartItems response element, 312

CartModify, 162, 224

CartNewReleases, 143, 143, 241

CartSimilarities, 244

CartTopSellers, 242

Category response element, 313

CEROAgeRating response element, 313

ClothingSize response element, 313

Code response element, 313

collection

parent, 157

Collection response element, 313

CollectionItem response element, 313

CollectionParent response element, 313

Color response element, 313

Comment response element, 313

ComponentType, 130

ComponentType response element, 313

condition, 75, 144

Condition, 77

Condition response element, 313

CorrectedQuery response element, 313

CouponCombinationType, 130

CouponCombinationType response element, 313

Creator response element, 314

CurrencyAmount response element, 314

CurrencyCode response element, 314

currently for sale, 74, 151

customer

motivating, 128

customer review , 132

D

DateAdded response element, 314

DateCreated response element, 314

debugging parameters, 66

Department response element, 314

Details response element, 314

Digital items, 156

dimension, 37

Director response element, 314

E

e-commerce, 9

e-mail me when the items become available, 74, 151

EAN, 21

EAN response element, 314

EANList response element, 314

EANListElement response element, 315

Edition response element, 315

editorial review , 131

EditorialReview, 245

EditorialReviewIsLinkSuppressed response element, 315

EISBN, 21, 82

EISBN response element, 315

EligibilityDescription, 130

EligibilityRequirement response element, 315

EligibilityRequirementDescription response element, 315

EligibilityRequirements response element, 315

EligibilityRequirementType response element, 315

eligible items, 129

EndDate response element, 315

endpoint, 42

endpoints, SOAP , 47

EpisodeSequence response element, 315

error, 169

 codes and messages, 172

 retrieving, 171

ESRBAgeRating response element, 315

Exclusive, 130

F

Feature response element, 315

Feedback response element, 315

finding

 items, 25

Fitment response element, 315

FitmentAttribute response element, 316

FitmentAttributes response element, 316

FixedAmount response element, 316

ForEachQuantityXGetQuantityFreeX, 128, 277

Format response element, 316

FormattedPrice, 130

FormattedPrice response element, 316

G

Genre response element, 316

getting started, 4

GroupClaimCode response element, 316

H

HardwarePlatform response element, 316

HasReviews response element, 316

HazardousMaterialType response element, 316

headers, SOAP, 61

Height response element, 316

HMAC, 160

HMAC response element, 316

HMAC signatures, 49, 51, 60

HTTPS

 with REST requests, 51

 with SOAP requests, 60

I

ID, offer listing , 153

identifier, 20, 160

IFrameURL response element, 317

image

 variation, 38

Image response element, 317

Images response group, 247

Intersection, 134

inversepricerank, 144

IsAdultProduct response element, 317
IsAutographed response element, 317
ISBN, 21, 82
ISBN response element, 317
ISBN-13, 21
IsCategoryRoot response element, 317
IsEligibleForPrime response element, 317
IsEligibleForPrimeFreeDigitalVideo response element, 317
IsEligibleForSuperSaverShipping response element, 317
IsEligibleForTradeIn response element, 317
IsEmailNotifyAvailable response element, 317
IsFit response element, 317
IsInBenefitSet, 129, 130
IsInBenefitSet response element, 318
IsInEligibilityRequirementSet, 130
IsInEligibilityRequirementSet response element, 318
IsInEligibilitySet, 129
IsLinkSuppressed response element, 318
IsMemorabilia response element, 318
IsNext response element, 318
IsPrevious response element, 318
IssuesPerYear response element, 318
IsValid, 68
IsValid response element, 318
ItemApplicability response element, 318
ItemAttributes, 249
ItemAttributes response element, 318
ItemDimensions response element, 318
ItemIds, 253
ItemLookup, 197
ItemPartNumber response element, 318
items, 149
 attributes, 70
 available, 77
 available for in-store pickup, 74
 benefit, 129
 eligible, 129
 find using browse nodes, 91
 finding, 25, 75
 finding out more about one item, 25
 for sale, 20
 for sale by third parties, 74
 identifier, 20
 organization of, 26
 parent, 149
 purchasing, 167
 removing from cart, 164
 similar IDs, 135
 similar items to those in cart, 135
 similar version, 136
 sold by merchants, 75
 specifying multiple, 134
 suggesting similar items, 133
 variation offer, 149
 you can't buy, 73
ItemSearch, 73, 75, 185
 Availability parameter, 154

common parameters, 77
request tips, 91
ItemSearch parameters
 values by locale, 328

