

## JSDoc: Class: domains

### domains(domainOrOptionsopt)

**new domains(domainOrOptionsopt)**

**Parameters:**

Name	Type	Attributes	Description												
domainOrOptions	string   object	<optional>	<p>The project domain or options object. If object, see the object's optional properties.</p> <p><b>Properties</b></p> <table><tr><th>Name</th><th>Type</th><th>Attributes</th><th>Description</th></tr><tr><td>domain</td><td>string</td><td>&lt;optional&gt;</td><td>The project domain</td></tr><tr><td>token</td><td>object</td><td>&lt;optional&gt;</td><td>auth token - object with value property and optional headerOrQueryName and isQuery properties</td></tr></table>	Name	Type	Attributes	Description	domain	string	<optional>	The project domain	token	object	<optional>	auth token - object with value property and optional headerOrQueryName and isQuery properties
Name	Type	Attributes	Description												
domain	string	<optional>	The project domain												
token	object	<optional>	auth token - object with value property and optional headerOrQueryName and isQuery properties												

Source:

- [domains.js](#), [line 2](#)

### Methods

**available(domain, checkType, forTransfer)**

Determine whether or not the specified domain is available for purchase

**Parameters:**

Name	Type	Description
domain	string	Domain name whose availability is to be checked
checkType	string	Optimize for time ('FAST') or accuracy ('FULL')
forTransfer	boolean	Whether or not to include domains available for transfer. If set to True, checkType is ignored

Source:

- [domains.js](#), [line 323](#)

### **availableBulk(domains, checkType)**

Determine whether or not the specified domains are available for purchase

**Parameters:**

Name	Type	Description
domains	array	Domain names for which to check availability
checkType	string	Optimize for time ('FAST') or accuracy ('FULL')

Source:

- [domains.js](#), [line 423](#)

### **cancel(domain)**

Cancel a purchased domain

**Parameters:**

Name	Type	Description
domain	string	Domain to cancel

Source:

- [domains.js](#), [line 142](#)

### **cancelPrivacy(xShopperId, domain)**

Submit a privacy cancellation request for the given domain

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper ID of the owner of the domain
domain	string	Domain whose privacy is to be cancelled

Source:

- [domains.js](#), [line 633](#)

### **ContactsValidate(xPrivateLabelId, marketId, body)**

Validate the request body using the Domain Contact Validation Schema for specified domains.

**Parameters:**

Name	Type	Description
xPrivateLabelId	integer	PrivateLabelId to operate as, if different from JWT
marketId	string	MarketId in which the request is being made, and for which responses should be localized
body	DomainsContactsBulk	An instance document expected for domains contacts validation

Source:

- [domains.js](#), [line 2070](#)

### **createIdentityDocument(xShopperId, identityDocument)**

Create an Identity Document from uploaded image

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper ID of the owner of the identity document. This is only required if you are a Reseller managing,identity documents for your customers. Use this header to pass in their subaccount ID
identityDocument	IdentityDocumentCreate	Identity document data

Source:

- [domains.js](#), [line 1476](#)

**createVerification(xShopperId, identityDocumentId, tlds)**

Initiate verifications for the specified Domains

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper ID of the owner of the identity document. This is only required if you are a Reseller managing,identity documents for your customers. Use this header to pass in their subaccount ID
identityDocumentId	string	Unique id of an identity document
tlds	array	An array of TLDs for which the verification was started. Alternately you can specify the whole domain from which the TLD will be extracted

Source:

- [domains.js](#), [line 1656](#)

**get(xShopperId, domain)**

Retrieve details for the specified Domain

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper ID expected to own the specified domain
domain	string	Domain name whose details are to be retrieved

Source:

- [domains.js](#), [line 230](#)

**getAgreement(xMarketId, tlds, privacy, forTransfer)**

Retrieve the legal agreement(s) required to purchase the specified TLD and add-ons

**Parameters:**

Name	Type	Description
xMarketId	string	Unique identifier of the Market used to retrieve/translate Legal Agreements
tlds	array	list of TLDs whose legal agreements are to be retrieved
privacy	boolean	Whether or not privacy has been requested
forTransfer	boolean	Whether or not domain tranfer has been requested

Source:

- [domains.js](#), [line 1960](#)

**getIdentityDocumentVerification(xShopperId, identityDocumentId, tlds)**

Retrieve a list of Verifications for the specified Identity Document

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper ID of the owner of the identity document. This is only required if you are a Reseller managing identity documents for your customers. Use this header to pass in their subaccount ID
identityDocumentId	string	Unique identifier of the current identity document
tlds	array	An array of TLDs for which to retrieve identity document verification jobs. Alternately you can specify the whole domain from which the TLD will be extracted

