

# Справочник всех функций

## Получение данных бота

Метод	Описание
<b>bot.getAllData(): array</b>	Возвращает все данные текущего бота
<b>bot.setData(string \$key, \$value): self</b>	Устанавливает значение данных текущего бота по ключу
<b>bot.getData(string \$key): mixed null</b>	Возвращает данные текущего бота по ключу

## Работа с лидами

### Методы bot

Метод	Описание
<b>bot.findLeads(array object \$columns = [], array object \$where = [], array object \$orderBy = [], int null \$limit = null, int null \$offset = null): array</b>	Находит лиды по указанным атрибутам
<b>bot.getGeoLocation(): array null</b>	Возвращает геолокацию лида

### Методы lead

Метод	Описание
-------	----------

<p><b>lead.getAllData(): array</b></p>	<p>Возвращает все данные лида</p> <ul style="list-style-type: none"><li>• <b>id</b> - id лида;</li><li>• <b>name</b> - имя лида;</li><li>• <b>identification</b> - внешний id лида;</li><li>• <b>identification_entity</b> - внешний id сущности;</li><li>• <b>person_id</b> - id персоны лида;</li><li>• <b>manager_id</b> - id менеджера привязанного к лиду;</li><li>• <b>bot_id</b> - id бота лида;</li><li>• <b>channel_id</b> - id канала лида;</li><li>• <b>lang_code</b> - код языка;</li><li>• <b>status_id</b> - id статуса лида;</li><li>• <b>is_mute</b> - включен ли бот для лида;</li><li>• все системные поля лида (по английски).</li></ul>
<p><b>lead.setData(string \$key, \$value): self</b></p>	<p>Устанавливает значение данных лида по ключу Возможные ключи:</p> <ul style="list-style-type: none"><li>• <b>id</b> - id лида;</li><li>• <b>name</b> - имя лида;</li><li>• <b>identification</b> - внешний id лида;</li><li>• <b>identification_entity</b> - внешний id сущности;</li><li>• <b>person_id</b> - id персоны лида;</li><li>• <b>manager_id</b> - id менеджера привязанного к лиду;</li><li>• <b>bot_id</b> - id бота лида;</li><li>• <b>channel_id</b> - id канала лида;</li><li>• <b>lang_code</b> - код языка;</li><li>• <b>status_id</b> - id статуса лида;</li><li>• <b>is_mute</b> - включен ли бот для лида;</li><li>• все системные поля лида (по английски).</li></ul>
<p><b>lead.getData(string \$key): mixed null</b></p>	<p>Возвращает значение данных лида по ключу Возможные ключи:</p> <ul style="list-style-type: none"><li>• <b>id</b> - id лида;</li><li>• <b>name</b> - имя лида;</li><li>• <b>identification</b> - внешний id лида;</li><li>• <b>identification_entity</b> - внешний id сущности;</li><li>• <b>person_id</b> - id персоны лида;</li><li>• <b>manager_id</b> - id менеджера привязанного к лиду;</li><li>• <b>bot_id</b> - id бота лида;</li><li>• <b>channel_id</b> - id канала лида;</li><li>• <b>lang_code</b> - код языка;</li><li>• <b>status_id</b> - id статуса лида;</li><li>• <b>is_mute</b> - включен ли бот для лида;</li><li>• все системные поля лида (по английски).</li></ul>

<b>lead.issetData(string \$key): bool</b>	<p>Проверяет наличие данных лида по ключу</p> <p>Возможные ключи:</p> <ul style="list-style-type: none"> <li>• <b>id</b> - id лида;</li> <li>• <b>name</b> - имя лида;</li> <li>• <b>identification</b> - внешний id лида;</li> <li>• <b>identification_entity</b> - внешний id сущности;</li> <li>• <b>person_id</b> - id персоны лида;</li> <li>• <b>manager_id</b> - id менеджера привязанного к лиду;</li> <li>• <b>bot_id</b> - id бота лида;</li> <li>• <b>channel_id</b> - id канала лида;</li> <li>• <b>lang_code</b> - код языка;</li> <li>• <b>status_id</b> - id статуса лида;</li> <li>• <b>is_mute</b> - включен ли бот для лида;</li> <li>• все системные поля лида (по английски).</li> </ul>
<b>lead.isDataExist(string \$key): bool</b>	
<b>lead.getSerialNumber(): int null</b>	Возвращает порядковый номер лида в текущем боте
<b>lead.getChannelCode(): string</b>	Возвращает код канала лида
<b>lead.isInlineButtonsAvailable(): bool null</b>	Проверяет включены ли инлайн кнопки
<b>lead.isInlineButtonsAvailableGeneralChannelSetting(): bool null</b>	Возвращает настройки кнопок канала

## Работа со скриптами

Метод	Описание
<b>bot.scheduleScript</b> (int null \$scriptId, int null \$leadId, int null \$ticketId = null, \$jobParams = []): <b>bool</b>	Планирует вызов скрипта
<b>bot.scheduleScriptByCode</b> (string \$scriptCode, int null \$leadId, int null \$ticketId = null, \$jobParams = []): <b>bool</b>	Планирует вызов скрипта по коду
<b>bot.getScriptIdByName</b> (string null \$scriptName): <b>int null</b>	Возвращает ID скрипта по наименованию
<b>bot.getScriptIdByCode</b> (string null \$scriptCode): <b>int null</b>	Возвращает ID скрипта по коду
<b>bot.getScriptCodeById</b> (int null \$scriptId): <b>int null</b>	Возвращает код скрипта по ID

<b>bot.runScriptForLead</b> (?int \$scriptId, ?int \$leadId, ?int \$ticketId = null, \$addToQueue = true, \$jobParams = []): <b>bool</b>	Запускает скрипт для лида по ID скрипта
<b>bot.runScriptByNameForLead</b> (string null \$scriptName, int null \$leadId, int null \$ticketId = null, \$addToQueue = true, \$jobParams = []): <b>bool</b>	Запускает скрипт для лида по наименованию скрипта
<b>bot.runScriptForPerson</b> (int null \$scriptId, int null \$personId, int null \$ticketId = null, \$addToQueue = true, \$jobParams = []): <b>bool</b>	Запускает скрипт для персоны по ID скрипта
<b>bot.runScriptByNameForPerson</b> (string null \$scriptName, int null \$personId, int null \$ticketId = null, \$addToQueue = true, \$jobParams = []): <b>bool</b>	Запускает скрипт для персоны по наименованию скрипта
<b>bot.runScriptByCodeForLead</b> (string null \$scriptCode, int null \$leadId, int null \$ticketId = null, \$addToQueue = true, \$jobParams = []): <b>bool</b>	Запускает скрипт для лида по коду
<b>bot.runScriptByCodeForPerson</b> (string null \$scriptCode, int null \$personId, int null \$ticketId = null, \$addToQueue = true, \$jobParams = []): <b>bool</b>	Запускает скрипт для персоны по коду
<b>bot.clearJobsByScriptId</b> (int null \$scriptId, int null \$leadId = null): <b>bool</b>	Очищает джобы по id скрипта
<b>bot.clearJobsByScriptName</b> (string null \$scriptName, int null \$leadId = null): <b>bool</b>	Очищает джобы по наименованию скрипта
<b>bot.clearJobsByScriptCode</b> (string null \$scriptCode, int null \$leadId = null): <b>bool</b>	Очищает джобы по коду скрипта

## Работа с атрибутами

### Методы bot

Метод	Описание
-------	----------

<code>bot.setAttr(string \$key, string \$value): self</code>	Устанавливает значение переменной бота
<code>bot.getAttr(string \$key): mixed null</code>	Возвращает значение переменной бота
<code>bot.getAllAttr(): array</code>	Возвращает массив всех переменных бота
<code>bot.getIntAttr(string \$key, int null \$default = 0): int null</code>	Возвращает значение int переменной бота
<code>bot.getFloatAttr(string \$key, float null \$default = 0.0): float null</code>	Возвращает значение float переменной бота
<code>bot.getBoolAttr(string \$key, bool null \$default = false): bool null</code>	Возвращает значение bool переменной бота
<code>bot.issetAttr(string \$key): bool</code>	Проверяет наличие переменной бота
<code>bot.isAttrExist(string \$key): bool</code>	
<code>bot.getAllJsonAttrs(): array</code>	Возвращает значение всех JSON переменных бота
<code>bot.setJsonAttr(string \$key, \$value): self</code>	Устанавливает значение JSON переменной бота
<code>bot.getJsonAttr(string \$key): mixed null</code>	Возвращает значение JSON переменной бота
<code>bot.issetJsonAttr(string \$key): bool</code>	Проверяет наличие JSON переменной бота
<code>bot.isJsonAttrKeyExist(string \$key): bool</code>	

