

Описание API сайта Broni.biz

Как настроить интеграцию расписания и заказов Broni.biz с другими сервисами и агрегаторами

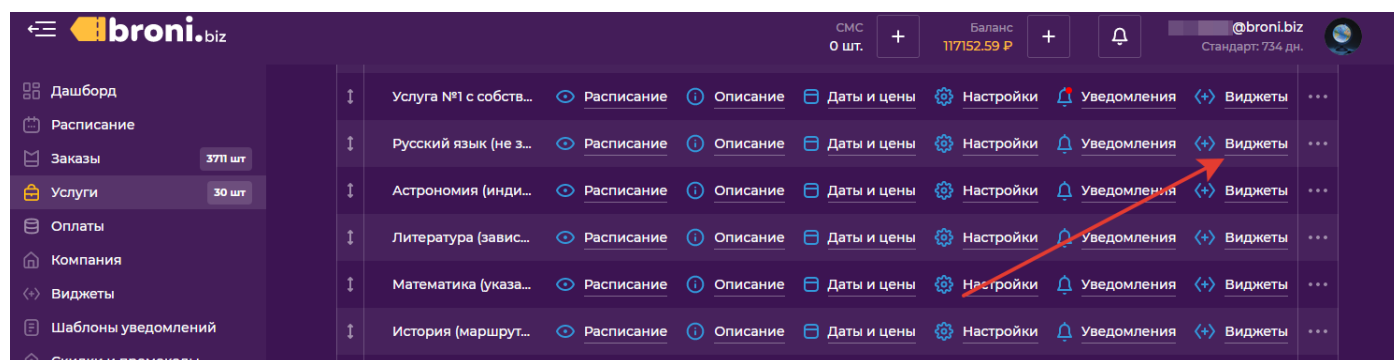
Оглавление

Описание API сайта Broni.biz.....	1
Оглавление.....	2
Где взять ссылки для интеграции?	4
Получение расписания услуги в формате JSON.....	5
Пример ответа и описание полей для индивидуальной записи	5
Список цен, зависящий от кол-ва участников	6
Список цен на конкретную дату и время.....	6
Список цен на конкретную дату и время для тарифов.....	7
Количество отображаемых дней в расписании	7
Price_type в старой версии API.....	8
price_type == 1	8
price_type == 2	8
price_type == 3	9
price_type == 4	9
price_type == 5	10
price_type == 6	10
Price_type в обновленной версии API	11
price_type == 1	11
price_type == 2	12
price_type == 3	13
price_type == 4	14
price_type == 5	15
price_type == 6	15
price_type == 7	16
price_type == 8	18
price_type == 9	18
price_type == 10	19
price_type == 11	20
Создание заказа (старый вариант).....	21
Создание заказа (обновленный вариант)	22
Тип записи individual.....	23
Тип записи group.....	23
Тип записи service	23
Тип записи tariff.....	24
Тип записи tickets	24

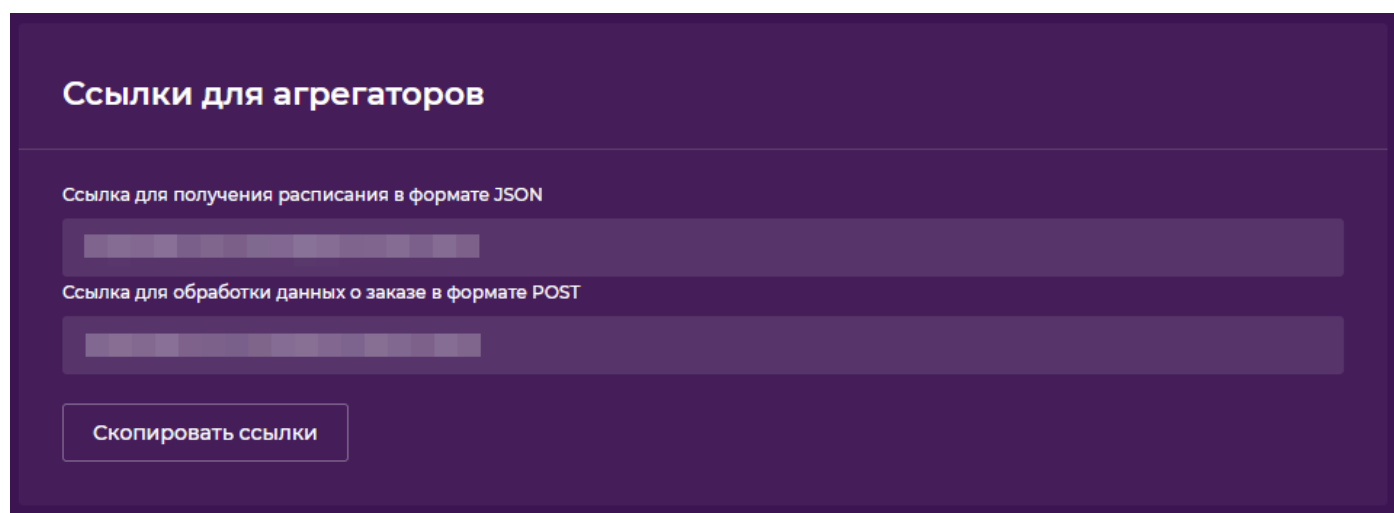
Тип записи rent	24
Оплата заказа	25
Отмена заказа	26
Обратная связь	26