Source:

- [domains.js, line 1759](#)

**list(xShopperId, statuses, statusGroups, limit, marker, includes, modifiedDate)**

Retrieve a list of Domains for the specified Shopper

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper ID whose domains are to be retrieved
statuses	array	Only include results with `status` value in the specified set
statusGroups	array	Only include results with `status` value in any of the specified groups
limit	integer	Maximum number of domains to return
marker	string	Marker Domain to use as the offset in results
includes	array	Optional details to be included in the response
modifiedDate	string	Only include results that have been modified since the specified date

Source:

- [domains.js, line 518](#)

**listIdentityDocuments(xShopperId)**

Get a collection of identity documents the current shopper owns

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper ID of the owner of the identity document. This is only required if you are a Reseller managing identity documents for your customers. Use this header to pass in their subaccount ID

Source:

- [domains.js, line 1571](#)

**purchase(xShopperId, body)**

Purchase and register the specified Domain

**Parameters:**

Name	Type	Description
xShopperId	string	The Shopper for whom the domain should be purchased
body	DomainPurchase	An instance document expected to match the JSON schema returned by `./schema/{tld}`

Source:

- [domains.js, line 899](#)

**purchasePrivacy(xShopperId, domain, body)**

Purchase privacy for a specified domain

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper ID of the owner of the domain
domain	string	Domain for which to purchase privacy
body	PrivacyPurchase	Options for purchasing privacy

Source:

- [domains.js](#), [line 1373](#)

### **recordAdd(xShopperId, domain, records)**

Add the specified DNS Records to the specified Domain

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper ID which owns the domain. NOTE: This is only required if you are a Reseller managing a domain purchased outside the scope of your reseller account. For instance, if you're a Reseller, but purchased a Domain via <a href="http://www.godaddy.com">http://www.godaddy.com</a>
domain	string	Domain whose DNS Records are to be augmented
records	array	DNS Records to add to whatever currently exists

Source:

- [domains.js](#), [line 2382](#)

### **recordGet(xShopperId, domain, type, name, offset, limit)**

Retrieve DNS Records for the specified Domain, optionally with the specified Type and/or Name

**Parameters:**



Name	Type	Description
xShopperId	string	Shopper ID which owns the domain. NOTE: This is only required if you are a Reseller managing a domain purchased outside the scope of your reseller account. For instance, if you're a Reseller, but purchased a Domain via <a href="http://www.godaddy.com">http://www.godaddy.com</a>
domain	string	Domain whose DNS Records are to be retrieved
type	string	DNS Record Type for which DNS Records are to be retrieved
name	string	DNS Record Name for which DNS Records are to be retrieved
offset	integer	Number of results to skip for pagination
limit	integer	Maximum number of items to return

Source:

- [domains.js](#), [line 2170](#)

### **recordReplace(xShopperId, domain, records)**

Replace all DNS Records for the specified Domain

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper ID which owns the domain. NOTE: This is only required if you are a Reseller managing a domain purchased outside the scope of your reseller account. For instance, if you're a Reseller, but purchased a Domain via <a href="http://www.godaddy.com">http://www.godaddy.com</a>
domain	string	Domain whose DNS Records are to be replaced
records	array	DNS Records to replace whatever currently exists

Source:

- [domains.js](#), [line 2279](#)

**recordReplaceType(xShopperId, domain, type, records)**

Replace all DNS Records for the specified Domain with the specified Type

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper ID which owns the domain. NOTE: This is only required if you are a Reseller managing a domain purchased outside the scope of your reseller account. For instance, if you're a Reseller, but purchased a Domain via <a href="http://www.godaddy.com">http://www.godaddy.com</a>
domain	string	Domain whose DNS Records are to be replaced
type	string	DNS Record Type for which DNS Records are to be replaced
records	array	DNS Records to replace whatever currently exists

Source:

- [domains.js](#), [line 2485](#)

**recordReplaceTypeName(xShopperId, domain, type, name, records)**

Replace all DNS Records for the specified Domain with the specified Type and Name

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper ID which owns the domain. NOTE: This is only required if you are a Reseller managing a domain purchased outside the scope of your reseller account. For instance, if you're a Reseller, but purchased a Domain via <a href="http://www.godaddy.com">http://www.godaddy.com</a>
domain	string	Domain whose DNS Records are to be replaced
type	string	DNS Record Type for which DNS Records are to be replaced

Name	Type	Description
name	string	DNS Record Name for which DNS Records are to be replaced
records	array	DNS Records to replace whatever currently exists

Source:

- [domains.js, line 2596](#)