## Методы lead

Метод	Описание
<code>lead.setAttr(string \$key, string \$value): self</code>	Устанавливает значение переменной лида
<code>lead.getAttr(string \$key): string null</code>	Возвращает значение переменной лида
<code>lead.getAllAttr(): array</code>	Возвращает все переменные лида

<b>lead.getIntAttr</b> (string \$key, ?int null \$default = 0): <b>int</b> <b> null</b>	Возвращает значение int переменной лида
<b>lead.getFloatAttr</b> (string \$key, ?float null \$default = 0.0): <b>float</b> <b> null</b>	Возвращает значение float переменной лида
<b>lead.getBoolAttr</b> (string \$key, ?bool null \$default = false): <b>bool</b> <b> null</b>	Возвращает значение bool переменной лида
<b>lead.issetAttr</b> (string \$key): <b>bool</b>	Проверяет наличие переменной лида
<b>lead.isAttrExist</b> (string \$key): <b>bool</b>	
<b>lead.deleteAttrs</b> (string object array null \$attrs, int null \$ticketId = null, \$scriptRequestParams = [], bool \$isForceDelete = false, bool \$isSuppressTriggers = false): <b>self</b>	Удаляет переменную/переменные лида
<b>lead.deleteAttr</b> (string object array null \$attrs, int null \$ticketId = null, \$scriptRequestParams = [], bool \$isForceDelete = false, bool \$isSuppressTriggers = false): <b>self</b>	
<b>lead.forceDeleteAttrs</b> (string object array null \$attrs, int null \$ticketId = null, \$scriptRequestParams = [], bool \$isSuppressTriggers = false): <b>self</b>	Принудительно удаляет переменную/переменные лида из БД
<b>lead.forceDeleteAttr</b> (string object array null \$attrs, int null \$ticketId = null, \$scriptRequestParams = [], bool \$isSuppressTriggers = false): <b>self</b>	
<b>lead.getAllJsonAttrs</b> (): <b>array</b>	Возвращает значение всех JSON переменных лида
<b>lead.setJsonAttr</b> (string \$key, \$value): <b>self</b>	Устанавливает значение JSON переменной лида
<b>lead.getJsonAttr</b> (string \$key): <b>mixed</b> <b> null</b>	Возвращает значение JSON переменной лида
<b>lead.issetJsonAttr</b> (string \$key): <b>bool</b>	Проверяет наличие JSON переменной лида
<b>lead.isJsonAttrKeyExist</b> (string \$key): <b>bool</b>	

<b>lead.deleteJsonAttrs(string object array null \$attrs, int null \$ticketId = null, \$scriptRequestParams = [], bool \$isForceDelete = false, bool \$isSuppressTriggers = false): self</b>	Удаляет json-переменную/json-переменные лида
<b>lead.deleteJsonAttr(string object array null \$attrs, int null \$ticketId = null, \$scriptRequestParams = [], bool \$isForceDelete = false, bool \$isSuppressTriggers = false): self</b>	
<b>lead.forceDeleteJsonAttrs(string object array null \$attrs, int null \$ticketId = null, \$scriptRequestParams = [], bool \$isSuppressTriggers = false): self</b>	Принудительно удаляет json-переменную/json-переменные лида из БД
<b>lead.forceDeleteJsonAttr(string object array null \$attrs, int null \$ticketId = null, \$scriptRequestParams = [], bool \$isSuppressTriggers = false): self</b>	
<b>lead.setAllLiveTexAttrs(string \$value): self</b>	Устанавливает значение всех LiveTex переменных лида
<b>lead.setLiveTexAttr(string \$key, string \$value): self</b>	Устанавливает значение LiveTex переменной лида
<b>lead.getAllLiveTexAttrs(): array</b>	Возвращает значение всех LiveTex переменных лида
<b>lead.getLiveTexAttr(string \$key): mixed null</b>	Возвращает значение LiveTex переменной лида

## Методы memory

Метод	Описание
<b>memory.setAttr(string \$key, string \$value): self</b>	Устанавливает значение временной переменной
<b>memory.getAttr(string \$key): string null</b>	Возвращает значение временной переменной
<b>memory.getIntAttr(string \$key, ?int null \$default = 0): int null</b>	Возвращает значение int временной переменной
<b>memory.getFloatAttr(string \$key, ?float null \$default = 0.0): float null</b>	Возвращает значение float временной переменной

<b>memory.getBoolAttr</b> (string \$key, ?bool null \$default = false): <b>bool null</b>	Возвращает значение bool временной переменной
<b>memory.issetAttr</b> (string \$key): <b>bool</b>	Проверяет наличие временной переменной
<b>memory.isAttrExist</b> (string \$key): <b>bool</b>	
<b>memory.setAllJsonAttrs</b> (array \$data): <b>self</b>	Устанавливает значение всех временных JSON переменных
<b>memory.getAllJsonAttrs</b> (): <b>array</b>	Возвращает значение всех временных JSON переменных
<b>memory.setJsonAttr</b> (string \$key, \$value): <b>self</b>	Устанавливает значение временной JSON переменной
<b>memory.getJsonAttr</b> (string \$key): <b>mixed null</b>	Возвращает значение временной JSON переменной
<b>memory.issetJsonAttr</b> (string \$key): <b>bool</b>	Проверяет наличие временной JSON переменной
<b>memory.isJsonAttrKeyExist</b> (string \$key): <b>bool</b>	

## Работа с триггерами

Метод	Описание
<b>bot.scheduleTrigger</b> (int null \$triggerId, int null \$leadId, int null \$ticketId = null, \$jobParams = []): <b>bool</b>	Планирует запуск триггера
<b>bot.scheduleTriggerByCode</b> (string \$triggerCode, int null \$leadId, int null \$ticketId = null, \$jobParams = []): <b>bool</b>	Планирует запуск триггера по коду
<b>bot.runTriggerForLead</b> (?int \$triggerId, ?int \$leadId, ?int \$ticketId = null, \$addToQueue = true, \$jobParams = []): <b>bool</b>	Запускает триггер для лида по ID триггера
<b>bot.runTriggerByCodeForLead</b> (string null \$triggerCode, int null \$leadId, int null \$ticketId = null, \$addToQueue = true, \$jobParams = []): <b>bool</b>	Планирует запуск триггера по коду
<b>bot.runTriggerByCodeForPerson</b> (string null \$triggerCode, int null \$personId, int null \$ticketId = null, \$addToQueue = true, \$jobParams = []): <b>bool</b>	Планирует запуск триггера по коду для последнего лида персоны

<code>bot.clearTriggerJobsForLead(int null \$leadId, int null \$ticketId = null): bool</code>	Удаляет триггеры для лида
<code>bot.clearTriggerJobsForAllPersonLeads(int null \$personId, int null \$ticketId = null): bool</code>	Удаляет триггеры для всех лидов персоны
<code>bot.getTriggerIdByName(string null \$triggerName): int null</code>	Возвращает ID триггера по наименованию
<code>bot.getTriggerIdByCode(string null \$triggerCode): int null</code>	Возвращает ID Триггера по Коду
<code>bot.runTriggerByNameForLead(string null \$triggerName, int null \$leadId, int null \$ticketId = null, \$addToQueue = true, \$jobParams = []): bool</code>	Запускает триггер для лида по наименованию
<code>bot.runTriggerForPerson(int null \$triggerId, int null \$personId, int null \$ticketId = null, \$addToQueue = true, \$jobParams = []): bool</code>	Запускает триггер для персоны по ID
<code>bot.runTriggerByNameForPerson(string null \$triggerName, int null \$personId, int null \$ticketId = null, \$addToQueue = true, \$jobParams = []): bool</code>	Запускает триггер для персоны по наименованию
<code>bot.clearTriggerJobsByScriptId(int null \$scriptId, int null \$leadId = null): bool</code>	Очищает джобы триггера по id скрипта
<code>bot.clearTriggerJobsByScriptName(string null \$scriptName, int null \$leadId = null): bool</code>	Очищает джобы триггера по наименованию скрипта
<code>bot.clearTriggerJobsById(int null \$triggerId, int null \$leadId = null): bool</code>	Очищает джобы по id триггера
<code>bot.clearTriggerJobsByCode(string null \$triggerCode, int null \$leadId = null, int \$ticketId = null): bool</code>	Очищает джобы по коду триггера