Где взять ссылки для интеграции?

Ссылки для интеграции для каждой услуги можно получить в разделе «Виджеты»:



В блоке «Ссылки для агрегаторов»:



Получение расписания услуги в формате JSON

Существует 3 типа записи: Индивидуальная, Групповая и Аренда.

Для получения расписания в формате JSON надо отправить GET-запрос на адрес:

https://app.broni.biz/api/prices/{hash}

, где hash – это идентификатор услуги из личного кабинета Broni.biz (далее ЛК).

В ответ вы получите массив в формате JSON.

Пример ответа и описание полей для индивидуальной записи

```
[
  {
    "date": "2022-07-06", // дата, в формате YYYY-MM-DD
    "time": "09:00", // время, в формате HH:MM (24 часа)
    "is_free": false, // boolean, где true означает доступность для бронирования,
    // если время услуги забронировано или прошло, то возвращается false
    "idSched": 396606, // id временного слота в формате целого числа
    "questId": "nwa5es", // hash услуги
    "additional_options": [ // дополнительные услуги/опции (массив)
      {
        "id": 854699, // идентификатор доп. услуги
        "name": "Указка", // наименование доп. услуги
        "price": 200, // стоимость доп. услуги
        "amount": 50 // доступное кол-во доп. услуг
      },
      {
        "id": 280043,
        "name": "Рабочая тетрадь",
        "price": 100,
        "amount": 50
      }
    ],
    "price": 1800, // минимальная цена базовой услуги, в формате целого числа
    "add_price": 0, // добавочная стоимость на выбранное время
    "price_type": 1, // тип ценообразования в формате целого числа от 1 до 11
    "price_groups": [ // группы цен, которые зависят от типа цен price_type*
      {
        "persons": 1,
        "persons_from": 1,
        "persons_to": 8,
        "price": 1800
      }
    ]
  }
]
```

В некоторых случаях в массив добавляются дополнительные параметры

*Содержимое параметра **price_groups** меняется в зависимости от **price_type** и действительно для всех типов записи. Далее в разделе «[Price_type в обновленной версии API](#)» мы рассмотрим каждый тип цен более подробно.

Список цен, зависящий от кол-ва участников

Если в ссылке для получения расписания указать дополнительный параметр **«true»**, то система вернёт массив цен*, зависящих от кол-ва участников. На базе массива можно в форме бронирования создать выпадающий список с выбором количества человек и меняющуюся стоимостью.

Пример: <https://app.broni.biz/api/prices/{hash}/true>

```
"price_groups": [
  {
    "persons": 1,
    "price": 1800
  },
  {
    "persons": 2,
    "price": 1800
  },
  {
    "persons": 3,
    "price": 1800
  },
  {
    "persons": 4,
    "price": 1800
  }
]
```

* Распространяется только на тип «Индивидуальная запись» со значением **price_type** от 1 до 4 и 7 в обновленной версии API.

Список цен на конкретную дату и время

Если в ссылке для получения расписания указать дополнительные параметр **«date»** и **«time»**, то система вернёт массив цен на конкретную дату и время.

Пример запроса: <https://app.broni.biz/api/prices/{hash}?date=2022-07-12&time=09:00>

Пример ответа*:

```
[
  {
    "persons": 1,
    "persons_from": 1,
    "persons_to": 8,
    "price": 1800
  }
]
```

* В ответе значения могут отличаться и зависеть для каждого конкретного price_type.