J
JAN, 21

K
Keywords, 77, 82
Keywords response element, 318

L
Label response element, 318
Language response element, 318
Languages response element, 319
Large, 73, 254
LargeImage response element, 319
LastModified response element, 319
LegalDisclaimer response element, 319
Length response element, 319
licensing agreement, 7
limited availability, 152
ListItemId response element, 319
ListPrice response element, 319
locale, 22, 181
 search indexes, 328
loyalty points, 273
LoyaltyPoints response element, 319
LoyaltyPoints response element (JP locale), 319

M
making money, 20
Manufacturer response element, 319
ManufacturerMaximumAge response element, 319
ManufacturerMinimumAge response element, 319
ManufacturerPartsWarrantyDescription response element, 319
marketplace
 defined, 19
 values by locale, 328
MaterialType response element, 319
maximum
 allowance, 155
MaximumHours response element, 319
MaximumPrice, 77
MediaType response element, 320
Medium, 259
MediumImage response element, 320
MerchandisingMessage response element, 320
MerchantId, 65
MerchantId response element, 320
Message response element, 320
MetalType response element, 320
MinimumHours response element, 320
MinimumPrice, 77

Model response element, 320
money, making, 20
MoreOffersUrl response element, 320
MostGifted response group, 263
MostWishedFor response group, 264
motivating customers, 128
MPN response element, 320
Music, 34

N

Name response element, 320
namespaces, 13
NarrowBy, 86, 286
 search index, 287
 values, 87
new releases, 74, 143
new releases, from shopping cart , 143
NewReleases, 143, 265
Nickname response element, 320
not available, 152
not stocked, 152
Not yet published, 152
Not yet released, 152
Number response element, 321
NumberOfDiscs response element, 321
NumberOfIssues response element, 321
NumberOfItems response element, 321
NumberOfPages response element, 321
NumberOfTracks response element, 321

O

OccasionDate response element, 321
offer, 149, 160
 listing ID, 153
 variation, 149
OfferFull, 73, 267
OfferListingId, 20, 160
OfferListingId response element, 321
OfferListings, 269
Offers response group, 147
OfferSummary , 146, 153, 275
OperatingSystem response element, 321
operation, 64
 batch, 61
 finding items, 23
 finding out more about one item, 23
 search, 73
 shopping cart, 24
 summary, 23
optional parameter, 64
Order Pipeline, 167
OtherCategoriesSimilarProducts response element, 321
Out of Print, 152
Out of Stock, 152, 156

P

PackageQuantity response element, 321
paging, 70
 maximum returned pages, 71
parameter
 compound, 45
 debugging, 66
 required, 64
 value, 45
 XML encoding, 65
parent
 collection, 157
 item, 149
 variation, 156
ParentASIN response element, 321
PartBrandBins response element, 321
PartBrowseNodeBins response element, 321
PartnerName response element, 321
PartNumber response element, 321
Platform response element, 322
popularity, 144
PortType, 16
power search, 81
 Keywords, 82
 operators, 83
 syntax, 82
 tips, 84
pre-orders , 74
pre-release order, 151
Preferential, 130
price, 144, 144
 ranges, 148
Price response element, 322
pricerank, 144
Proceed to Checkout, 168
Product Advertising API, 19, 24
ProductGroup response element, 322
ProductTypeSubcategory response element, 322
promotion, 128
 category, 128
Promotion response element, 322
PromotionId response element, 322
Promotions response element, 322
PromotionSummary, 277
pubdate, 82
PublicationDate response element, 322
publisher, 82
Publisher response element, 322
PurchaseURL, 160, 162, 168
PurchaseURL response element, 322