### **renew(xShopperId, domain, body)**

Renew the specified Domain

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper for whom Domain is to be renewed. NOTE: This is only required if you are a Reseller managing a domain purchased outside the scope of your reseller account. For instance, if you're a Reseller, but purchased a Domain via <a href="http://www.godaddy.com">http://www.godaddy.com</a>
domain	string	Domain to renew
body	DomainRenew	Options for renewing existing Domain

Source:

- [domains.js, line 994](#)

### **schema(tld)**

Retrieve the schema to be submitted when registering a Domain for the specified TLD

**Parameters:**

Name	Type	Description
------	------	-------------

Name	Type	Description
tld	string	The Top-Level Domain whose schema should be retrieved

Source:

- [domains.js](#), [line 1092](#)

**setToken(value, headerOrQueryName, isQuery)**

Set Token

**Parameters:**

Name	Type	Description
value	string	token's value
headerOrQueryName	string	the header or query name to send the token at
isQuery	boolean	true if send the token as query param, otherwise, send as header param

Source:

- [domains.js](#), [line 24](#)

**suggest(xShopperId, query, country, city, sources, tlds, lengthMax, lengthMin, limit, waitMs)**

Suggest alternate Domain names based on a seed Domain, a set of keywords, or the shopper's purchase history

**Parameters:**

Name	Type	Description
xShopperId	string	Shopper ID for which the suggestions are being generated
query	string	Domain name or set of keywords for which alternative domain names will be suggested

Name	Type	Description
country	string	Two-letter ISO country code to be used as a hint for target region  NOTE: These are sample values, there are many <a href="#">more</a>
city	string	Name of city to be used as a hint for target region
sources	array	Sources to be queried <ul style="list-style-type: none"><li>• <b>CC_TLD</b> - Varies the TLD using Country Codes</li><li>• <b>EXTENSION</b> - Varies the TLD</li><li>• <b>KEYWORD_SPIN</b> - Identifies keywords and then rotates each one</li><li>• <b>PREMIUM</b> - Includes variations with premium prices</li></ul>
tlds	array	Top-level domains to be included in suggestions  NOTE: These are sample values, there are many <a href="#">more</a>
lengthMax	integer	Maximum length of second-level domain
lengthMin	integer	Minimum length of second-level domain
limit	integer	Maximum number of suggestions to return
waitMs	integer	Maximum amount of time, in milliseconds, to wait for responses If elapses, return the results compiled up to that point

Source:

- [domains.js](#), [line 2715](#)

### **tlds()**

Retrieves a list of TLDs supported and enabled for sale

Source:

- [domains.js](#), [line 819](#)

**transferIn(xShopperId, domain, body)**

Purchase and start or restart transfer process

**Parameters:**

Name	Type	Description
xShopperId	string	The Shopper to whom the domain should be transfered
domain	string	Domain to transfer in
body	DomainTransferIn	Details for domain transfer purchase

Source:

- [domains.js, line 1270](#)

**update(domain, xShopperId, body)**

Update details for the specified Domain

**Parameters:**

Name	Type	Description
domain	string	Domain whose details are to be updated
xShopperId	string	Shopper for whom Domain is to be updated. NOTE: This is only required if you are a Reseller managing a domain purchased outside the scope of your reseller account. For instance, if you're a Reseller, but purchased a Domain via http://www.godaddy.com
body	DomainUpdate	Changes to apply to existing Domain

Source:

- [domains.js, line 39](#)

**updateContacts(xShopperId, domain, contacts)**

## Update domain

### Parameters:

Name	Type	Description
xShopperId	string	Shopper for whom domain contacts are to be updated. NOTE: This is only required if you are a Reseller managing a domain purchased outside the scope of your reseller account. For instance, if you're a Reseller, but purchased a Domain via <a href="http://www.godaddy.com">http://www.godaddy.com</a>
domain	string	Domain whose Contacts are to be updated.
contacts	DomainContacts	Changes to apply to existing Contacts

Source:

- [domains.js](#), [line 1857](#)

### validate(body)

Validate the request body using the Domain Purchase Schema for the specified TLD

### Parameters:

Name	Type	Description
body	DomainPurchase	An instance document expected to match the JSON schema returned by <code>./schema/{tld}`</code>

Source:

- [domains.js](#), [line 1180](#)

### verifyEmail(xShopperId, domain)

Re-send Contact E-mail Verification for specified Domain

### Parameters:

Name	Type	Description
xShopperId	string	Shopper for whom domain contact e-mail should be verified. NOTE: This is only required if you are a Reseller managing a domain purchased outside the scope of your reseller account. For instance, if you're a Reseller, but purchased a Domain via <a href="http://www.godaddy.com">http://www.godaddy.com</a>
domain	string	Domain whose Contact E-mail should be verified.

Source:

- [domains.js](#), [line 726](#)