AsyncTasks API

TotalApi SDK

# **Описание функций**

1. Получение списка выполняемых задач.
2. Управление задачами (запуск/отмена/приостановка/возобновление/удаление).
3. Получение текущего статуса/результата выполнения задачи.
4. Конфигурирование параметров выполнения задач.
5. Генерация событий **AsyncTasks API**.

Для управления любыми асинхронными распределёнными задачами используется универсальное API, называемое **AsyncTasks API**. Данный API является базовым для других API, которые реализуют специфические асинхронные задачи (например, [ReportsAPI](ReportsAPI.docx)).

## Параметры выполнения задач

Эти параметры влияют на процесс выполнения и хранения результатов, но не влияют на данные, возвращаемые этими задачами. Это, например, **время хранения результатов, время кеширования результатов, тайм-аут выполнения** и т.д.

## Идентификатор процесса выполнения задачи - AsyncTaskId

При старте асинхронной задачи метод сервиса сразу отдаёт управление, возвращая идентификатор процесса выполнения задачи. Это **строковое значение**, которое можно использовать для дальнейшей работы с этой задачей (получение текущего статуса выполнения, отмены выполнения, получения результата).

Кроме того, во время выполнения, модуль рассылает сообщения OnProgress в которых в качестве идентификатора будет использоваться [AsyncTaskId.](#h.1fob9te)

# **Бизнес-объекты**

|  |  |  |
| --- | --- | --- |
| AsyncTaskObject | | |
| **Свойство** | **Описание** | **Тип** |
| TaskName | Уникальный идентификатор задачи. По факту - полное наименование класса параметров задачи. | string |
| TaskParams | Параметры задачи. | Dictionary<string, object> |

|  |  |  |
| --- | --- | --- |
| AsyncTaskSettings | | |
| **Свойство** | **Описание** | **Тип** |
| StorageHandlerPeriod | Период работы цикла обслуживания оперативного кэша первого уровня.  Этот цикл очищает кэш от устаревших заданий и отменяет задания, превысившие свой таймаут. | TimeSpan |
| StorePeriod | Время хранения результатов задачи в оперативном кэше первого уровня.  После выполнения задачи, её результат некоторое время хранится на сервере, чтобы результат можно было получить при помощи команды [GetResult](#_GetResult(taskId)__WebApi) [.](#h.qsh70q) После вызова этой команды, в случае если задача завершила своё выполнение, результаты выполнения будут очищены. Результаты также будут очищены автоматически, если команда [GetResult](#_GetResult(taskId)__WebApi)не была вызвана спустя **StorePeriod** после окончания выполнения задачи. | TimeSpan |
| CachingPeriod | Время кеширования результатов задачи. *При повторном вызове задачи возможна отдача результата предыдущего вызова этой задачи с теми же входными параметрами. Время хранения результатов выполнения задач в кэше задаётся этим параметром.* | TimeSpan |
| Timeout | Тайм-аут выполнения задачи.  *Если задача выполняется больше указанного этим параметром промежутка времени она будет автоматически прекращена, путём автоматического вызова метода* [*Cancel*](#_Cancel(taskId)__WebApi)[*.*](#h.2bn6wsx) *При этом статус задачи устанавливается в* Faulted *и код ошибки будет* TimeoutException*.* | TimeSpan |

|  |  |
| --- | --- |
| AsyncTaskStatus | |
| **Свойство** | **Описание** |
| Running | Выполняется. |
| Cancelled | Отменена. |
| Done | Завершена без ошибок. |
| Faulted | Завершена с ошибкой. |

|  |  |  |
| --- | --- | --- |
| AsyncTaskException | | |
| **Свойство** | **Описание** | **Тип** |
| Type | Тип исключения. По факту - полное наименование класса исключения. | string |
| Message | Текст ошибки. | string |

|  |  |  |
| --- | --- | --- |
| AsyncTaskResult | | |
| **Свойство** | **Описание** | **Тип** |
| TaskId | Идентификатор задачи. | [AsyncTaskId](#_Идентификатор_процесса_выполнения) |
| Status | Текущий статус выполнения задачи. | [AsyncTaskStatus](#_AsyncTaskStatus) |
| Result | Результат выполнения задачи. Это произвольный объект – для каждой задачи свой. В случае ошибки – тут объект [AsyncTaskException](#_AsyncTaskException)[.](#h.3j2qqm3) | оbject |

