

Inlearno Booking API

Содержание

- Содержание
- Термины и определения
- Статус документа и API
 - Тестирование API
- Действия сторон, предшествующие обмену данными
- Общие правила обмена
 - Действия при обмене данными
- Получение данных о событиях
 - Даты и время проведения
 - Главная фотография мероприятия
 - Дополнительные фотографии к мероприятию
 - Регионы, для которых доступно мероприятие
 - Полный пример передаваемых данных
- Получение стоимости заявки
 - Параметры запроса
 - Формат ответа
- Получение списка свободных мест
- Создание нового заказа
- Обновление состояния заявки

Термины и определения

| Термин | Значение термина |
|----------------------------------|--|
| Компания inLearno | ООО "Образовательные технологии" |
| inLearno Booking, Booking | Система бронирования inLearno |
| inLearno Booking API, API | Интерфейс обмена данными с Магазином. Вызовы интерфейса, как правило, располагаются в пространстве адресации http://booking.inlearno.com/api/ |
| Магазин | Интернет-сайт/мобильное приложение или любой другой клиентский модуль, предназначенный для отображения каталога событий (мероприятий) и предоставления пользователям функций для заказа мероприятий |
| Пользователь | Любой посетитель Магазина, совершивший одно из действий, приводящих к его взаимодействию с системой Магазина и/или с системой inLearno Booking |
| Событие (Мероприятие) | Экскурсия, мастер-класс, квест, занятие, кружок, курс, лагерь или любое другое мероприятие, проводимое Организатором, и имеющееся в базе данных Booking. Каждое событие имеет ряд параметров (полей), задаваемых Booking. При изменении набора таких параметров, происходит изменение настоящей документации |
| Организатор События | Юридическое или физическое лицо, предоставляющее услуги по проведению События, заключившее агентский договор с компанией inLearno |
| Заявка (Заказ) | Сформированный Магазином запрос на бронирование События. После создания, заявка обрабатывается системой inLearno Booking и специалистами компании inLearno, с привлечением Организатора, и, в некоторых случаях, с участием Пользователя |
| Состояние заявки (Статус заявки) | Стадия обработки Заявки в системе inLearno Booking и компании inLearno, а также у Организатора и/или Пользователя |
| Выездное мероприятие | Мероприятие, проводимое не на территории Организатора События. При заказе такого мероприятия Пользователем, необходимо предоставить ему возможность указать адрес проведения мероприятия |
| Индивидуальное мероприятие | Мероприятие, на которое Организатором События установлена возможность покупки единичных билетов. Каждый билет покупается по фиксированной ставке. |
| Групповое мероприятие | Мероприятие, на которое Организатором События установлена возможность покупки билетов только на группу посетителей, не менее минимальной. Но и не более максимального. |

| | |
|-----------------|--|
| Наборная группа | Мероприятие, проводимое Организатором События в виде группового, но на данное мероприятие существует возможность покупки произвольного доступного количества билетов Пользователем (как для Индивидуального мероприятия) |
|-----------------|--|

Статус документа и API

По мере развития API, в него добавляется новая и изменяется существующая функциональность, но все изменения сохраняют обратную совместимость. Т.е. все описанные в настоящем документе методы и свойства будут работать как описано. В случае необходимости внесения изменений, нарушающий этот принцип, Магазин заблаговременно будет уведомлен с целью доработки протокола обмена на своей стороне.

ВАЖНО Все поля и методы, которые не описаны в документе могут быть изменены без предупреждения.

Тестирование API

Для тестирования API необходимо использовать имя , чтобы мы могли различать тестовые запросы от реальных.

Также для целей тестирования существуют специальные тестовые занятия, список которых можно [посмотреть в букинге](#), партнер "-".

Действия сторон, предшествующие обмену данными

До обмена данными, сторонам необходимо пройти следующие действия:

1. Заключение договора между компанией inLearno и Магазином. Для получения формы договора необходимо написать письмо на адрес ak@inlearno.ru
2. Сотрудники Магазина заполняют [Анкеты](#) (скачивание по ссылке - файл Анкета Магазина.docx) и направляют её на адрес ak@inlearno.ru. Критичными для начала обмена данными являются сведения о URL обработчика на стороне Магазина и координаты ответственных лиц Магазина
3. Сотрудники компании inLearno прописывают данные Магазина в системе inLearno Booking и сообщают о готовности ответственному сотруднику Магазина, указанному в Анкете
4. Стороны проводят тестирование обмена данными

Общие правила обмена

В процессе обмена данными участвуют 2 стороны - зарегистрированный в системе Магазин и InLearno Booking. Инициатором обмена, в общем случае, может выступать любая из сторон. Конкретно для каждого действия будет указано - какая из сторон может инициировать процесс обмена данными.