<b>bot.clearTriggerJobsByCodeForAllPersonLeads</b> (string null \$triggerCode, int null \$personId, int \$ticketId = null): bool	Удаляет триггеры для всех лидов персоны по коду
<b>bot.clearTriggerJobsByName(string null \$triggerName, int null \$leadId = null): bool</b>	Очищает джобы по наименованию триггера
<b>bot.clearTriggerJobsByScriptCode(string null \$scriptCode, int null \$leadId = null, int null \$ticketId = null): bool</b>	Очищает джобы по коду скрипта
<b>bot.clearTriggerJobsByIdForTicket(int null \$triggerId, int null \$ticketId): bool</b>	Очищает джобы по id триггера и ID заявки
<b>bot.clearTriggerJobsByNameForTicket(string null \$triggerName, int null \$ticketId): bool</b>	Очищает джобы по наименованию триггера и ID заявки
<b>bot.clearTriggerJobsByCodeForTicket(string null \$triggerCode, int null \$ticketId): bool</b>	Очищает джобы по коду триггера и ID заявки

## Работа со статусами

Метод	Описание
<b>lead.isInStatus(string \$statusName): bool</b>	Проверяет существует ли статуса у лида
<b>lead.getStatusId(): int null</b>	Возвращает id статуса лида
<b>lead.getAllStatuses(): array</b>	Возвращает все статусы лида
<b>lead.getAllStatusesIndexedByName(): array</b>	Возвращает все наименования статусов лида
<b>lead.setStatusId(int \$statusId, string \$name): self</b>	Устанавливает id статуса лида
<b>lead.setStatusName(string \$name): self</b>	Устанавливает наименование статуса лида
<b>lead.getStatusName(): string</b>	Возвращает наименование статуса лида

<b>lead.setStatus(int \$statusId, string \$name): self</b>	Устанавливает статус лида
<b>lead.setStatusById(int \$statusId): self</b>	Устанавливает статус лида по id
<b>lead.issetStatusById(string \$id): bool</b>	Проверяет существует ли статус у лида по id
<b>lead.issetStatusByName(string \$name): bool</b>	Проверяет существует ли статуса у лида по наименованию
<b>lead.getStatusIdByName(string \$statusName): int</b>	Возвращает id статуса лида по наименованию
<b>lead.setStatusByName(string \$statusName): self</b>	Устанавливает статус лида по наименованию

## Работа с рассылками

Метод	Описание
<b>bot.scheduleBroadcast(int null \$broadcastId, int null \$ticketId = null, \$jobParams = []): bool</b>	Планирует запуск рассылки
<b>bot.scheduleBroadcastByCode(string null \$broadcastCode, int null \$ticketId = null, \$jobParams = []): bool</b>	Планирует запуск рассылки по коду
<b>bot.getBroadcastIdByCode(string null \$broadcastCode): int null</b>	Возвращает ID рассылки по коду

## Работа с тэгами

Метод	Описание
<b>lead.isTagExist('some_tag'): bool</b>	Проверяет наличие тэга лида
<b>lead.issetTag('some_tag'): bool</b>	
<b>lead.setTag(string \$key, \$value): self</b>	Устанавливает значение тэга лида
<b>lead.getTag(string \$key): string null</b>	Возвращает значение тэга лида
<b>lead.getAllTags(): array</b>	Возвращает массив всех тэгов лида

<b>lead.addTags(string object array null \$tags, int \$ticketId = null, \$scriptRequestParams = [], bool \$isSuppressTriggers = false): self</b>	Добавляет тэги лиду
<b>lead.addTag(string object array null \$tags, int null \$ticketId = null, \$scriptRequestParams = [], bool \$isSuppressTriggers = false): self</b>	
<b>lead.deleteTags(string object array null \$tags, int null \$ticketId = null, \$scriptRequestParams = [], bool \$isForceDelete = false, bool \$isSuppressTriggers = false): self</b>	Удаляет тэги лида
<b>lead.deleteTag(string object array null \$tags, int null \$ticketId = null, \$scriptRequestParams = [], bool \$isForceDelete = false, bool \$isSuppressTriggers = false): self</b>	
<b>lead.forceDeleteTags(string object array null \$tags, int null \$ticketId = null, \$scriptRequestParams = [], bool \$isSuppressTriggers = false): self</b>	Принудительно удаляет тэги лида из БД
<b>lead.forceDeleteTag(string object array null \$tags, int null \$ticketId = null, \$scriptRequestParams = [], bool \$isSuppressTriggers = false): self</b>	

## Работа с контекстами

Метод	Описание
<b>lead.setContext(string \$key, \$value): self</b>	Устанавливает значение контекста лида
<b>lead.isContextExist('some_tag'): bool</b>	Проверяет наличие контекста лида
<b>lead.issetContext('some_tag'): bool</b>	
<b>lead.getContext(string \$key): string null</b>	Возвращает значение контекста лида
<b>lead.getAllContexts(): array</b>	Возвращает массив всех контекстов лида

# Работа с переключением на оператора

## Методы bot

Метод	Описание
<code>bot.getForwardEventType(): bool</code>	Проверяет кто переключил на оператора
<code>bot.isForwardedToOperator(): bool</code>	Проверяет переключения на оператора

## Методы lead

Метод	Описание
<code>lead.setForwarded(bool int \$state): self</code>	Переключает бот на оператора

# Работа с заявками

## Методы bot

Метод	Описание
<code>bot.createTicket(\$data): int null</code>	Создает заявку
<code>bot.updateTicketById(int \$ticketId, \$data): bool</code>	Обновляет заявку
<code>bot.getMaxRejectsTicketCount(): int null</code>	Возвращает максимальное количество отказов
<code>bot.findTickets(\$where = [], bool \$findInAllBusinessBots = false, \$orderBy = [], int null \$limit = null, int null \$offset = null): array</code>	Находит заявку по заданным атрибутам
<code>bot.findTicketIdByCustomField(string \$fieldName, \$value): int null</code>	Возвращает id заявки по значению поля
<code>bot.getTicketTypeIdByName(string \$typeName): int null</code>	Возвращает id заявки по наименованию вида заявки
<code>bot.getTicketTypeNameById(int null \$ticketTypeId): string null</code>	Возвращает наименование заявки по id вида заявки

<b>bot.getTicketStatusIdByName(string \$statusName): int null</b>	Возвращает id заявки по наименованию статуса заявки
<b>bot.setCurrentTicketById(int null \$ticketId): bool</b>	Устанавливает указанную заявку как текущую
<b>bot.clearTicketAssignedPersonsLog(): bool</b>	Очищает лог назначения тикета персонам

## Методы ticket

Метод	Описание
<b>ticket.update(\$data): bool</b>	Обновляет заявку, через объект заявки
<b>ticket.getAllData(): array</b>	Возвращает значение всех данных заявки
<b>ticket.setData(string \$key, \$value): self</b>	<p>Устанавливает значение данных заявки по ключу В качестве ключа можно указать:</p> <ul style="list-style-type: none"> <li>• <b>id</b> - id заявки;</li> <li>• <b>business_id</b> - id бизнеса;</li> <li>• <b>bot_id</b> - id бота;</li> <li>• <b>author_person_id</b> - id автора заявки;</li> <li>• <b>client_person_id</b> - id клиента заявки;</li> <li>• <b>assigned_person_id</b> - id исполнителя заявки;</li> <li>• <b>ticket_type_id</b> - id вида заявки, виды заявки создаются вручную;</li> <li>• <b>payment_type_id</b> - id способа оплаты. Доступные значения: <ul style="list-style-type: none"> <li>◦ <b>0</b> - PAYMENT_TYPE_NONE;</li> <li>◦ <b>1</b> - PAYMENT_TYPE_ROBOKASSA;</li> <li>◦ <b>2</b> - PAYMENT_TYPE_YANDEXKASSA;</li> <li>◦ <b>3</b> - PAYMENT_TYPE_CASH;</li> </ul> </li> <li>• <b>status_id</b> - id статуса заявки;</li> <li>• <b>title</b> - заголовок заявки;</li> <li>• <b>description</b> - описание заявки;</li> <li>• <b>created_at</b> - время создания заявки;</li> <li>• <b>updated_at</b> - время обновления заявки;</li> <li>• <b>deleted_at</b> - время удаления заявки;</li> <li>• все системные поля заявки (по английски).</li> </ul>

**ticket.getData(string \$key): mixed|null**

Возвращает значение данных заявки по ключу  
В качестве ключа можно указать:

- **id** - id заявки;
- **business\_id** - id бизнеса;
- **bot\_id** - id бота;
- **author\_person\_id** - id автора заявки;
- **client\_person\_id** - id клиента заявки;
- **assigned\_person\_id** - id исполнителя заявки;
- **ticket\_type\_id** - id вида заявки, виды заявки создаются вручную;
- **payment\_type\_id** - id способа оплаты.  
Доступные значения:
  - **0** - PAYMENT\_TYPE\_NONE;
  - **1** - PAYMENT\_TYPE\_ROBOKASSA;
  - **2** - PAYMENT\_TYPE\_YANDEXKASSA;
  - **3** - PAYMENT\_TYPE\_CASH;
- **status\_id** - id статуса заявки;
- **title** - заголовок заявки;
- **description** - описание заявки;
- **created\_at** - время создания заявки;
- **updated\_at** - время обновления заявки;
- **deleted\_at** - время удаления заявки;
- все системные поля заявки (по английски).