# **Сервисы**

# 

# **AsyncTasks** *Требует* ***ApiKey*** *или* ***AppUser*** *авторизации.*

|  |  |  |
| --- | --- | --- |
| All() WebApi GET: / | | |
| Выдаёт список всех выполняющихся или уже выполненных задач. | | |
| **Параметр** | **Описание** | **Тип** |
| **Результат** | Список задач. | [**AsyncTaskResult**](#_AsyncTaskResult)[[](#h.1y810tw)] |

|  |  |  |
| --- | --- | --- |
| GetResult(…) WebApi GET: /{taskId} | | |
| Получает текущий результат выполнения задачи. | | |
| **Параметр** | **Описание** | **Тип** |
| taskId | Идентификатор задачи. | [**AsyncTaskId**](#_Идентификатор_процесса_выполнения) |
| **Результат** | Текущий результат выполнения задачи. | [**AsyncTaskResult**](#_AsyncTaskResult) |

|  |  |  |
| --- | --- | --- |
| Start(…) WebApi POST**:** / | | |
| Запускает задачу на выполнение. Обработчик задачи указанного типа должен быть зарегистрирован в подсистеме, обслуживающей этот API. В противном случае будет брошено исключение о том, что тип задачи не найден. | | |
| **Параметр** | **Описание** | **Тип** |
| task | Параметры задачи, одновременно задающие и тип задачи. | [**AsyncTaskObject**](#_AsyncTaskObject) |
| **Результат** | Идентификатор процесса выполнения задачи. | [**AsyncTaskId**](#_Идентификатор_процесса_выполнения) |

|  |  |  |
| --- | --- | --- |
| Cancel(…) WebApi DELETE**:** /{taskId} | | |
| Отменяет выполнение задачи. В случае вызова метода [**Cancel**](#_Cancel(taskId)__WebApi)будет сделана попытка «мягко» прервать процесс: статус задачи изменится с Runningна Cancelled[,](#h.26in1rg) а метод, выполняющий процесс, будет уведомлен о том, что пользователь процесс прервал и, если в этом методе поддерживается обработка таких уведомлений, то процесс будет прерван штатным способом самого метода выполнения. Однако в случае, если процесс завис или обработчик не поддерживает штатное прерывание – ничего не произойдёт. | | |
| **Параметр** | **Описание** | **Тип** |
| taskId | Идентификатор задачи. | [**AsyncTaskId**](#_Идентификатор_процесса_выполнения) |
| **Результат** | нет | void |

|  |  |  |
| --- | --- | --- |
| GetSettings() WebApi GET**:** /Settings | | |
| Получает [параметры выполнения задач](#_AsyncTaskSettings) по умолчанию. | | |
| **Параметр** | **Описание** | **Тип** |
| **Результат** | Параметры выполнения задач. | [**AsyncTaskSettings**](#_AsyncTaskSettings) |

|  |  |  |
| --- | --- | --- |
| SetSettings(…) WebApi PUT**:** /Settings | | |
| Сохраняет [параметры выполнения задач](#_AsyncTaskSettings) по умолчанию. | | |
| **Параметр** | **Описание** | **Тип** |
| taskSettings | Параметры выполнения задач. | [**AsyncTaskSettings**](#_AsyncTaskSettings) |

# [**Events**](https://docs.google.com/document/d/1rTiW7ghno8E9AZKWC7dOzCQ8G8Cdsx87Hth66S08rF0/edit#heading=h.4ilxt58idshs) *Требует авторизации.*

* [OnAsyncTaskStarted](#_OnAsyncTaskStarted)
* [OnAsyncTaskFinished](#_OnAsyncTaskFinished)
* [OnProgress](EventsAPI.docx#OnProgress)

# Приложение 1. Список событий AsyncTask API

|  |  |  |
| --- | --- | --- |
| OnAsyncTaskStarted | | |
| Событие, посылаемое при запуске задачи. | | |
| **Свойство** | **Описание** | **Тип** |
| taskId | Идентификатор задачи. | [**AsyncTaskId**](#_Идентификатор_процесса_выполнения) |
| task | Параметры задачи, одновременно задающие и тип задачи. | [**AsyncTaskObject**](#_AsyncTaskObject) |

|  |  |  |
| --- | --- | --- |
| OnAsyncTaskFinished | | |
| Событие, посылаемое при запуске задачи. | | |
| **Свойство** | **Описание** | **Тип** |
| taskId | Идентификатор задачи. | [**AsyncTaskId**](#_Идентификатор_процесса_выполнения) |
| task | Параметры задачи, одновременно задающие и тип задачи. | [**AsyncTaskObject**](#_AsyncTaskObject) |
| Status | Статус выполнения задачи. | [**AsyncTaskStatus**](#_AsyncTaskStatus) |

# 