Обмен происходит по протоколу HTTP. Запросы на получение информации посылаются методом GET. Запросы на обновление информации посылаются методом POST.

Если одна из сторон не смогла принять запрос (вернулся любой HTTP-ответ, отличный от 200), вторая сторона обязана повторить свое сообщение через некоторое время.

Обмен данными происходит в формате [JSON](#). В данный момент поддерживается исключительно кодировка UTF-8.

Каждый POST запрос строится по следующему принципу:

Пример POST-запроса

```
{
  security: "...",
  shop_id: 1,
  some_entity: {
    ....
  }
}
```

Где

- security - секретный ключ, выдаваемый Магазину при регистрации,
- shop_id - идентификатор Магазина, к которому или который делает запрос,
- some_entity - сущность, по которой обновляется информация.

При направлении запросов со стороны Inlearno Booking конкретному Магазину, используются security и shop_id этого Магазина.

При направлении запросов со стороны Магазина, используются security и shop_id этого Магазина. Все запросы, которые содержат security и shop_id должны производиться с сервера Магазина. Запросы с клиентской части НЕ ДОПУСКАЮТСЯ. В случае невыполнения данного правила, все риски нарушения конфиденциальности передаваемой информации лежат на Магазине.

ПРИМЕР Магазин реализует заказ Мероприятий через собственное мобильное приложение. В таком случае запрещается передача, хранение, обработка в таком приложении данных security и shop_id. Приложение должно обратиться на сервер Магазина, который и инициирует обмен данными с Inlearno Booking.

Пример ответа

```
{
  err_code: ...,
  data: {
    ...
  }
}
```

Код ошибки, если она произошла, должен быть передан в ключе err_code. Каждая порция данных передается в блоке data.

Действия при обмене данными

Существует несколько основных ситуаций, когда необходимо запустить процесс обмена данными:

1. Передача новых (обновленных) данных о Событиях из inLearno Booking в Магазин. Иницируется каждый раз при появлении новых данных о Событиях на стороне inLearno Booking
2. Получение Магазином актуальной цены События из inLearno Booking. Иницируется Магазином при открытии Пользователем карточки События в Магазине
3. Получение списка свободных мест на Событие типа "наборная группа". Иницируется Магазином при открытии Пользователем карточки События, имеющей тип "наборная группа". Используется только, если Магазин работает с наборными группами
4. Создание новой Заявки. Иницируется Магазином в случае бронирования/покупки Пользователем в Магазине одного из Событий
5. Обновление состояния Заявки. Иницируется inLearno Booking в случае изменения параметров Заявки, включая ее статус


Получение данных о событиях

Все параметры и описания каждого события определяются организаторами в InLearno Booking.

При изменении сведений InLearno Booking организует отправку оповещений для всех магазинов-участников. Конкретный URL, на который будут отправлены данные, предоставляется Магазином в заполненной Анкете и прописывается сотрудниками inLearno в свойства магазина. URL должен быть представлен до тестового запуска обмена данными между Магазином и Inlearno Booking.

Сведения отправляются при любом изменении данных в карточке события. Каждое событие отправляется отдельным сообщением на URL, предоставленный партнером. Все параметры передаются методом POST

| Ключ | Тип значения | Назначение | Примечание |
|----------|--------------|-----------------------|---|
| event | array | Обновляемая сущность | Используется Магазином для определения какому объекту соответствуют данные |
| id | integer | Идентификатор события | Уникальное значение идентификатора события. Значение может быть только целым, положительным числом. В дальнейшем, при обмене данными необходимо ссылаться именно на это значение идентификатора |
| name | string (100) | Название мероприятия | Строковое поле, максимальный размер - 100 знаков. |
| announce | text | Анонс мероприятия | Краткое описание мероприятия. Максимальное количество символов 65535. |