**ticket.getCustomData(string \$key): mixed|null**

Считывает произвольные поля заявки по ключу  
В качестве ключа можно указать:

- **названия всех произвольных полей заявки;**
- **id** - id заявки;
- **business\_id** - id бизнеса;
- **bot\_id** - id бота;
- **author\_person\_id** - id автора заявки;
- **client\_person\_id** - id клиента заявки;
- **assigned\_person\_id** - id исполнителя заявки;
- **ticket\_type\_id** - id вида заявки, виды заявки создаются вручную;
- **payment\_type\_id** - id способа оплаты.  
Доступные значения:
  - **0** - PAYMENT\_TYPE\_NONE;
  - **1** - PAYMENT\_TYPE\_ROBOKASSA;
  - **2** - PAYMENT\_TYPE\_YANDEXKASSA;
  - **3** - PAYMENT\_TYPE\_CASH;
- **status\_id** - id статуса заявки;
- **title** - заголовок заявки;
- **description** - описание заявки;
- **created\_at** - время создания заявки;
- **updated\_at** - время обновления заявки;
- **deleted\_at** - время удаления заявки;
- все системные поля заявки (по английски).

<p><b>ticket.issetData(string \$key): bool</b></p>	<p>Проверяет наличие данных по заявке по ключу В качестве ключа можно указать:</p> <ul style="list-style-type: none"> <li>• <b>id</b> - id заявки;</li> <li>• <b>business_id</b> - id бизнеса;</li> <li>• <b>bot_id</b> - id бота;</li> <li>• <b>author_person_id</b> - id автора заявки;</li> <li>• <b>client_person_id</b> - id клиента заявки;</li> <li>• <b>assigned_person_id</b> - id исполнителя заявки;</li> <li>• <b>ticket_type_id</b> - id вида заявки, виды заявки создаются вручную;</li> <li>• <b>payment_type_id</b> - id способа оплаты.</li> </ul>
<p><b>ticket.isDataExist(string \$key): bool</b></p>	<p>Доступные значения:</p> <ul style="list-style-type: none"> <li>◦ <b>0</b> - PAYMENT_TYPE_NONE;</li> <li>◦ <b>1</b> - PAYMENT_TYPE_ROBOKASSA;</li> <li>◦ <b>2</b> - PAYMENT_TYPE_YANDEXKASSA;</li> <li>◦ <b>3</b> - PAYMENT_TYPE_CASH;</li> </ul> <ul style="list-style-type: none"> <li>• <b>status_id</b> - id статуса заявки;</li> <li>• <b>title</b> - заголовок заявки;</li> <li>• <b>description</b> - описание заявки;</li> <li>• <b>created_at</b> - время создания заявки;</li> <li>• <b>updated_at</b> - время обновления заявки;</li> <li>• <b>deleted_at</b> - время удаления заявки;</li> <li>• все системные поля заявки (по английски).</li> </ul>
<p><b>ticket.reloadCustomData(bool \$isReloadModel = false): bool</b></p>	<p>Перезагружает кастомные данные</p>
<p><b>ticket.loadCustomDataIfNotLoaded(bool \$isReloadModel = false): bool</b></p>	<p>Загружает кастомные данные, если они не загружены</p>
<p><b>ticket.setAllCustomData(array \$customData): self</b></p>	<p>Переопределяет список кастомных полей</p>
<p><b>ticket.getAllCustomData(): array</b></p>	<p>Возвращает список кастомных полей</p>

**ticket.setCustomData(string \$key, \$value): self**

Устанавливает значение кастомного поля  
В качестве ключа можно указать:

- **названия всех произвольных полей заявки;**
- **id** - id заявки;
- **business\_id** - id бизнеса;
- **bot\_id** - id бота;
- **author\_person\_id** - id автора заявки;
- **client\_person\_id** - id клиента заявки;
- **assigned\_person\_id** - id исполнителя заявки;
- **ticket\_type\_id** - id вида заявки, виды заявки создаются вручную;
- **payment\_type\_id** - id способа оплаты.  
Доступные значения:
  - **0** - PAYMENT\_TYPE\_NONE;
  - **1** - PAYMENT\_TYPE\_ROBOKASSA;
  - **2** - PAYMENT\_TYPE\_YANDEXKASSA;
  - **3** - PAYMENT\_TYPE\_CASH;
- **status\_id** - id статуса заявки;
- **title** - заголовок заявки;
- **description** - описание заявки;
- **created\_at** - время создания заявки;
- **updated\_at** - время обновления заявки;
- **deleted\_at** - время удаления заявки;
- все системные поля заявки (по английски).

**ticket.issetCustomData(string \$key): bool**

Проверяет имеет ли значение кастомное поле  
В качестве ключа можно указать:

- **названия всех произвольных полей заявки;**
- **id** - id заявки;
- **business\_id** - id бизнеса;
- **bot\_id** - id бота;
- **author\_person\_id** - id автора заявки;
- **client\_person\_id** - id клиента заявки;
- **assigned\_person\_id** - id исполнителя заявки;
- **ticket\_type\_id** - id вида заявки, виды заявки создаются вручную;
- **payment\_type\_id** - id способа оплаты.  
Доступные значения:
  - **0** - PAYMENT\_TYPE\_NONE;
  - **1** - PAYMENT\_TYPE\_ROBOKASSA;
  - **2** - PAYMENT\_TYPE\_YANDEXKASSA;
  - **3** - PAYMENT\_TYPE\_CASH;
- **status\_id** - id статуса заявки;
- **title** - заголовок заявки;
- **description** - описание заявки;
- **created\_at** - время создания заявки;
- **updated\_at** - время обновления заявки;
- **deleted\_at** - время удаления заявки;
- все системные поля заявки (по английски).

<p><b>ticket.isCustomDataExist(string \$key): bool</b></p>	<p>Проверяет существует ли кастомное поле В качестве ключа можно указать:</p> <ul style="list-style-type: none"> <li>• <b>названия всех произвольных полей заявки;</b></li> <li>• <b>id</b> - id заявки;</li> <li>• <b>business_id</b> - id бизнеса;</li> <li>• <b>bot_id</b> - id бота;</li> <li>• <b>author_person_id</b> - id автора заявки;</li> <li>• <b>client_person_id</b> - id клиента заявки;</li> <li>• <b>assigned_person_id</b> - id исполнителя заявки;</li> <li>• <b>ticket_type_id</b> - id вида заявки, виды заявки создаются вручную;</li> <li>• <b>payment_type_id</b> - id способа оплаты. Доступные значения: <ul style="list-style-type: none"> <li>◦ <b>0</b> - PAYMENT_TYPE_NONE;</li> <li>◦ <b>1</b> - PAYMENT_TYPE_ROBOKASSA;</li> <li>◦ <b>2</b> - PAYMENT_TYPE_YANDEXKASSA;</li> <li>◦ <b>3</b> - PAYMENT_TYPE_CASH;</li> </ul> </li> <li>• <b>status_id</b> - id статуса заявки;</li> <li>• <b>title</b> - заголовок заявки;</li> <li>• <b>description</b> - описание заявки;</li> <li>• <b>created_at</b> - время создания заявки;</li> <li>• <b>updated_at</b> - время обновления заявки;</li> <li>• <b>deleted_at</b> - время удаления заявки;</li> <li>• все системные поля заявки (по английски).</li> </ul>
<p><b>ticket.getTicketTypeName(): string null</b></p>	<p>Возвращает название типа заявки</p>
<p><b>ticket.getStatusId(): int null</b></p>	<p>Возвращает ID статуса заявки</p>
<p><b>ticket.getAttr('status_id'): int null</b></p>	
<p><b>ticket.getStatusName(): string null</b></p>	<p>Возвращает название статуса заявки</p>
<p><b>ticket.getStatusType(): string null</b></p>	<p>Возвращает тип статуса заявки</p>
<p><b>ticket.isInStatus(\$statusName): bool</b></p>	<p>Проверяет статус заявки</p>
<p><b>ticket.setStatusId(int null \$statusId, int null \$personQueueIdForLog = null, string null \$personQueueTableForLog = null): bool</b></p>	<p>Устанавливает статус заявки по id</p>
<p><b>ticket.setStatusByName(string \$statusName, int null \$personQueueIdForLog = null, string null \$personQueueTableForLog = null): bool</b></p>	<p>Устанавливает статус заявки по имени</p>