Список цен на конкретную дату и время для тарифов

Если в ссылке для получения расписания указать помимо «**date**» и «**time**» дополнительный параметр «**source=tk**», то система вернёт массив с названиями тарифов и ценами на конкретную дату и время.

Пример запроса: <https://app.broni.biz/api/prices/{hash}?date=2022-07-12&time=09:00&source=tk>

Пример ответ*:

```
{
  "2 участника - 1000":1000,
  "3 участника - 2000":2000,
  "5 участников - 5000":5000
}
```

Так же параметр «**source=tk**» можно использовать без «**date**» и «**time**».

Пример запроса: <https://app.broni.biz/api/prices/{hash}?source=tk>

Пример ответа*:

```
"price_groups": {
  "2 участника - 1000":1000,
  "3 участника - 2000":2000,
  "5 участников - 2000":5000
}
```

Количество отображаемых дней в расписании

По умолчанию расписание отдается на текущую неделю + 14 дней. Если в настройках пользователя не задано меньшее количество отображаемых дней.

Если надо получить расписание по максимальному количеству дней, тогда надо предать дополнительный параметр «**&limit=fromsettings**».

Пример запроса: <https://app.broni.biz/api/prices/{hash}?&limit=fromsettings>

Price_type в старой версии API

В старой версии API для «Индивидуальной записи» было 6 типов ценообразования.

price_type == 1

В настройках ценообразования указывается оплата за группу, которая не зависит от количества участников.

В **price** будет 1500, а в **price_groups**:

```
"price_groups": [
  {
    "persons": 1,
    "price": 1500
  }
]
```

Это означает, что для группы с любым количеством участников будет фиксированная стоимость 1500 руб.

price_type == 2

В настройках ценообразования указывается оплата за 1 участника.

В **price** будет 600, а в **price_groups**:

```
"price_groups": [
  {
    "persons": 2,
    "price": 1200
  },
  {
    "persons": 10,
    "price": 6000
  }
]
```

Это означает, что стоимость для каждого участника составит 600 руб. Минимально количество участников = 2. Стоимость за двоих составит 1200 руб. (минимальная стоимость игры), максимальное количество участников = 10, стоимость для десятих составит 6000 руб.

price_type == 3

В настройках ценообразования указывается оплата за группу, минимальное и максимальное количество участников в группе.

В **price** будет 1500, а в **price_groups**:

```
"price_groups": [
  {
    "persons": 3,
    "price": 1500
  },
  {
    "persons": 8,
    "price": 1500
  }
]
```

Это означает, что минимальное количество участников = 3, а максимальное количество участников = 8. При этом стоимость услуги не зависит от кол-ва участников и составляет 1500 руб.

price_type == 4

В настройках ценообразования указывается оплата за группу, и доплата за участника сверх максимального количества участников в группе.

В **price** будет 2500, а в **price_groups**:

```
"price_groups": [
  {
    "persons": 2,
    "price": 2500
  },
  {
    "persons": 5,
    "price": 3000
  },
  {
    "persons": 5,
    "price": 4500
  }
]
```

В первом значении указано минимальное количество участников = 2 и стоимость за двоих 2500 руб. Во втором значении указан номер участника, с которого начинается доплата = 5 и стоимость за пятерых участников 3000 руб. В третьем значении указано максимальное количество участников = 8 и стоимость для восьмерых 4500 руб. Это означает, что с группы из 2-4 человек оплата составит 2500 руб., а начиная с пятого участника доплата составит 500 руб. (3000 – 2500 = 500 руб.).

price_type == 5

В настройках ценообразования указывается оплата за участника в зависимости от количества участников в группе.

В **price** будет 400, а в **price_groups**:

```
"price_groups": [
  {
    "persons": 2,
    "price": 1100
  },
  {
    "persons": 3,
    "price": 1000
  },
  {
    "persons": 4,
    "price": 900
  },
  {
    "persons": 5,
    "price": 800
  }
]
```

В **persons** указывается количество участников, а в **price** стоимость за одного участника. То есть за 3-х участников итоговая стоимость составит 3000 руб., а за 5-х участников 4000 руб.

price_type == 6

В настройках ценообразования указывается оплата за услугу.

В **price** будет 500, а в **price_groups**:

```
"price_groups": [
  {
    "name": "Маникюр",
    "price": 6000
  },
  {
    "name": "Педикюр",
    "price": 6500
  },
  {
    "name": "Маникюр + снятие",
    "price": 7000
  }
]
```

В **name** будет название услуги, а в **price** стоимость услуги.