Q

Quantity response element, 322

R

Random, 134

RegionCode response element, 322
RegistryName response element, 322
RelatedItem response element, 322
RelatedItemPage response element, 323
RelatedItemCount response element, 323
RelatedItems response element, 323
RelatedItemsCount response element, 323
Relationship response element, 323
RelationshipType response element, 323
ReleaseDate response element, 323
relevancerank, 144
request
 examples, 95
 format, 41
 limitations, 44
 parameters, common, 228
 response, 66
Request, 281
RequestId response element, 323
required parameters, 64
response , 66
 anatomy of, 67
 elements, 309
 elements common, 309
 examples, 95
 groups, 43
response element
 About, 310
 AboutMe, 310
 Actor, 310
 AdditionalName, 310
 AlternateVersion, 311
 Amount, 311
 Artist, 311
 ASIN, 311
 AspectRatio, 311
 AudienceRating, 311
 AudioFormat, 311
 Author, 311
 Availability, 311
 AvailabilityAttributes, 311
 Benefit, 311
 BenefitDescription, 312
 Benefits, 312
 BenefitType, 312
 Bin, 312
 Binding, 312
 BinItemCount, 312
 BinName, 312
 BinParameter, 312
 Brand, 312
 BrowseNodeId, 312
 CartId, 312
 CartItem, 312
 CartItem, 312
 CartItems, 312
 Category, 313
 CEROAgeRating, 313
 ClothingSize, 313
 Code, 313
 Collection, 313
 CollectionItem, 313
 CollectionParent, 313
 Color, 313
 Comment, 313
 ComponentType, 313
 Condition, 313
 CorrectedQuery, 313
 CouponCombinationType, 313
 Creator, 314
 CurrencyAmount, 314
 CurrencyCode, 314
 DateAdded, 314
 DateCreated, 314
 Department, 314
 Details, 314
 Director, 314
 EAN, 314
 EANList, 314
 EANListElement, 315
 Edition, 315
 EditorialReviewIsLinkSuppressed, 315
 EISBN, 315
 EligibilityRequirement, 315
 EligibilityRequirementDescription, 315
 EligibilityRequirements, 315
 EligibilityRequirementType, 315
 EndDate, 315
 EpisodeSequence, 315
 ESRBAgeRating, 315
 Feature, 315
 Feedback, 315
 Fitment, 315
 FitmentAttribute, 316
 FitmentAttributes, 316
 FixedAmount, 316
 Format, 316
 FormattedPrice, 316
 Genre, 316
 GroupClaimCode, 316
 HardwarePlatform, 316
 HasReviews, 316
 HazardousMaterialType, 316
 Height, 316
 HMAC, 316
 IFrameURL, 317
 Image, 317
 IsAdultProduct, 317
 IsAutographed, 317
 ISBN, 317
 IsCategoryRoot, 317
 IsEligibleForPrime, 317
 IsEligibleForPrimeFreeDigitalVideo, 317
 IsEligibleForSuperSaverShipping, 317
 IsEligibleForTradeIn, 317

IsEmailNotifyAvailable, 317
IsFit, 317
IsInBenefitSet, 318
IsInEligibilityRequirementSet, 318
IsLinkSuppressed, 318
IsMemorabilia, 318
IsNext, 318
IsPrevious, 318
IssuesPerYear, 318
IsValid, 318
ItemApplicability, 318
ItemAttributes, 318
ItemDimensions, 318
ItemPartNumber, 318
Keywords, 318
Label, 318
Language, 318
Languages, 319
LargeImage, 319
LastModified, 319
LegalDisclaimer, 319
Length, 319
ListItemId, 319
ListPrice, 319
LoyaltyPoints, 319
LoyaltyPoints (JP locale), 319
Manufacturer, 319
ManufacturerMaximumAge, 319
ManufacturerMinimumAge, 319
ManufacturerPartsWarrantyDescription, 319
MaterialType, 319
MaximumHours, 319
MediaType, 320
MediumImage, 320
MerchandisingMessage, 320
MerchantId, 320
Message, 320
MetalType, 320
MinimumHours, 320
Model, 320
MoreOffersUrl, 320
MPN, 320
Name, 320
Nickname, 320
Number, 321
NumberOfDiscs, 321
NumberOfIssues, 321
NumberOfItems, 321
NumberOfPages, 321
NumberOfTracks, 321
OccasionDate, 321
OfferListingId, 321
OperatingSystem, 321
OtherCategoriesSimilarProducts, 321
PackageQuantity, 321
ParentASIN, 321
PartBrandBins, 321
PartBrowseNodeBins, 321
PartnerName, 321
PartNumber, 321
Platform, 322
Price, 322
ProductGroup, 322
ProductTypeSubcategory, 322
Promotion, 322
PromotionId, 322
Promotions, 322
PublicationDate, 322
Publisher, 322
PurchaseURL, 322
Quantity, 322
RegionCode, 322
RegistryName, 322
RelatedItem, 322
RelatedItemPage, 323
RelatedItemCount, 323
Relationship, 323
RelationshipType, 323
ReleaseDate, 323
RequestId, 323
Role, 323
RunningTime, 323
SalesRank, 323
SavedForLaterItem, 323
SearchBinSet, 323
SearchBinSets, 323
SeikodoProductCode, 324
ShipmentItems, 324
Shipments, 324
SimilarProducts, 324
SimilarViewedProducts, 324
Size, 324
SKU, 324
SmallImage, 324
Source, 324
StartDate, 324
StoreId, 324
StoreName, 324
Studio, 324
SubscriptionLength, 324
Summary, 325
SwatchImage, 325
TermsAndConditions, 325
ThumbnailImage, 325
TinyImage, 325
Title, 325
TopItem, 325
TopItemSet, 325
TotalCollectible, 325
TotalItems, 325
TotalNew, 325
TotalOfferPages, 325
TotalOffers, 325
TotalPages, 326