<b>ticket.reloadTicket(): bool</b>	Загружает заявку повторно
<b>ticket.getAuthorPersonId(): int null</b>	Возвращает ID автора заявки
<b>ticket.getAssignedPersonId(): int null</b>	Возвращает ID исполнителя заявки
<b>ticket.getClientPersonId(): int null</b>	Возвращает ID клиента заявки
<b>ticket.getAuthorPersonFullName(): string</b>	Возвращает ФИО автора заявки
<b>ticket.getAssignedPersonFullName(): string</b>	Возвращает ФИО исполнителя заявки
<b>ticket.getClientPersonFullName(): string</b>	Возвращает ФИО клиента заявки
<b>ticket.getAuthorPersonRoleId(): int null</b>	Возвращает ID роли автора заявки
<b>ticket.getAssignedPersonRoleId(): int null</b>	Возвращает ID роли исполнителя заявки
<b>ticket.getClientPersonRoleId(): int null</b>	Возвращает ID роли клиента заявки
<b>ticket.getAuthorPersonRoleName(): string</b>	Возвращает имя роли автора заявки
<b>ticket.getAssignedPersonRoleName(): string</b>	Возвращает имя роли исполнителя заявки
<b>ticket.getClientPersonRoleName(): string</b>	Возвращает имя роли клиента заявки
<b>ticket.getAuthorPersonRoleTitle(): string</b>	Возвращает заголовок роли автора заявки
<b>ticket.getAssignedPersonRoleTitle(): string</b>	Возвращает заголовок роли исполнителя заявки
<b>ticket.getClientPersonRoleTitle(): string</b>	Возвращает заголовок роли клиента заявки
<b>ticket.getAuthorPersonRoleFormattedTitle(): string</b>	Возвращает форматированное название роли автора заявки
<b>ticket.getAssignedPersonRoleFormattedTitle(): string</b>	Возвращает форматированное название роли исполнителя заявки
<b>ticket.getClientPersonRoleFormattedTitle(): string</b>	Возвращает форматированное название роли клиента заявки

<b>ticket.getPerson1Id(): int null</b>	Возвращает id первой дополнительной персоны
<b>ticket.getPerson2Id(): int null</b>	Возвращает id второй дополнительной персоны
<b>ticket.getPerson3Id(): int null</b>	Возвращает id третьей дополнительной персоны
<b>ticket.setPerson1Id(int null \$personId): bool</b>	Устанавливает id первой дополнительной персоны
<b>ticket.setPerson2Id(int null \$personId): bool</b>	Устанавливает id второй дополнительной персоны
<b>ticket.setPerson3Id(int null \$personId): bool</b>	Устанавливает id третьей дополнительной персоны
<b>ticket.getTotalFormatted(): string</b>	Возвращает форматированную итоговую стоимость заказа
<b>ticket.getSubtotalFormatted(): string</b>	Возвращает форматированную промежуточную стоимость заказа
<b>ticket.getTotalFeeFormatted(): string</b>	Возвращает форматированную стоимость заказа
<b>ticket.getDeliveryFeeFormatted(): string</b>	Возвращает форматированную стоимость доставки
<b>ticket.getTicketLines(): ScriptTicketLine</b>	Возвращает массив объектов ScriptTicketLine

## Методы ScriptTicketLine

Метод	Описание
<b>ScriptTicketLine.reload(): bool</b>	Обновляет данные по строке заявки из бд
<b>ScriptTicketLine.getId(): int null</b>	Возвращает id ScriptTicketLine
<b>ScriptTicketLine.getTicketId(): int null</b>	Возвращает id заявки
<b>ScriptTicketLine.getProductId(): int null</b>	Возвращает id продукта заявки
<b>ScriptTicketLine.getProduct(): ScriptProduct null</b>	Возвращает все данные продукта заявки
<b>ScriptTicketLine.getQuantity(): int null</b>	Возвращает количество продукта в заявке
<b>ScriptTicketLine.getPrice(): float</b>	Возвращает цену продукта в заявке

<code>ScriptTicketLine.getPriceFormatted(): string</code>	Возвращает форматированную цену продукта в заявке
<code>ScriptTicketLine.getPriceDateFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	Возвращает цену продукта в заявке в указанном форматировании
<code>ScriptTicketLine.priceDateFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	
<code>ScriptTicketLine.getFee(): float</code>	Возвращает стоимость заявки
<code>ScriptTicketLine.getFeeFormatted(): string</code>	Возвращает форматированную стоимость заявки
<code>ScriptTicketLine.getSubtotal(): float</code>	Возвращает итоговую стоимость заявки
<code>ScriptTicketLine.getSubtotalFormatted(): string</code>	Возвращает форматированную итоговую стоимость заявки
<code>ScriptTicketLine.getCreatedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	Возвращает форматированную дату создания заказа
<code>ScriptTicketLine.createdAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	
<code>ScriptTicketLine.getUpdatedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	Возвращает форматированную дату обновления заказа
<code>ScriptTicketLine.updatedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	
<code>ScriptTicketLine.getDeletedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	Возвращает форматированную дату удаления заказа
<code>ScriptTicketLine.deletedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	

## Методы ScriptProduct

Метод	Описание
<code>ScriptProduct.reload(): bool</code>	Обновить данные по продукту из бд
<code>ScriptProduct.getId(): int null</code>	Возвращает id продукта

<code>ScriptProduct.getTitle(): string null</code>	Возвращает наименование продукта
<code>ScriptProduct.getCode(): string null</code>	Возвращает код продукта
<code>ScriptProduct.getPrice(): float null</code>	Возвращает цену продукта
<code>ScriptProduct.getMainCategoryId(): int null</code>	Возвращает id категории продукта
<code>ScriptProduct.getIsEnabled(): bool null</code>	Проверяет активен ли продукт
<code>ScriptProduct.getIsAvailableInStock(): bool null</code>	Проверяет есть ли продукт в наличии
<code>ScriptProduct.getUrl(): string null</code>	Возвращает url продукта
<code>ScriptProduct.getThumbUrl(): string null</code>	Возвращает thumb url продукта
<code>ScriptProduct.getPriceFormatted(): string null</code>	Возвращает форматированную цену продукта
<code>ScriptProduct.getCreatedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	Возвращает форматированную дату создания продукта
<code>ScriptProduct.createdAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	
<code>ScriptProduct.getUpdatedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	Возвращает форматированную дату обновления продукта
<code>ScriptProduct.updatedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	
<code>ScriptProduct.getDeletedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	Возвращает форматированную дату удаления продукта
<code>ScriptProduct.deletedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	

## Работа со статусами заявок

### Методы ticketStatus

Метод	Описание
<code>ticketStatus.reload(): bool</code>	Перезагружает статус в БД
<code>ticketStatus.getId(): int</code>	Возвращает id статуса
<code>ticketStatus.getName(): string</code>	Возвращает наименование статуса
<code>ticketStatus.getType(): string null</code>	Возвращает тип статуса
<code>ticketStatus.getOrderPos(): int</code>	Возвращает значение статуса
<code>ticketStatus.isSameStatusName(\$statusName): bool</code>	Проверяет наименование статуса на сходство
<code>ticketStatus.getCreatedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	Возвращает форматированную дату создания статуса
<code>ticketStatus.createdAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	
<code>ticketStatus.getUpdatedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	Возвращает форматированную дату обновления статуса
<code>ticketStatus.updatedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	
<code>ticketStatus.getDeletedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	Возвращает форматированную дату удаления статуса
<code>ticketStatus.deletedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): string null</code>	