Price_type в обновленной версии API

В обновленной версии сохранились старые типы ценообразования и добавились новые.

price_type == 1

Цена за услугу является фиксированной за сеанс и не зависит от количества мест, фактически просто бронируется время.

В **price_groups** будет:

```
"price_groups": [
  {
    "persons": 1, // для совместимости со старой версией
    "persons_from": 1, // минимальное кол-во участников в группе
    "persons_to": 5, // максимальное кол-во уч. в группе (может быть пустым)
    "price": 1800 // цена за группу
  }
]
```

Это означает, что для группы от 1 до 8 участников будет фиксированная стоимость 1800 руб.

Пример: <https://app.broni.biz/api/prices/nwa5es>

Если использовать ссылку для формирования списка участников и цен с окончанием **{hash}/true**

```
"price_groups": [
  {
    "persons": 1,
    "price": 1800
  },
  {
    "persons": 2,
    "price": 1800
  },
  {
    "persons": 3,
    "price": 1800
  },
  {
    "persons": 4,
    "price": 1800
  },
  {
    "persons": 5,
    "price": 1800
  }
]
```

Пример: <https://app.broni.biz/api/prices/nwa5es/true>

`price_type == 2`

Цена указывается за место и зависит от количества участников.

В `price_groups` будет:

```
"price_groups": [
  {
    "persons": 1, // стартовое кол-во участников в группе
    "persons_from": 1, // минимальное кол-во участников в группе для указанной цены
    "persons_to": 10, // максимальное кол-во участников в группе для указанной цены
    "price": 799 // цена за участника, если их от 1 до 10
  },
  {
    "persons": 11,
    "persons_from": 11,
    "persons_to": 20,
    "price": 699
  },
  {
    "persons": 21,
    "persons_from": 21,
    "persons_to": 30, // в последнем элементе в ЛК значение может быть пустым
    "price": 599
  }
]
```

Это означает, что для группы от 1 до 10 участников стоимость будет 799 руб. за 1 участника. А если группа будет от 21 до 30 участников, стоимость составит 599 руб. за 1 участника.

Пример: <https://app.broni.biz/api/prices/e6umtq>

Если использовать ссылку для формирования списка участников и цен с окончанием `{hash}/true`

```
"price_groups": [
  {
    "persons": 1,
    "price": 799
  },
  {
    "persons": 2,
    "price": 1598
  },
  ...
  {
    "persons": 29,
    "price": 17371
  },
  {
    "persons": 30, // если в последнем элементе в ЛК значение быть пустым, то
    "price": 17970
  }
]
```

Пример: <https://app.broni.biz/api/prices/e6umtq/true>

price_type == 3

Цена указывается за группу и зависит от количества участников.

В **price_groups** будет:

```
"price_groups": [
  {
    "persons":1, // стартовое кол-во участников в группе
    "persons_from":1, // минимальное кол-во участников в группе для указанной цены
    "persons_to":3, // максимальное кол-во участников в группе для указанной цены
    "price":2000 // цена за группу
  },
  {
    "persons":4,
    "persons_from":4,
    "persons_to":6,
    "price":3000
  }
]
```

Это означает, что для группы от 1 до 3 участников стоимость будет 2000 руб. за группу. А если группа будет от 4 до 6 участников, стоимость составит 3000 руб. за группу.

Пример: <https://app.broni.biz/api/prices/hj3zm4>

Если использовать ссылку для формирования списка участников и цен с окончанием **{hash}/true**

```
"price_groups": [
  {
    "persons":1,
    "price":2000
  },
  {
    "persons":2,
    "price":2000
  },
  {
    "persons":3,
    "price":2000
  },
  {
    "persons":4,
    "price":3000
  },
  {
    "persons":5,
    "price":3000
  },
  {
    "persons":6,
    "price":3000
  }
]
```

Пример: <https://app.broni.biz/api/prices/hj3zm4/true>

price_type == 4

Цена указывается за группу и зависит от количества участников. Можно настроить оплату за дополнительные места.

