<ResponseCache name="ResponseCache">

<CacheKey>

<KeyFragment ref="request.queryparam.w" />

</CacheKey>

<ExpirySettings>

<TimeoutInSec>600</TimeoutInSec>

</ExpirySettings>

</ResponseCache>

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<ResponseCache async="false" continueOnError="false" enabled="true" name="Response-Cache-1">

<DisplayName>Response Cache 1</DisplayName>

<Properties/>

<CacheKey>

<Prefix/>

<KeyFragment ref="request.uri" />

</CacheKey>

<Scope>Exclusive</Scope>

<ExpirySettings>

<ExpiryDate/>

<TimeOfDay/>

<TimeoutInSec ref="flow.variable.here">300</TimeoutInSec>

</ExpirySettings>

<CacheResource>cache\_to\_use</CacheResource>

<CacheLookupTimeoutInSeconds/>

<ExcludeErrorResponse/>

<SkipCacheLookup/>

<SkipCachePopulation/>

<UseAcceptHeader/>

<UseResponseCacheHeaders/>

</ResponseCache>

<CacheLookupTimeoutInSeconds> element

Specifies the number of seconds after which an unsuccessful cache lookup will be considered a cache miss. If this occurs, flow resumes along the cache-miss path.

<CacheLookupTimeoutInSeconds>30</CacheLookupTimeoutInSeconds>

|  |  |
| --- | --- |
| **Default:** | 30 |
| **Presence:** | Optional |
| **Type:** | Integer |

<CacheResource> element

Specifies the cache where messages should be stored. Omit this element to use the included shared cache. You should specify a CacheResource by name if you wish to be able to administratively clear entries contained in the cache

<CacheResource>*cache\_to\_use*</CacheResource>

|  |  |
| --- | --- |
| **Default:** | N/A |
| **Presence:** | Optional |
| **Type:** | String |

*<ExpirySettings>*/<ExpiryDate> element

Specifies the date on which a cache entry should expire. Use the form mm-dd-yyyy. When present, this element's sibling, <TimeoutInSec>, overrides <ExpiryDate>.

<ExpirySettings>

<ExpiryDate ref="{*date\_variable*}">*expiration\_date*</ExpiryDate>

</ExpirySettings>

|  |  |
| --- | --- |
| **Default:** | N/A |
| **Presence:** | Optional |
| **Type:** | String |

<CacheKey>/<KeyFragment> element

Specifies a value that should be included in the cache key, creating a namespace for matching requests to cached responses.

<KeyFragment ref="*variable\_name*"/>

<KeyFragment>*literal\_string*</KeyFragment>

|  |  |
| --- | --- |
| **Default:** | N/A |
| **Presence:** | Optional |
| **Type:** | N/A |

This can be a key (a static name that you provide) or a value (a dynamic entry set by referencing a variable). All specified fragments combined (plus the prefix) are concatenated to create the cache key.

<KeyFragment>apiAccessToken</KeyFragment>

<KeyFragment ref="request.queryparam.client\_id" />

You use the <KeyFragment> element in conjunction with <Prefix> and <Scope>. For more information, see [Working with cache keys](http://docs.apigee.com/api-services/reference/working-cachekeys).