| | | | |
|----------------------|---------|-----------------------------------|--|
| description | text | Описание мероприятия | Полное описание мероприятия. Максимальное количество символов 65535. |
| special | text | Особые условия | Содержит текст особых условий для мероприятия. Может быть пустым. |
| event_rules | text | Правила участия | Форматированный текст (HTML) |
| child_advantage | text | Что получает ребенок | Для удобства работы с описанием, полное описание разбито на части, которые выводятся на странице вместе. Форматированный текст (HTML) |
| conditions | text | Условия участия | Для удобства работы с описанием, полное описание разбито на части, которые выводятся на странице вместе. Форматированный текст (HTML) |
| teachers | text | Преподаватели | Для удобства работы с описанием, полное описание разбито на части, которые выводятся на странице вместе. Форматированный текст (HTML) |
| exercise_description | text | Как проходит занятие | Для удобства работы с описанием, полное описание разбито на части, которые выводятся на странице вместе. Форматированный текст (HTML) |
| escort | text | Сопровождающие | Для удобства работы с описанием, полное описание разбито на части, которые выводятся на странице вместе. Форматированный текст (HTML) |
| special_conditions | text | Особые условия | Для удобства работы с описанием, полное описание разбито на части, которые выводятся на странице вместе. Форматированный текст (HTML) |
| programm | text | Программа смены | Для удобства работы с описанием, полное описание разбито на части, которые выводятся на странице вместе. Форматированный текст (HTML) |
| living_conditions | text | Условия проживания | Для удобства работы с описанием, полное описание разбито на части, которые выводятся на странице вместе. Форматированный текст (HTML) |
| documents_list | text | Перечень документов | Для удобства работы с описанием, полное описание разбито на части, которые выводятся на странице вместе. Форматированный текст (HTML) |
| transfer | text | Трансфер | Для удобства работы с описанием, полное описание разбито на части, которые выводятся на странице вместе. Форматированный текст (HTML) |
| length | integer | Продолжительность занятия | |
| length_unit | integer | Единица продолжительности занятия | <p>Может принимать следующие значения (здесь и далее - приводим значения в виде констант, которые можно использовать в PHP-коде)</p> <pre> const CLENGTH_UNIT_MINUTE = 1; // минуты const CLENGTH_UNIT_HOUR = 2; // часы const CLENGTH_UNIT_DAY = 3; // дни const CLENGTH_UNIT_WEEK = 4; // недели const CLENGTH_UNIT_MONTH = 5; // месяцы const CLENGTH_UNIT_YEAR = 6; // годы </pre> |
| period_start | integer | Отсрочка начала занятий | <p>Используется для сдвига первого доступного для заказа занятия.</p> <p>Например, 2 - означает, что ближайшее доступное занятие может быть не ранее чем через 2 дня. Т.е. если у нас по расписанию занятие доступно каждый день, то завтра и послезавтра заказать занятие нельзя.</p> |
| state | integer | Статус занятия | <p>Полный список статусов занятий:</p> <pre> const CSTATE_MODERATED = 2; // событие на модерации const CSTATE_PUBLISHED = 3; // опубликованное событие - только такие события может быть опубликовано магазине. const CSTATE_DELETED = 4; // событие удалено </pre> <p>В каталоге предложений можно показывать только события со статусом CSTATE_PUBLISHED, все остальные статусы информационные, и фактически означают не возможность заказать событие (при попытке заказать события со статусом, отличным от CSTATE_PUBLISHED, сервис вернет ошибку).</p> <p> Статус занятия не влияет на доступность мероприятия для пользователя. Мероприятие в любом статусе должно быть доступно для просмотра пользователем по прямой ссылке. Удаление мероприятия происходит только установкой статуса "Удалено".</p> |

| | | | |
|-------------|---------|--|--|
| member_type | integer | Тип участия | <p>Может принимать следующий список значений:</p> <pre> const CMEMBER_TYPE_GROUP = 1; // только групповое участие const CMEMBER_TYPE_SPECIAL = 2; // только индивидуальное участие const CMEMBER_TYPE_GROUP_OR_SPECIAL = 3; // есть возможность как групповых, так и индивидуальных занятий </pre> |
| persons | integer | Максимальное количество участников группы | <p>Определяет максимальный размер группы, который может присутствовать на событии одновременно. Может быть только положительным, целым числом. Рекомендуется проверять (программно) заказ клиента до отправки его на обработку в бучинг и просить клиента исправить ошибку. Не валидные заявки будут все равно отфильтрованы.</p> |
| min_persons | integer | Минимальное количество участников группы | <p>Определяет минимальный размер группы, который может присутствовать на событии одновременно. Может быть только положительным, целым числом. Рекомендуется проверять (программно) заказ клиента до отправки его на обработку в бучинг и просить клиента исправить ошибку. Не валидные заявки будут все равно отфильтрованы.</p> |
| type | array | Тип события | <p>Может иметь несколько значений, поэтому передается в виде массива (даже если всего элемент). Значения массива соответствуют битовым маскам:</p> <pre> const STYPE_MASTER_CLASS = 0b0000000000000001; // мастер-класс const STYPE_EXCURSION = 0b0000000000000010; // экскурсия const STYPE_TRAINING = 0b0000000000000100; // тренинг const STYPE_HOLIDAY = 0b0000000000001000; // праздничное мероприятие const STYPE_SUMMER_VACATION = 0b0000000000100000; // летний отдых const STYPE_CAMPING_TRIP = 0b0000000001000000; // туристический поход const STYPE_WEEKEND_TOUR = 0b0000000010000000; // тур выходного дня const STYPE_QUEST_GAME = 0b0000000100000000; // игра, квест const STYPE_TRAINING_COURSE = 0b0000001000000000; // учебный курс const STYPE_LECTURE = 0b0000010000000000; // лекция const STYPE_SPORTING_EVENT = 0b0000100000000000; // спортивное мероприятие const STYPE_EXHIBITION = 0b0001000000000000; // выставка const STYPE_HISTORICAL_RECONSTRUCTION = 0b0010000000000000; // историческая реконструкция const STYPE_GAME_SPACE = 0b0100000000000000; // игровое пространство const STYPE_B_DAY = 0b1000000000000000; // день рождения </pre> |
| age_from | integer | Возраст, на который рассчитано мероприятие | Значение от |