В **price_groups** будет:

```
"price_groups": [
  {
    "persons": 1, // стартовое кол-во участников в группе
    "persons_from": 1, // минимальное кол-во участников в группе для указанной цены
    "persons_to": 3, // максимальное кол-во участников в группе для указанной цены
    "price": 6050 // цена за группу
  },
  {
    "persons": 4,
    "persons_from": 4,
    "persons_to": 7,
    "price": 7050
  }
],
"group_surcharge": [ // добавляется параметр, отвечающий за дополнительных участников
  {
    "persons_from": 1, // минимальное кол-во участников в группе для указанной цены
    "persons_to": 3, // минимальное кол-во участников в группе для указанной цены
    "price": 700 // цена за дополнительного участника, если их от 1 до 3
  },
  {
    "persons_from": 4,
    "persons_to": 5,
    "price": 600
  }
]
]
```

Это означает, что для группы от 1 до 3 участников стоимость будет 6050 руб. за группу. Если группа будет от 4 до 7 участников, стоимость составит 7050 руб. за группу. Если будет 9 участников, то они попадают во вторую группу + 2 дополнительных участника по 700 руб. – итоговая стоимость составит 8450 руб.

Пример: <https://app.broni.biz/api/prices/wmtqf7>

Если использовать ссылку для формирования списка участников и цен с окончанием **{hash}/true**

```
"price_groups": [
  {
    "persons": 1,
    "price": 6050
  },
  ...
  {
    "persons": 12,
    "price": 10050
  }
]
]
```

Пример: <https://app.broni.biz/api/prices/wmtqf7/true>

`price_type == 5`

В обновленном API – это тоже самое, что `price_type == 2`.

`price_type == 6`

Цена указывается за услугу.

В `price_groups` будет:

```
"price_groups": [
  {
    "id": 206207, // идентификатор услуги
    "name": "Римская", // наименование услуги
    "duration": 60, // продолжительность услуги
    "time_unit": "minutes", // единица измерения продолжительности (еще может быть
hours или days)
    "price": 2050 // стоимость услуги
  },
  {
    "id": 398370,
    "name": "Древняя",
    "duration": 90,
    "time_unit": "minutes",
    "price": 3050
  },
  {
    "id": 347231,
    "name": "Новейшая",
    "duration": 120,
    "time_unit": "minutes",
    "price": 3550
  },
  {
    "id": 708281,
    "name": "Старейшая",
    "duration": 150,
    "time_unit": "minutes",
    "price": 4050
  }
]
```

Это означает, что услуга «Древняя» будет стоить 3050 руб., а «Старейшая» 4050 руб.

Пример: <https://app.broni.biz/api/prices/vh54jm>

`price_type == 7`

Цена зависит от тарифа и количества участников.

В `price_groups` будет:

```
"price_groups": [
  {
    "id": 221772, // идентификатор тарифа
    "name": "Стандартный", // наименование тарифа
    "duration": 60, // продолжительность услуги
    "time_unit": "minutes", // единица измерения продолжительности (еще может быть
hours или days)
    "data": [ // данные о стоимости
      {
        "from": 1, // минимальное количество участников
        "to": 5, // максимальное количество участников
        "price": 800 // стоимость за участника
      }
    ]
  },
  {
    "id": 931129,
    "name": "Расширенный",
    "duration": 90,
    "time_unit": "minutes",
    "data": [
      {
        "from": 1,
        "to": 7,
        "price": 1100
      },
      {
        "from": 7,
        "to": 10,
        "price": 2050
      }
    ]
  }
]
```

Это означает, что для 5 человек в тарифе «Стандартный» стоимость составит 4000 руб., а в тарифе «Расширенный» 5500 руб.

Пример: <https://app.broni.biz/api/prices/94sgwr>

Если использовать ссылку для формирования списка участников и цен с окончанием **{hash}/true**, то для каждого тарифа будет сформирован список с количеством участников и стоимостью.

```
"price_groups": [
  {
    "id": 221772,
    "name": "Стандартный",
    "duration": 60,
    "time_unit": "minutes",
    "data": [
      {
        "persons": 1,
        "price": 800
      },
      {
        "persons": 2,
        "price": 1600
      },
      {
        "persons": 3,
        "price": 2400
      },
      {
        "persons": 4,
        "price": 3200
      },
      {
        "persons": 5,
        "price": 4000
      }
    ]
  }
]
```

Пример: <https://app.broni.biz/api/prices/94sgwr/true>

price_type == 8

Цена указывается за билет/место и задается лимит на общее количество билетов или на каждый отдельно взятый сеанс/событие.