## Методы ticketStatuses

Метод	Описание
<code>ticketStatuses.reload(): bool</code>	Перезагружает все статусы из БД
<code>ticketStatuses.getCollection(): Collection null</code>	Возвращает Laravel Collection

ticketStatuses. <b>find</b> (array object null \$where, bool \$isReturnCollection = false): <b>ScriptTicketStatus null</b>	Возвращает статусы заявок
ticketStatuses. <b>findById</b> (string int null \$id): <b>ScriptTicketStatus null</b>	Возвращает статус по ID
ticketStatuses. <b>findByName</b> (string \$name): <b>ScriptTicketStatus null</b>	Возвращает статус по имени
ticketStatuses. <b>findAllByNames</b> (string array object null \$names, string \$operator = '=', bool \$isReturnCollection = false): <b>Collection null</b>	Возвращает статусы по имени
ticketStatuses. <b>findAllByTypes</b> (string array object null \$types, string \$operator = '=', bool \$isReturnCollection = false): <b>Collection null</b>	Возвращает статусы по типу статуса

## Работа с персонами

### Методы bot

Метод	Описание
<b>bot.createPerson(\$data): int null</b>	Создает персону
<b>bot.findPersons</b> (\$where = [], \$orderBy = [], int null \$limit = null, int null \$offset = null, bool \$isLogErrorIfManyPersonsFound = false): <b>array</b>	Возвращает данные персон соответствующих атрибутам
<b>bot.updatePersonById</b> (int \$personId, \$data): <b>bool</b>	Обновляет персону по id
<b>bot.deletePersonById</b> (int \$personId): <b>bool</b>	Удаляет персону по id
<b>bot.getDefaultPersonRoleId(): int null</b>	Возвращает id роли персоны по умолчанию

### Методы lead

Метод	Описание
<b>lead.getPersonId(): int null</b>	Возвращает id персоны

<b>lead.setPersonId(int null \$personId): bool</b>	Привязывает персону к лиду
<b>lead.createPersonForCurrentLead(\$data): int null</b>	Создает персону для лида

## Методы person

Метод	Описание
<b>person.update(\$data): bool</b>	Обновляет данные персоны
<b>person.delete(): bool</b>	Удаляет персону
<b>person.getAllData(): array</b>	Возвращает все поля персоны
<b>person.setData(string \$key, \$value): self</b>	<p>Устанавливает поле персоны по ключу В качестве ключа можно указать:</p> <ul style="list-style-type: none"> <li>• <b>id</b> - id персоны;</li> <li>• <b>business_id</b> - id бизнеса;</li> <li>• <b>bot_id</b> - id бота;</li> <li>• <b>person_role_id</b> - id роли;</li> <li>• <b>is_external</b> - значение поля <b>Сотрудник/Клиент</b>;</li> <li>• <b>user_id</b> - id пользователя;</li> <li>• <b>firstname</b> - имя;</li> <li>• <b>lastname</b> - фамилия;</li> <li>• <b>middlename</b> - отчество;</li> <li>• <b>email</b> - адрес электронной почты;</li> <li>• <b>phone</b> - номер телефона;</li> <li>• <b>organization_name</b> - название организации;</li> <li>• <b>organization_address</b> - адрес организации;</li> <li>• <b>comment</b> - комментарий;</li> <li>• <b>created_at</b> - дата создания;</li> <li>• <b>updated_at</b> - дата последнего редактирования;</li> <li>• <b>deleted_at</b> - дата удаления;</li> <li>• все системные поля персоны (по английски).</li> </ul>

<p><b>person.getData(string \$key)</b></p>	<p>Возвращает значение данных персоны В качестве ключа можно указать:</p> <ul style="list-style-type: none"> <li>• <b>id</b> - id персоны;</li> <li>• <b>business_id</b> - id бизнеса;</li> <li>• <b>bot_id</b> - id бота;</li> <li>• <b>person_role_id</b> - id роли;</li> <li>• <b>is_external</b> - значение поля <b>Сотрудник/Клиент</b>;</li> <li>• <b>user_id</b> - id пользователя;</li> <li>• <b>firstname</b> - имя;</li> <li>• <b>lastname</b> - фамилия;</li> <li>• <b>middlename</b> - отчество;</li> <li>• <b>email</b> - адрес электронной почты;</li> <li>• <b>phone</b> - номер телефона;</li> <li>• <b>organization_name</b> - название организации;</li> <li>• <b>organization_address</b> - адрес организации;</li> <li>• <b>comment</b> - комментарий;</li> <li>• <b>created_at</b> - дата создания;</li> <li>• <b>updated_at</b> - дата последнего редактирования;</li> <li>• <b>deleted_at</b> - дата удаления;</li> <li>• все системные поля персоны (по английски).</li> </ul>
<p><b>person.issetData(string \$key): bool</b></p>	<p>Проверяет наличие данных по персоне В качестве ключа можно указать:</p> <ul style="list-style-type: none"> <li>• <b>id</b> - id персоны;</li> <li>• <b>business_id</b> - id бизнеса;</li> <li>• <b>bot_id</b> - id бота;</li> <li>• <b>person_role_id</b> - id роли;</li> <li>• <b>is_external</b> - значение поля <b>Сотрудник/Клиент</b>;</li> <li>• <b>user_id</b> - id пользователя;</li> <li>• <b>firstname</b> - имя;</li> <li>• <b>lastname</b> - фамилия;</li> <li>• <b>middlename</b> - отчество;</li> <li>• <b>email</b> - адрес электронной почты;</li> <li>• <b>phone</b> - номер телефона;</li> <li>• <b>organization_name</b> - название организации;</li> <li>• <b>organization_address</b> - адрес организации;</li> <li>• <b>comment</b> - комментарий;</li> <li>• <b>created_at</b> - дата создания;</li> <li>• <b>updated_at</b> - дата последнего редактирования;</li> <li>• <b>deleted_at</b> - дата удаления;</li> <li>• все системные поля персоны (по английски).</li> </ul>
<p><b>person.isDataExist(string \$key): bool</b></p>	<p>Проверяет наличие данных по персоне В качестве ключа можно указать:</p> <ul style="list-style-type: none"> <li>• <b>id</b> - id персоны;</li> <li>• <b>business_id</b> - id бизнеса;</li> <li>• <b>bot_id</b> - id бота;</li> <li>• <b>person_role_id</b> - id роли;</li> <li>• <b>is_external</b> - значение поля <b>Сотрудник/Клиент</b>;</li> <li>• <b>user_id</b> - id пользователя;</li> <li>• <b>firstname</b> - имя;</li> <li>• <b>lastname</b> - фамилия;</li> <li>• <b>middlename</b> - отчество;</li> <li>• <b>email</b> - адрес электронной почты;</li> <li>• <b>phone</b> - номер телефона;</li> <li>• <b>organization_name</b> - название организации;</li> <li>• <b>organization_address</b> - адрес организации;</li> <li>• <b>comment</b> - комментарий;</li> <li>• <b>created_at</b> - дата создания;</li> <li>• <b>updated_at</b> - дата последнего редактирования;</li> <li>• <b>deleted_at</b> - дата удаления;</li> <li>• все системные поля персоны (по английски).</li> </ul>
<p><b>person.getLeads(bool \$isFindInAllBusinessBots = false): array</b></p>	<p>Возвращает лиды персоны</p>
<p><b>person.getId(): int null</b></p>	<p>Возвращает ID персоны</p>

<b>person.getFullName(): string</b>	Возвращает полное имя персоны
<b>person.getFIO(): string</b>	
<b>person.getRoleId(): int null</b>	Возвращает ID роли персоны
<b>person.getRoleName(): string</b>	Возвращает название роли персоны
<b>person.getRoleTitle(): string</b>	Возвращает заголовок роли персоны
<b>person.getRoleFormattedTitle(): string</b>	Возвращает форматированное название роли персоны.
<b>person.getPhone(): string null</b>	Возвращает номер телефона персоны
<b>person.setPhone(string null \$phone): bool</b>	Устанавливает номер телефона персоны
<b>person.getEmail(): string null</b>	Возвращает почту персоны