| | | | |
|----------------------|--|--|---|
| age_to | integer | Возраст, на который рассчитано мероприятие | Значение до |
| location | text | Место проведения | Если мероприятие не выездное, то будет содержаться адрес (в виде строки), по которому проводится мероприятие. Данный адрес может быть использован для отметки на карте. |
| event_dates | array | Даты и время проведения | Подробное описание элементов находится ниже |
| event_dates_combined | array | Даты и время проведения для наборных групп | Структура и значения полей полностью совпадают с event_dates |
| price | string | Минимальная цена на мероприятие | Может использоваться в представлении "Цена от ...". Реальная цена, будет рассчитана по данным, представленным пользователем в заявке. Для получения цены заявки необходимо использовать вызов getPrice |
| is_deleted | "Y" (удалено) или "N"(не удалено) | Признак "Удалено" | Может иметь значения "Y" (удалено) или "N"(не удалено). Мероприятие, помеченное как удаленное не должно быть доступно даже для просмотра пользователем. Мероприятие, на которое уже нельзя сделать заказ, будет иметь соответствующий статус. |
| is_external | "Y" "N" | Признак выездного мероприятия | Мероприятие может проводиться на территории заказчика, в таком случае в ключе будет передано значение "Y". Если мероприятие происходит только на территории организатора - это не выездное мероприятие, соответственно значение будет "N". |
| image | array | Главная промо фотография мероприятия | Ключи элемента будут описаны ниже |
| images | array | Дополнительные фотографии мероприятия | Ключи элемента будут описаны ниже |
| regions | array | Регионы, для которых доступно мероприятие | Ключи элемента будут описаны ниже. |
| has_date | "Y" "N" | Имеются ли у мероприятия даты в календаре | Используется для облегчения процесса отбора событий. Если у события (мероприятия) нет дат проведения в будущем, такое событие, например, не должно попадать в каталог событий, но должно быть доступно по прямой ссылке |

Даты и время проведения

Каждый элемент массива содержит следующий набор

| Ключ | Тип значения | Назначение | Примечание |
|-------------|--------------|--|--|
| date | | дата и время начала мероприятия | в формате Y-m-d H:i:s |
| date_finish | | дата и время окончания мероприятия | в формате Y-m-d H:i:s |
| slot | | количество доступных для заказа мест и/или билетов | <p>Необходимо пояснить что такое slot. По сути это число определяет сколько человек одновременно может посетить мероприятие, оно формируется с одной стороны из ограничения количества одновременно посещающих людей, с другой - ограничение на количество "экскурсоводов". Каждый заказ уменьшает количество доступных для блокировки слотов. Если количество доступных слотов равно нулю - на данную дату-время не возможна запись.</p> <p>Другими словами слоты предотвращают возможность овербукинга.</p> <p>Данное поле является справочным, и может использоваться партнером только для предварительного информирования клиента о доступности мероприятия для заказа. В простом случае можно не использовать, поскольку проверка на наличие свободных мест происходит в момент бронирования.</p> |

Пример передаваемых данных

```
"event_dates": [  
  {  
    "slot": "1",  
    "date": "2015-06-29 10:00:00",  
    "date_finish": "2015-06-29 11:00:00"  
  },  
  {  
    "slot": "1",  
    "date": "2015-07-06 10:00:00",  
    "date_finish": "2015-07-06 11:00:00"  
  }  
],
```