К основным параметрам добавляются tickets и tickets_remained, а в **price_groups** будет:

```
"tickets":15, // общее количество билетов на выбранное время
"tickets_remained":11, // оставшееся количество билетов
"price_groups":[
  {
    "id":720876, // идентификатор билета
    "name":"Взрослый", // наименование билета
    "price":1449 // цена билета
  },
  {
    "id":577829,
    "name":"Детский",
    "price":1249
  }
]
```

Это означает, что на данное время указано 15 билетов, осталось в продаже 11 шт. «Взрослый» билет стоит 1449 руб., а «Детский» 1249 руб.

Пример: <https://app.broni.biz/api/prices/h5w94e>

price_type == 9

Цена указывается за билет/место и задается лимит на количество билетов для каждого отдельного типа билета.

В **price_groups** будет:

```
"price_groups":[
  {
    "id":878222, // идентификатор билета
    "name":"Взрослый", // наименование билета
    "tickets":10, // количество билетов
    "tickets_remained":8, // оставшееся количество билетов данного типа
    "price":550 // цена билета
  },
  {
    "id":272547,
    "name":"Детский",
    "tickets":5,
    "tickets_remained":5,
    "price":450
  }
]
```

Это означает, что на данное время указано 10 билетов типа «Взрослый» по цене 550 руб., но в продаже осталось всего 8 шт., а билетов типа «Детский» осталось 5 шт., по цене 450 руб.

Пример: <https://app.broni.biz/api/prices/at825c>

`price_type == 10`

Цена указывается за время аренды.

К основным параметрам добавляется блок с информацией о продолжительности:

```
"duration":{
  "time_unit":"hours", // единица измерения времени (еще может быть minutes или days)
  "session_duration":1, // продолжительность сеанса/шаг
  "minimum_rent":2 // минимальное время аренды
}
```

, а в `price_groups` будет:

```
"price_groups":[
  {
    "from":2, // минимальное значение интервала времени
    "to":4, // максимальное значение интервала времени
    "price":1349 // стоимость за единицу времени в заданном интервале
  },
  {
    "from":5,
    "to":8,
    "price":1199
  },
  {
    "from":9,
    "to":null, // не указан максимальный интервал аренды, т.е. до бесконечности
    "price":999
  }
]
```

Это означает, что если вы арендуете от 2-х до 4-х часов, стоимость за 1 час аренды составит 1349 руб., а аренда в диапазоне от 5 до 8 часов составит 1199 руб. за 1 час.

Пример: <https://app.broni.biz/api/prices/gpe6yr>

price_type == 11

Цена указывается за время аренды, а также можно указать кол-во доступных мест и скидку за единицу времени, зависящую от кол-ва мест.

К основным параметрам добавляется блок с информацией о продолжительности:

```
"duration":{
  "time_unit":"hours", // единица измерения времени (еще может быть minutes или days)
  "session_duration":1, // продолжительность сеанса/шаг
  "minimum_rent":2 // минимальное время аренды
}
```

и блок с информацией по скидкам, зависящих от кол-ва мест:

```
"price_groups_places":[
  {
    "from":1, // минимальное значение диапазона времени
    "to":3, // минимальное значение диапазона времени
    "discount":null, // скидка
    "discount_unit":"rub" // в чем предоставляется скидка (фиксированная в валюте)
  },
  {
    "from":4,
    "to":7,
    "discount":50, // скидка
    "discount_unit":"rub" // в чем предоставляется скидка (фиксированная в валюте)
  },
  {
    "id":null,
    "from":8,
    "to":10,
    "discount":10, // скидка в %
    "discount_unit":"procent" // в чем предоставляется скидка
  }
]
```

, а в **price_groups** будет:

```
"price_groups":[
  {
    "from":2, // минимальное значение интервала времени
    "to":4, // максимальное значение интервала времени
    "price":1349 // стоимость за единицу времени в заданном интервале
  },
  {
    "from":5,
    "to":8,
    "price":1199
  }
]
```

Это означает, что при аренде в интервале от 2-х до 4-х, часов стоимость за 1 час составит 1349 руб. за 1 место. Если мест будет от 4-х до 7, то предоставляется скидка на каждый час по 50 руб.

Пример: <https://app.broni.biz/api/prices/hszj35>

Создание заказа (старый вариант)

Обратите внимание, что ниже в разделе «[Создание заказа \(новый вариант\)](#)» будет описан обновленный принцип создания заказов. Старый вариант оставлен для совместимости версий. В старом варианте для бронирования услуги необходимо отправить POST-запрос на адрес:

`https://broni.biz/api/create/{hash}`

, где hash – это идентификатор услуги.