# Работа с очередями исполнителей

## Методы ticket

Метод	Описание
ticket. <b>assignToNextQueuePerson</b> (int \$queueMode = 1, int null \$personQueueId = null, string null \$personQueueTable = null, int null \$currentAssignAttempt = null, bool \$isIncRejectsCount = true, bool \$isMoveToEndIfRejectsExceed = true): <b>bool</b>	Назначает заявку следующему исполнителю
ticket. <b>moveAssignedPersonToEndOfQueue</b> (int \$queueMode = 1, int null \$personQueueId = null, string null \$personQueueTable = null, bool \$isResetRejectsCount = true): <b>bool</b>	Переносит исполнителя в конец очереди
ticket. <b>assignToPersonById</b> (int null \$personId, int null \$personQueueIdForLog = null, string null \$personQueueTableForLog = null, int null \$assignAttemptForLog = null): <b>bool</b>	Назначает исполнителя заявки

<code>ticket.<b>assignToNextPersonAndMoveToEnd(): bool</b></code>	Назначает заявку следующему исполнителю и переносит его в конец очереди
<code>ticket.<b>getAssignedPersonRejectsCount</b>(int \$personQueueId = null, string null \$personQueueTable = null): <b>int null</b></code>	Возвращает количество отказов от заявок
<code>ticket.<b>setAssignedPersonRejectsCount</b>(int \$rejectsCount, int null \$personQueueId = null, string null \$personQueueTable = null): <b>bool</b></code>	Устанавливает количество отказов от заявок
<code>ticket.<b>incAssignedPersonRejectsCount</b>(int null \$personQueueId = null, string null \$personQueueTable = null): <b>bool</b></code>	Увеличивает количество отказов от заявок

## Методы person

Метод	Описание
<code>person.<b>addToQueue</b>(int null \$personQueueId = null, string null \$personQueueTable = null, int null \$ticketTypeId = null, bool \$isActive = true, int \$rejectsCount = 0): <b>int null</b></code>	Добавляет персону в очередь
<code>person.<b>removeFromQueue</b>(int null \$personQueueId = null, string null \$personQueueTable = null, int null \$ticketTypeId = null): <b>bool</b></code>	Удаляет персону из очереди
<code>person.<b>moveToEndOfQueue</b>(int \$queueMode = 1, int null \$personQueueId = null, string null \$personQueueTable = null, int null \$ticketTypeId = null, bool \$isResetRejectsCount = true): <b>bool</b></code>	Перемещает персону в конец очереди
<code>person.<b>setIsActiveInQueue</b>(bool \$isActive = true): <b>bool</b></code>	Активирует/деактивирует персону в очереди

## Методы personQueue

Метод	Описание
<code>personQueue.<b>getId</b>(): <b>int</b></code>	Возвращает id очереди персон
<code>personQueue.<b>getCode</b>(): <b>string</b></code>	Возвращает код очереди персон
<code>personQueue.<b>getIsEnabled</b>(): <b>bool</b></code>	Проверяет активна ли очередь персон

<code>personQueue.getName(): <b>string</b></code>	Возвращает наименование очереди персон
<code>personQueue.isSameName(\$name): <b>bool</b></code>	Проверяет наименование очереди персон на сходство
<code>personQueue.getCreatedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): <b>string null</b></code>	Возвращает форматированную дату создания очереди персон
<code>personQueue.createdAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): <b>string null</b></code>	
<code>personQueue.getUpdatedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): <b>string null</b></code>	Возвращает форматированную дату обновления очереди персон
<code>personQueue.updatedAtFormatted(string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): <b>string null</b></code>	

## Методы personQueues

Метод	Описание
<code>personQueues.find(\$where, bool \$isReturnCollection = false): <b>ScriptPersonQueue[] Collection</b></code>	Возвращает очередь исполнителей
<code>personQueues.findById(string int null \$id): <b>ScriptPersonQueue null</b></code>	Возвращает очередь исполнителей по id
<code>personQueues.findByName(string \$name, bool \$isEnabled = null): <b>ScriptPersonQueue null</b></code>	Возвращает очередь исполнителей по наименованию
<code>personQueues.findAllByNames(\$names, bool \$isEnabled = null, string \$findNameOperator = 'IN', bool \$isReturnCollection = false): <b>ScriptPersonQueue[] Collection</b></code>	Возвращает очереди исполнителей по наименованию
<code>personQueues.findByCode(string \$code, bool \$isEnabled = null, string \$findCodeOperator = '='): <b>ScriptPersonQueue null</b></code>	Возвращает очередь исполнителей по коду
<code>personQueues.findAllByCodes(\$codes, bool \$isEnabled = null, string \$findCodeOperator = 'IN', bool \$isReturnCollection = false): <b>ScriptPersonQueue[] Collection</b></code>	Возвращает очереди исполнителей по коду

<code>personQueues.reload(): bool</code>	Обновляет очередь исполнителей
<code>personQueues.getCollection(): Collection null</code>	Возвращает коллекцию очередей

# Работа с интеграциями NLP

## Методы bot

Метод	Описание
<code>bot.getIntegrationIdByCode(string null \$integrationCode): int null</code>	Возвращает id интеграции бота по коду
<code>bot.getEnabledIntegrationIdByCode(string null \$integrationCode): int null</code>	Возвращает id включенной интеграции бота по коду
<code>bot.isEnabledIntegrationById(int null \$integrationId): bool</code>	Проверяет активность интеграции по id
<code>bot.isEnabledIntegrationByCode(string null \$integrationCode): bool</code>	Проверяет активность интеграции по коду

## Методы lead

Метод	Описание
<code>lead.setActiveNlpIntegrationByCode(string null \$integrationCode): bool</code>	Присваивает интеграцию лиду по коду
<code>lead.setActiveNlpIntegrationById(int null \$nlpIntegrationId): bool</code>	Присваивает интеграцию лиду по id
<code>lead.hasActiveNlpIntegration(): bool</code>	Проверяет активна ли интеграция
<code>lead.getActiveNlpIntegrationId(int null \$nlpIntegrationId): int null</code>	Возвращает интеграцию по id
<code>lead.getActiveNlpIntegrationCode(string null \$integrationCode): string null</code>	Возвращает интеграцию по коду
<code>lead.getActiveOrFirstNlpIntegrationId(int null \$nlpIntegrationId): int null</code>	Возвращает активную или первую интеграцию по id

lead.**getActiveOrFirstNpIntegrationCode**(string|null  
\$integrationCode): **string|null**

Возвращает активную или первую интеграцию по коду

## Работа с интеграциями

### Методы lead

Метод	Описание
lead. <b>getForwardEventTypeForFB</b> (): <b>string null</b>	Возвращает тип события для ФБ

## Работа с кастомными таблицами

### Методы table

Метод	Описание
table. <b>createItem</b> (string \$tableName, array object \$data): <b>ScriptCustomTableItem</b>	Создает запись в таблице
table. <b>find</b> (string \$tableName, array object \$columns = [], array object \$where = [], array object \$orderBy = [], int null \$limit = null, int null \$offset = null): <b>array</b>	Ищет запись в таблице
table. <b>count</b> (string \$tableName, array object \$where = []): <b>int</b>	Подсчитывает количество записей
table. <b>sum</b> (string \$tableName, string \$column, array object \$where = []): <b>int</b>	Подсчитывает сумму значений по полю
table. <b>max</b> (string \$tableName, string \$column, array object \$where = []): <b>mixed null</b>	Ищет максимальное значение по полю

### Методы item

За объект **item** принимаем запись кастомной таблицы полученную с помощью **table.createItem** или **table.find**.