Главная фотография мероприятия

| Ключ | Тип значения | Назначение | Примечание |
|------------------------|--------------|---|-------------------|
| big_thumb_http_path | string | Полный путь к фотографии исходного размера | |
| little_thumb_http_path | string | Полный путь к фотографии, используемой в качестве preview | Размер 244 на 159 |
| full_http_path | string | Полный путь к фотографии | |

Пример передаваемых данных

```
"image": {  
  "full_http_path": "http://inlearno.local/images/88dbfba8fo4b76fa42e02b9392e394fc.jpg",  
  "little_thumb_http_path": "http://inlearno.local/images/88dbfba8fo4b76fa42e02b9392e394fc_244x159.jpg",  
  "big_thumb_http_path": "http://inlearno.local/images/88dbfba8fo4b76fa42e02b9392e394fc_big.jpg"  
}
```

Дополнительные фотографии к мероприятию

Обычно используются на странице карточки мероприятия. Представляет собой массив элементов, каждый из которых по своей структуре повторяет описание главной фотографии мероприятия.

Пример передаваемых данных

```
"images": [  
  {  
    "full_http_path":  
    "http://inlearno.local/images/levents/preview/7f/97/7fb7b42a973c1458d549c358a6ee20f7.jpg",  
    "little_thumb_http_path":  
    "http://inlearno.local/images/levents/preview/7f/97/7fb7b42a973c1458d549c358a6ee20f7_244x159.jpg",  
    "big_thumb_http_path":  
    "http://inlearno.local/images/levents/preview/7f/97/7fb7b42a973c1458d549c358a6ee20f7_big.jpg"  
  },  
  {  
    "full_http_path":  
    "http://inlearno.local/images/levents/preview/46/ad/462460eaadb094786480bf5fc64473db.jpg",  
    "little_thumb_http_path":  
    "http://inlearno.local/images/levents/preview/46/ad/462460eaadb094786480bf5fc64473db_244x159.jpg",  
    "big_thumb_http_path":  
    "http://inlearno.local/images/levents/preview/46/ad/462460eaadb094786480bf5fc64473db_big.jpg"  
  }  
]
```

Регионы, для которых доступно мероприятие

Представляет собой массив элементов следующей структуры:

| Ключ | Тип значения | Назначение | Примечание |
|------|--------------|----------------------------------|--|
| id | integer | Уникальный идентификатор региона | Может использоваться при обновлении данных |
| text | string | Название региона | |

Пример передаваемых данных

```
"regions": [  
  {  
    "id": "108569",  
    "text": "Самарская область"  
  },  
  {  
    "id": "104885",  
    "text": "Московская область"  
  }  
]
```