В запросе должны присутствовать поля:

- **first_name:** имя клиента – **обязательное поле**
- **family_name:** фамилия клиента
- **phone:** телефон клиента
- **email:** e-mail клиента
- **comment:** комментарий от клиента
- **idSched:** ID слота – **обязательное поле**
- **date:** дата игры – **обязательное поле**
- **time:** время игры – **обязательное поле**
- **price:** цена игры – **обязательное поле**
- **players:** количество игроков
- **source:** название источника брони (пример: test.ru)

В случае успешной обработки данных вернётся JSON ответ:

```
{  
  "success": true  
}
```

Если заказ не получилось создать, вернётся ответ:

```
{  
  "success": false,  
  "message": "error message"  
}
```

При этом в сообщении будет указана ошибка, например: «Указанное время занято».

Создание заказа (обновленный вариант)

Для бронирования услуги необходимо отправить POST-запрос на адрес:

`https://app.broni.biz/api/create/{hash}`

, где hash – это идентификатор услуги.

В зависимости от «Типа ценообразования» мы ожидаем параметры:

```
{
// данные о временном слоте
  "resource_id": "hszj35", // хэш услуги - обязательное
  "schedule_id": 344916, // id слота - обязательное
  "date": "2022-07-07", // дата заказа - обязательное
  "time": "16:00", // время заказа - обязательное
// данные о клиенте
  "client_info": {
    "name": "Олег", // имя клиента
    "surname": "Олег", // фамилия клиента
    "tel": "+7 (000) 000-00-00", // телефон клиента - обязательное
    "email": "support@broni.biz", // e-mail клиента
    "comments": null, // комментарий клиента
    "promocode": {
      "name": "test", // наименование промокода
      "value": 10, // значение скидки
      "unit": "percent", // единица измерения (еще может быть rub)
    }
  },
// данные о заказе*
  "book_info": { // обязательный блок, для каждого типа записи
    "total_price": 3100, // итоговая цена
    "totalPriceDiscount": null, // итоговая цена со скидкой
    "individual": null, // тип записи «Индивидуальная»* (зависит от кол-ва
участников)
    "group": null, // тип записи «Индивидуальная» (зависит от группы)
    "service": null, // тип записи «Индивидуальная» (зависит от услуги и ее
продолжительности)
    "tariff": null, // тип записи «Индивидуальная» (зависит от тарифа)
    "tickets": null, // тип записи «Групповая» (билеты)
    "rent": { // тип записи «Аренда»
      "price": 1550,
      "discount": "0",
      "discount_unit": "rub", // единица измерения (еще может быть percent)
      "places": 1,
      "duration": 2,
      "unit": "hour"
    }
  },
// дополнительные опции
  "additional_options": {
    "id": 854699, // идентификатор доп. услуги
    "name": "Указка", // наименование доп. услуги
    "price": 200, // стоимость доп. услуги
    "amount": 50, // доступное кол-во доп услуг
    "selected": 1 // выбранное кол-во доп услуг
  }
}
```

* Необходимые параметры под каждый тип записи мы рассмотрим отдельно ниже.

Тип записи individual

Для типов записи price_type == 1, 2, 3 необходимо отправлять параметр individual:

```
"individual": {  
  "price": 1800, // базовая стоимость - обязательное  
  "persons": 4, // количество участников - обязательное  
  "duration": 1, // продолжительность услуги - опционально  
  "unit": "hour" // единица измерения продолжительности услуги (еще может быть  
minutes или days) - опционально  
}
```

Тип записи group

Для типов записи price_type == 4 необходимо отправлять параметр group:

```
"group": {  
  "id": 691824, // идентификатор группы - обязательное  
  "price": 7000, // стоимость за группу - обязательное  
  "persons_from": 4, // минимальное кол-во участников в группе - обязательное  
  "persons_to": 7, // максимальное кол-во участников в группе - обязательное  
  "add_persons": 4, // кол-во доп. участников - опционально  
  "price_add_persons": 800, // стоимость за доп. участника - опционально  
  "duration": 60, // продолжительность услуги - опционально  
  "unit": "minutes" // единица измерения продолжительности услуги (еще может быть  
hours или days) - опционально  
}
```