TotalRefurbished, 326
TotalResults, 326
Totals, 326
TotalTimesRead, 326
TotalUsed, 326
TotalVotes, 326
Track, 326
TradeInValue, 326
TransactionDate, 326
TransactionDateEpoch, 326
TransactionId, 326
TransactionItem, 326
TransactionItemId, 326
TransactionItems, 326
Type, 327
UPC, 327
UPCList, 327
UPCListElement, 327
URL, 327
URLEncodedHMAC, 327
UserAgent, 327
UserId, 327
VariationAttribute, 327
VariationDimension, 327
Warranty, 327
WEETaxValue, 327
Weight, 327
Width, 328
Year, 328
ResponseGroup, 65
REST, 9, 41, 44
 authentication, 51
review
 customer, 132
 editorial, 131
reviewrank, 144
Reviews response group, 283
RFC 3986, 55
Role response element, 323
root category, 28
RunningTime response element, 323

S
salesrank, 144
SalesRank, 285
SalesRank response element, 323
SavedForLaterItem, 155, 163, 163, 219
SavedForLaterItem response element, 323
schema, 18, 19
scratchpad, 51
search
 operations, 73
 power, 81
search bin, 85
 (see also SearchBins)
search index , 33
 browse node, 30

by locale, 328
combined, 34
ItemSearch parameters, 76
search, All, 79
SearchBins, 148, 286
 (see also search bin)
SearchBinSet response element, 323
SearchBinSets response element, 323
searching by keyword, 120
SeikodoProductCode response element, 324
seller
 top, 140
separator characters, 44
Service, 17, 42, 64
ShipmentItems response element, 324
Shipments response element, 324
shipping restrictions, 182
shopping cart, 26
 (see also cart)
signature
 example, 55
signature version 2, 51
signatures, 48
 REST, 51
similar
 item IDs, 135
 items, 133
Similarities, 291
SimilarLookup, 133, 203
SimilarProducts response element, 324
SimilarViewedProducts response element, 324
Size response element, 324
SKU, 21
SKU response element, 324
Small response group, 293
SmallImage response element, 324
SOAP, 9, 46, 47
 authentication, 59
 headers, 61
sort, 71
 default values, 72
 popularity, price, condition, 144
 values by locale, 328
Source response element, 324
special order, 74, 151, 152
SSL
 with REST requests, 51
 with SOAP requests, 60
StartDate response element, 324
StoreId response element, 324
StoreName response element, 324
Studio response element, 324
subject, 82
subject-begins, 82
subject-words-begin, 82
SubscriptionLength response element, 324
Summary response element, 325
SwatchImage response element, 325

T

TermsAndConditions, 130
TermsAndConditions response element, 325
ThumbnailImage response element, 325
TinyImage response element, 325
Title, 77, 82
Title response element, 325
title-begins, 82
title-words-begin, 82
Too Low to Display, response string, 146
top seller, 140, 141
 from shopping cart, 142
TopItem response element, 325
TopItemSet response element, 325
TopSellers response group, 294
TotalCollectible response element, 325
TotalItems response element, 325
TotalNew response element, 325
TotalOfferPages response element, 325
TotalOffers response element, 325
TotalPages response element, 326
TotalRefurbished response element, 326
TotalResults response element, 326
Totals response element, 326
TotalTimesRead response element, 326
TotalUsed response element, 326
TotalVotes response element, 326
Track response element, 326
Tracks response group, 296
TradeInValue response element, 326
TransactionDate response element, 326
TransactionDateEpoch response element, 326
TransactionId response element, 326
TransactionItem response element, 326
TransactionItemId response element, 326
TransactionItems response element, 326
traversing browse nodes, 92
troubleshooting, 171
Type response element, 327

U

Unrestricted, 130
UPC, 21
UPC response element, 327
UPCList response element, 327
UPCListElement response element, 327
URL response element, 327
URLEncodedHMAC, 160, 161
URLEncodedHMAC response element, 327
UserAgent response element, 327
UserId response element, 327
Usually ships in %X, 152

V

Validate, 66
validation, 68

variation, 35, 150
 dimension, 37, 38, 297
 images, 38
 parent, 35, 156
VariationAttribute response element, 327
VariationAttributes, 37
VariationDimension, 151
VariationDimension response element, 327
VariationDimensions, 37
VariationImages, 300
VariationMatrix response group, 302
VariationOffers response group, 304
Variations response group, 297
VariationSummary response group, 307
Version, 13, 65, 67
Video, 34
visual introduction, 24

W

Warranty response element, 327
web service, 9
WEEE Tax Value response element, 327
Weight response element, 327
Width response element, 328
WSDL, 9, 11
 anatomy of, 12
 location of, 18

X

XML, 9, 9
 encoding parameters, 65
 structures, 10
 syntax, 10
 using, 11
XMLEscaping, 65

Y

Year response element, 328