Полный пример передаваемых данных

Пример передаваемых данных

```
{  
  security: "ZDSfs^fd",  
  shop_id: 1,  
}
```



```
event: {
  "id": "5",
  "length_unit": "3",
  "period_start": "2",
  "period_pay": "1",
  "period_notify": "10",
  "state": "3",
  "min_persons": "0",
  "persons": "5",
  "name": "Quia voluptatem qui.",
  "announce": "Vel quis repudiandae qui. Dignissimos adipisci consequuntur ipsa cumque blanditiis qui.",
  "description": "Aut eligendi veniam repellat quas. Quia rerum aut animi corrupti commodi. Rem maxime ab
  aspernatur voluptate provident. Et fugit fugit praesentium voluptatibus sit cupiditate sed.",
  "length": "0",
  "member_type": "2",
  "type": [
    32
  ],
  "contacts": "Iusto eaque autem aut quia eos magni recusandae. Libero qui id ad animi autem. Occaecati id sunt
  blanditiis deleniti quaerat aut. Voluptatem et ea quam nihil omnis omnis quia.",
  "special": "Tempore ut facere voluptatem architecto. Modi non similique voluptatem laboriosam aut totam non.
  Ab nostrum facere et laudantium eum. Aperiam ex voluptatem voluptas rerum cum voluptate impedit.",
  "age_from": "4",
  "age_to": "11",
  "location": "888931, Саратовская область, город Люберцы, пер. Сталина, 79",
  "price": "115",
  "is_deleted": "N",
  "is_external": "N",
  "event_rules": "",
  "has_date": "Y",
  "child_advantage": "<p>Что получает ребенок</p>\n",
  "conditions": "<p>Условия участия</p>\n",
  "teachers": "<p>Преподаватели</p>\n",
  "exercise_description": "<p>Как проходит занятие</p>\n",
  "escort": "",
  "special_conditions": "",
  "programm": "<p>Программа смены</p>\n",
  "living_conditions": "<p>Условия проживания</p>\n",
  "documents_list": "",
  "transfer": "",
  "image": {
    "full_http_path": "http://inlearno.local/images/88dbfba8fo4b76fa42e02b9392e394fc.jpg",
    "little_thumb_http_path": "http://inlearno.local/images/88dbfba8fo4b76fa42e02b9392e394fc_244x159.jpg",
    "big_thumb_http_path": "http://inlearno.local/images/88dbfba8fo4b76fa42e02b9392e394fc_big.jpg"
  },
  "images": [
    {
      "full_http_path":
"http://inlearno.local/images/levents/preview/7f/97/7fb7b42a973c1458d549c358a6ee20f7.jpg",
      "little_thumb_http_path":
"http://inlearno.local/images/levents/preview/7f/97/7fb7b42a973c1458d549c358a6ee20f7_244x159.jpg",
      "big_thumb_http_path":
"http://inlearno.local/images/levents/preview/7f/97/7fb7b42a973c1458d549c358a6ee20f7_big.jpg"
    },
    {
      "full_http_path":
"http://inlearno.local/images/levents/preview/46/ad/462460eaadb094786480bf5fc64473db.jpg",
      "little_thumb_http_path":
"http://inlearno.local/images/levents/preview/46/ad/462460eaadb094786480bf5fc64473db_244x159.jpg",
```

```
    "big_thumb_http_path":  
    "http://inlearno.local/images/levents/preview/46/ad/462460eaadb094786480bf5fc64473db_big.jpg"  
  },  
  ],  
  "event_dates": [  
    {  
      "slot": "1",  
      "date": "2015-06-29 10:00:00",  
      "date_finish": "2015-06-29 11:00:00",  
      "slot_block": "0"  
    },  
    {  
      "slot": "1",  
      "date": "2015-07-06 10:00:00",  
      "date_finish": "2015-07-06 11:00:00",  
      "slot_block": "0"  
    }  
  ],  
  "event_dates_combined": [  
    {  
      "slot": "1",  
      "date": "2015-06-29 10:00:00",  
      "date_finish": "2015-06-29 11:00:00",  
      "slot_block": "0"  
    }  
  ],  
  "regions": [  
    {  
      "id": "108569",  
      "text": "Самарская область"  
    },  
    {  
      "id": "104885",  
      "text": "Московская область"  
    }  
  ]  
}
```

```
]
}
}
```

Получение стоимости заявки

URL для запроса - <http://booking.inlearno.local/api/getPrice/>

Метод получения - GET

Параметры запроса

| Ключ | Тип | Обязательное поле | Значение | Примечание |
|---------------|---------|-------------------|-----------------------------|--|
| shop_id | integer | Да | Идентификатор магазина | Выдается при подключении магазина |
| security | string | Да | Секретный ключ магазина | |
| event_id | integer | Да | Идентификатор занятия | Занятия, для которого необходимо рассчитать цену |
| date_start | string | Да | Дата и время начала события | В формате YYYY-MM-DD H:i:oo (секунды не являются значимыми, потому для соблюдения формата рекомендуем указывать их как 00) |
| tickets_count | integer | Да | Количество билетов в заявке | |
| is_combined | 0 1 | Нет | Признак наборной группы | Для расчета цены занятия по тарифу "наборная группа". По умолчанию "false" |

Пример запроса

```
shop_id:1,
security: "ZDSfs^fd",
event_id:10,
date_start: "2015-03-20 11:23",
tickets_count:100
```

Формат ответа

Ответ InLearno Booking содержит следующие поля

| Ключ | Тип | Значение | Примечание |
|-----------|------------------|---------------------------------------|---|
| main | array | Секция цены обычного занятия | |
| combined | array | Цена и места для наборной группы | |
| price | array | Массив, содержащие данные по цене | |
| value | float integer | Запрошенная цена | |
| currency | string | Признак валюты | В данный момент существует только одна валюта "RUB" |
| vacancies | array | Количество мест для наборной группы | Присутствует только в секции наборной группы |
| total | integer | Всего мест на данное мероприятие | На это дату и время |
| free | integer | Свободно для заказа на текущий момент | |

Пример ответа

```
{
  err_code:0,
  data: {
    main:{
      price:{
        value:10000,
        currency:"RUB"
      }
    },
    combined:{
      price:{
        value:10000,
        currency:"RUB"
      },
      vacancies:{
        total:50,
        free:10
      }
    }
  }
}
```

Получение списка свободных мест

Используется для получения списка свободных мест для мероприятий с наборными группами за указанный период.