Тип записи service

Для типа записи price_type == 6 необходимо отправлять параметр service:

```
"service": {  
  "id": 347231, // идентификатор услуги - обязательное  
  "name": "Новейшая", // наименование услуги - обязательное  
  "price": 3800, // стоимость услуги - обязательное  
  "duration": "120", // продолжительность услуги - опционально  
  "unit": "minutes" // единица измерения продолжительности услуги (еще может быть  
hours или days) - опционально  
}
```

Тип записи tariff

Для типа записи price_type == 7 необходимо отправлять параметр tariff:

```
"tariff": {  
  "id": 931129, // идентификатор тарифа - обязательное  
  "tariff_name": "Расширенный", // наименование тарифа - обязательное  
  "persons": 4, // количество участников - обязательное  
  "price": 800, // стоимость за 1 участника в выбранном тарифе - обязательное  
  "duration": "90", // продолжительность услуги - опционально  
  "unit": "minutes" // единица измерения продолжительности услуги (еще может быть  
hours или days) - опционально  
}
```

Тип записи tickets

Для типов записи price_type == 8, 9 необходимо отправлять параметр tickets:

```
"tickets": [  
  {  
    "id": 878222 // идентификатор билета - обязательное  
    "name": "Взрослый", // наименование билета - обязательное  
    "price": 500, // стоимость билета - обязательное  
    "select_ticket": 3, // кол-во выбранных билетов  
  },  
  {  
    "id": 272547  
    "name": "Детский",  
    "price": 400,  
    "select_ticket": 2,  
  }  
]
```

Тип записи rent

Для типов записи price_type == 10, 11 необходимо отправлять параметр rent:

```
"rent": {  
  "price": 1500, // базовая стоимость единицы времени - обязательное  
  "duration": 5, // количество арендуемого времени - обязательное  
  "unit": "hour" // единица измерения времени аренды (еще может быть minutes или  
days) - обязательное  
  // далее поля, которые нужны только при аренде нескольких мест (price_type == 11)  
  "places": 4, // количество участников - обязательное для price_type == 11  
  "discount": 199, // скидка за 1 единицу времени  
  "discount_unit": "procent" // единица измерения скидки (еще может быть rub)  
}
```


Оплата заказа

Для того чтобы система вернула ссылку на оплату услуги онлайн при создании заказа в POST-запрос на адрес

<https://app.broni.biz/api/create/{hash}>

надо дополнительно отправить:

```
"pay_type": "online", // тип оплаты - опционально (также может быть offline)
"pay_summ": 1000, // сумма оплаты - опционально
```

Если вы указали тип оплаты **online** и сумма меньше чем итоговая стоимость заказ, то оплата онлайн будет считаться как предоплата. Если сумма не передана или больше чем стоимость заказа, то будет указана оплата из стоимости заказа.

Если заказ будет успешно сформирован, то в ответе дополнительно будет возвращен хэш заказа, ссылка на оплату, id эквайринга и хэш оплаты:

```
{
  "success": true
  "booking_hash": "S3fS3s", // хэш заказа
  "payment_link": "https://securepay.tinkoff.ru/new/6wMIJPaD", // ссылка на оплату
  "payment_acquiring_id": 2, // id эквайринга в Broni.biz
  "payment_hash": "FpS3Se" // хэш оплаты
}
```

Если клиент оплатит заказ по сформированной ссылке для оплаты онлайн, то данные зафиксируются в Broni.biz.

Так же вы можете сформировать оплату онлайн/оффлайн на своей стороне и указать Broni.biz как был оплачен заказ. Для того чтобы указать, что у заказа была оплата надо в POST-запрос на адрес

https://app.broni.biz/api/pay/api_payment

отправить данные

```
{
  "pay_type": "online", // тип оплаты - обязательное (также может быть offline)
  "pay_summ": 1000, // сумма оплаты - обязательное
  "pay_hash": "FpS3Se", // хэш оплаты - обязательное
}
```

Если оплаты онлайн не было, то у Оплаты можно изменить тип оплаты и оплатить заказ.

Отмена заказа

Для того чтобы отменить заказ надо в POST-запросе на адрес

<https://app.broni.biz/api/cancelbooking>

отправить хэш заказа:

```
{  
  "booking_hash": "S3fS3s", // хэш заказа  
}
```

Обратная связь

Если у вас остались вопросы, вы можете направить их нам на почту support@broni.biz.