URL <http://booking.inlearno.local/api/getCombinedVacancies/>

Метод получения - GET

Параметры запроса

| Ключ | Тип | Значение | Примечание |
|-------------|---------|----------------------------------|--|
| shop_id | integer | Идентификатор магазина | |
| security | string | Секретный код | Совпадает с кодом, выданным партнеру при регистрации. Используется для валидации партнером факта, что запрос пришел именно от Inlearno Booking API |
| date_start | string | Дата начала периода | В формате YYYY-MM-DD. Обязательный параметр |
| date_finish | string | Дата конца периода | В формате YYYY-MM-DD. Обязательный параметр |
| event_id | integer | Уникальный идентификатор события | Не обязательный параметр. Позволяет получить информацию по конкретному событию |

Формат ответа

| Ключ | Тип | Значение | Примечание |
|----------|---------|---------------------------------------|-------------------------------|
| date | string | Дата и время начала события | В формате YYYY-MM-DD H:i:s |
| event_id | integer | Уникальный идентификатор события | |
| total | integer | Всего мест на данное мероприятие | На это дату и время |
| free | integer | Свободно для заказа на текущий момент | На момент формирования ответа |

Пример ответа

```
{
  err_code: 0,
  data: [
    {
      date: "2015-07-06 10:00:00",
      event_id: 10,
      total: 21,
      free: 13
    },
    {
      date: "2015-07-06 12:00:00",
      event_id: 10,
      total: 21,
      free: 21
    }
  ]
}
```

Обращаем внимание, что на одни сутки для одного события может быть несколько времен проведения.

Создание нового заказа

URL <http://booking.inlearno.com/api/createEventRequest>

Параметры передаются методом POST

ВАЖНО

Оформление заказа на сайте Магазина, как правило, инициируется Пользователем путем нажатия соответствующей кнопки "Забронировать"/"Купить".

После инициирования Пользователем оформления заказа, сайт Магазина должен получить у Пользователя необходимые для заказа данные (как правило, путем предоставления Пользователю возможности заполнения данных в поля на сайте Магазина). Минимальный набор данных составляет поля раздела "Данные клиента", приведенные в таблице ниже, а также, для наборных групп, поля раздела "Данные о детях".

После заполнения Пользователем и проверкой Магазином корректности (с точки зрения формата) вводимых данных, Магазин, как правило, предоставляет Пользователю возможность перейти к следующему шагу путем нажатия кнопки "Забронировать"/"Оплатить".

Заказ может оформляться в нескольких вариантах:

1. Заказ с одновременной оплатой банковской картой через Систему бронирования (кнопка "Купить").
Для занятий, не требующих подтверждения участия (бронирования), то есть, покупка осуществляется сразу после оплаты. Такой заказ оформляется, если поле "Тип участия" (member_type) у События равно const MEMBER_TYPE_SPECIAL = 2 (только индивидуальное).
В этом случае Магазин выводит в iframe платежную страницу Системы бронирования, в которой Пользователем осуществляется процесс оплаты заказа.
2. Заказ без оплаты (только бронирование, кнопка "Забронировать")
Для занятий, требующих предварительной записи (бронирования мест).
Такой заказ оформляется, если поле "Тип участия" (member_type) у События является Групповым или Индивидуальным.
После нажатия Пользователем на кнопку "Забронировать", сайт Магазина отправляет заказ в Систему бронирования.
3. Информационное партнерство (кнопка "Оставить заявку").
Бронирование и оплата не производится, пользователь только оставляет заявку, а Магазин передает эту заявку Партнеру.

ЗАМЕЧАНИЕ Если занятие является бесплатным, то форма оплаты не вызывается.

Рекомендуется после завершения оформления заказа Пользователем, предоставить последнему на сайте Магазина информационное сообщение об успешном оформлении заказа и дальнейших действиях.

| Ключ | Тип | Значение | Обязательное | Примечание |
|------|-----|----------|--------------|------------|
|------|-----|----------|--------------|------------|

| | | | | |
|---|------------|---|---------------------|--|
| event_id | integer | Уникальный идентификатор события | Да | |
| date_start | string | Дата и время начала события | Да | В формате YYYY-MM-DD H:i:s. Должно быть передано в значении, в котором пришло из event_dates или event_dates_combined |
| date_finish | string | Дата и время окончания события | Да | В формате YYYY-MM-DD H:i:s. Должно быть передано в значении, в котором пришло из event_dates или event_dates_combined |
| Параметры заказа | | | | |
| id | integer | Идентификатор заказа в магазине партнера | Да | В дальнейшем по этому идентификатору будут присылаться уведомления о смене статуса заказа, так же этот идентификатор участвует в формировании формы оплаты (см. ниже) |
| created_ts | string | Время создания заказа в магазине партнера | Да | В формате YYYY-MM-DD H:i:s |
| tickets_count | integer | Количество заказанных билетов | Да | Совпадает с кодом, выданным партнеру при регистрации. Используется для валидации партнером факта, что запрос пришел именно от Inlearno Booking API |
| is_preorder | "Y" "N" | Признак предзаказа | Нет | Указывает что клиент делает предзаказ для данного мероприятия (позволяет не бронируя места получить информацию о событии). По умолчанию считается что производится заказ. |
| is_combined | "Y" "N" | Сборная группа | Нет | Указывает что клиент выбрал для данного мероприятия наборную группу. Т.е. при расчете будет осуществлен подбор выбран тариф наборных групп. По умолчанию считается что заказывается обычное занятие. |
| location | string | Место проведения | Только для выездных | Для выездных мероприятий клиент указывает адрес, по которому будет проводиться мероприятие |
| Данные клиента | | | | |
| name | string | Имя клиента | Да | |
| phone | string | Телефон для связи | Да | |
| email | string | Почта клиента | Да | |
| user_comment | text | Комментарий к заказу | Нет | |
| Данные о детях (только для сборных групп) | | | | |
| children | array | Массив, в котором передаются информация о детях | Да | Используется только для сборных групп. Количество элементов массива должно соответствовать количеству заказываемых билетов |
| name | string | ФИО ребенка | Да | |
| age | integer | Возраст ребенка | Да | |

Ответ

| Ключ | Тип | Значение | Примечание |
|---------|--------|--------------------------|---|
| pay_url | string | Адрес странице с оплатой | Может использоваться для моментальной оплаты клиентом своего заказа. По данному URL будет сформирована страница с оплатой заказа клиента. |

Пример ответа

```
{
  err_code:0,
  data: {
    pay_url: "http://booking.inlearno.com/api/payFromSite?site_bill_id=42&shop_id=1"
  }
}
```

Обновление состояния заявки

Инициатором обмена является Booking. Обновление присылается каждый раз, когда происходят изменения значимых полей заявки.

URL - предоставленный партнером при регистрации

Метод POST

Параметры запроса

| Ключ | Тип | Значение | Примечание |
|---------------|---------|--|---|
| request | array | Массив, в котором передаются значения состояния заявки | См. пример запроса |
| external_id | string | Идентификатор заявки в системе партнера | |
| event_id | integer | Идентификатор мероприятия | |
| date_start | string | Дата и время начала события | В формате YYYY-MM-DD H:i:s. Может измениться в процессе согласования события с клиентом |
| date_finish | string | Дата и время окончания события | В формате YYYY-MM-DD H:i:s. Может измениться в процессе согласования события с клиентом |
| tickets_count | integer | Количество билетов | Может измениться в процессе согласования события с клиентом |
| price | float | Сумма заявки | Текущая сумма заказа |
| state | integer | Статус заявки | Заявка может находиться в одном из нижеперечисленных статусов <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"><pre>const STATE_NEW = "new"; // Новая заявка const STATE_PENDING = "pending"; // Ждет одобрения const STATE_PAID = "paid"; // Оплачена const STATE_CANCELLED = "cancel"; // Отменена const STATE_APPROVED = "approve"; // Одобрена, ждет оплаты const STATE_NOMONEY = "nomoney"; // Прошел срок оплаты заявки const STATE_RETURN = "return"; // Возврат средств const STATE_HOLD = "hold"; // Заявка на ожидании const STATE_FINISH = "finish"; // Заявка полностью выполнена const STATE_PREORDER = "preorder"; // Предварительная заявка</pre></div> |
| pay_url | string | Адрес страницы оплаты | Может использоваться для моментальной оплаты клиентом своего заказа. По данному URL будет сформирована страница с оплатой заказа клиента. Предоставляется, если оплата заявки уже возможна. |

Пример запроса

```
{
  security: "ZDSfs^fd",
  shop_id: 1,
  request: {
    external_id: 42,
    event_id: 10,
    date_start: "2015-06-29 10:00:00",
    date_finish: "2015-06-29 11:00:00",
    tickets_count: 12,
    price: 1000,
    pay_url: "http://booking.inlearno.com/api/payFromSite?site_bill_id=42&shop_id=1"
  }
}
```

Любой ответ с кодом 200 будет принят как положительный