Метод	Описание
-------	----------

<code>item.update(array object \$data): <b>bool</b></code>	Обновляет записи в таблице
<code>item.delete(): <b>bool</b></code>	Удаляет записи из таблицы
<code>item.getDateFormatted(string \$fieldName, string \$format = 'Y-m-d H:i:s', string null \$timeZone = null): <b>string null</b></code>	Возвращает форматированную дату по названию поля
<code>item.getDateFormattedString(string DateTimeInterface \$date, string \$toFormat = 'Y-m-d H:i:s', string null \$fromFormat = 'Y-m-d H:i:s', string null \$toTimeZone = null, string null \$fromTimeZone = null): <b>string null</b></code>	Возвращает форматированную дату из произвольной строки
<code>item.reload(): <b>bool</b></code>	Перезагружает данные записи из БД

## Работа с блокировками

Метод	Описание
<code>bot.getLockNameForBot(string \$lockName, string \$lockPrefix = "): <b>string</b></code>	Генерирует имя блокировки по боту
<code>bot.getLockNameForBusiness(string \$lockName, string \$lockPrefix = "): <b>string</b></code>	Генерирует имя блокировки по бизнесу
<code>bot.waitForBusinessLock(string \$lockName, string \$lockPrefix = ", int null \$ttlSec = null, \$maxWaitSec = 300): <b>bool</b></code>	Захватывает блокировку по бизнесу
<code>bot.waitForBotLock(string \$lockName, string \$lockPrefix = ", int null \$ttlSec = null, \$maxWaitSec = 300): <b>bool</b></code>	Захватывает блокировку по боту
<code>bot.releaseAllCurrentLocks(): <b>bool</b></code>	Освобождает все блокировки захваченные в текущем скрипте
<code>bot.releaseCurrentLockForBusiness(string \$lockName, string \$lockPrefix = "): <b>bool</b></code>	Освобождает блокировку захваченную текущим скрипте по бизнесу
<code>bot.releaseCurrentLockForBot(string \$lockName, string \$lockPrefix = "): <b>bool</b></code>	Освобождает блокировку захваченную в текущем скрипте по боту

<code>bot.releaseLockForBusiness(string \$lockName, string \$lockPrefix = "): bool</code>	Освобождает блокировку по бизнесу
<code>bot.releaseLockForBot(string \$lockName, string \$lockPrefix = "): bool</code>	Освобождает блокировку по боту
<code>bot.hasLockForBusiness(string \$lockName, string \$lockPrefix = "", \$checkValue = null): bool</code>	Проверяет существование блокировки по бизнесу
<code>bot.hasLockForBot(string \$lockName, string \$lockPrefix = "", \$checkValue = null): bool</code>	Проверяет существование блокировки по боту

## Работа с контакт-центром

### Методы bot

Метод	Описание
<code>bot.canAutoForward(): bool</code>	Узнает доступно ли авто-назначение оператора в данном скрипте

### Методы lead

Метод	Описание
<code>lead.canAutoForward(): bool</code>	Узнает доступно ли авто-назначение оператора в данном скрипте
<code>lead.getNextOperators(string \$departmentCode = null, bool \$isOnline = true, int null \$limit = 1, bool \$useLocks = true): int null</code>	Возвращает список следующих операторов
<code>lead.assignDialogToNextOperator(string null \$departmentCode = null, bool \$isOnline = true, bool \$isForwardIfOperatorNotFound = true, bool \$useLocks = true): int null</code>	Назначает диалог на следующего оператора полученного с помощью <b>getNextOperators</b>
<code>lead.assignDialogToOperator(int null \$operatorId, bool \$isForwardIfOperatorNotFound = true, bool \$useLocks = true): int null</code>	Назначает диалог на указанного в методе оператора

# Работа с API

## Методы bot

Метод	Описание
<code>bot.scheduleJob(\$jobParams = []): bool</code>	Планирует любую задачу
<code>bot.randomStr(int \$length = 16): string</code>	Генерирует случайную буквенно-цифровую строку
<code>bot.newGuid(): string</code>	Генерирует случайный уникальный guid
<code>bot.newOrderedGuid(): string</code>	Генерирует случайный уникальный guid упорядоченный по времени

## Методы lead

Метод	Описание
<code>lead.getLastRequest(): ScriptRequest null</code>	Возвращает объект с параметрами последнего реквеста
<code>lead.getLang(): string null</code>	Возвращает язык установленный в бизнесе
<code>lead.setLang(string null \$langCode): string null</code>	Устанавливает язык в бизнесе
<code>lead.isLangExist(string \$langCode, bool null \$isEnabled = true): bool</code>	Проверяет существует ли язык

## Методы request

За объект **request** принимаем объект с параметрами последнего реквеста полученный с помощью **lead.getLastRequest**.

Метод	Описание
<code>request.json</code>	Доступ к json параметрам в виде javascript json-объекта
<code>request.asJson(): ScriptRequest null</code>	

<code>request.array</code>	Доступ к json параметрам в виде javascript json-объекта
<code>request.asArray(): <b>array</b></code>	
<code>request.string</code>	Входящие параметры в виде json объекта
<code>request.asString(): <b>string</b></code>	

## Методы last\_request

За объект **last\_request** принимаем объект с параметрами последнего реквеста полученный с помощью **lead.getLastRequest**.

Метод	Описание
<code>last_request.json</code>	Доступ к последнему реквесту в виде javascript json-объекта
<code>last_request.asJson(): <b>ScriptRequest null</b></code>	
<code>last_request.array</code>	Доступ к последнему реквесту в виде массива
<code>last_request.asArray(): <b>array</b></code>	
<code>last_request.string</code>	Доступ к последнему реквесту в виде json объекта
<code>last_request.asString(): <b>string</b></code>	

## Методы api

Метод	Описание
<code>api.callEndpoint(   string \$externalApiEndpointCode,   string null \$uri = null,   string null \$method = null,   string null \$format = null,   array object string null \$body = null,   array object string null \$headers = [],   array object string null \$requestOptions = [],   array object string null \$httpClientConfig = [],   string null \$proxyUrl = null ): <b>array string null</b></code>	Вызывает внешнее API из чат-бота

<p><code>api.request(string \$method, string \$uri, \$requestOptions = [], \$httpClientConfig = [], \$body = null, \$jsonBody = null, \$headers = [], bool \$isForceAddJsonHeaders = false): <b>ResponseInterface null mixed</b></code></p>	Выполняет HTTP запрос, с помощью Guzzle
<p><code>api.post(string \$uri, \$body = [], \$requestOptions = [], \$headers = [], \$httpClientConfig = []): <b>string null</b></code></p>	Выполняет HTTP POST запрос, с помощью Guzzle
<p><code>api.get(string \$uri, \$requestOptions = [], \$headers = [], \$httpClientConfig = []): <b>string null</b></code></p>	Выполняет HTTP GET запрос, с помощью Guzzle
<p><code>api.getJson(string \$uri, \$jsonBody = [], \$headers = [], \$requestOptions = [], \$httpClientConfig = []): <b>array null</b></code></p>	Выполняет HTTP POST запрос с jsonBody, с помощью Guzzle
<p><code>api.postJson(string \$uri, \$jsonBody = [], \$headers = [], \$requestOptions = [], \$httpClientConfig = []): <b>array null</b></code></p>	Выполняет HTTP GET запрос с jsonBody, с помощью Guzzle
<p><code>api.getHttpClientConfig(): <b>array</b></code></p>	Возвращает HTTP Config
<p><code>api.setHttpClientConfig(\$config): <b>self</b></code></p>	Устанавливает HTTP Config
<p><code>api.getRequestOptions(): <b>array</b></code></p>	Возвращает параметры HTTP запроса
<p><code>api.setRequestOptions(\$options): <b>self</b></code></p>	Устанавливает параметры HTTP запроса
<p><code>api.getHeaders(): <b>array</b></code></p>	Возвращает заголовки HTTP запроса
<p><code>api.setHeaders(\$headers): <b>self</b></code></p>	Устанавливает заголовки HTTP запроса
<p><code>api.getProxyUrl(): <b>string null</b></code></p>	Возвращает Proxy Url HTTP запроса
<p><code>api.setProxyUrl(\$proxyUrl): <b>self</b></code></p>	Устанавливает Proxy Url HTTP запроса
<p><code>api.getLastResponse(): <b>mixed ResponseInterface null</b></code></p>	Возвращает последний ответ HTTP запроса
<p><code>api.getLastResponseCode(): <b>int</b></code></p>	Возвращает код последнего ответа HTTP запроса
<p><code>api.getLastResponseContent(): <b>string null</b></code></p>	Возвращает содержимое последнего ответа HTTP запроса
<p><code>api.getLastJsonResponse(): <b>array null</b></code></p>	Возвращает последний ответ HTTP запроса в формате JSON

<code>api.enableExceptions(): self</code>	Включает исключения для HTTP запросов
<code>api.disableExceptions(): self</code>	Выключает исключения для HTTP запросов
<code>api.setTimeout(float \$timeout): self</code>	Устанавливает время задержки для HTTP запросов
<code>api.getTimeout(): float</code>	Возвращает время задержки для HTTP запросов
<code>api.isExceptionThrown(): bool</code>	Проверяет существует ли исключение
<code>api.isConnectExceptionThrown(): bool</code>	Проверяет существует ли исключение подключения
<code>api.getLastResponseTime(): float</code>	Возвращает время последнего ответа HTTP запроса
<code>api.getLastErrorBody(bool \$isAutoConvertToJson = true): string array null</code>	Возвращает содержимое ошибки последнего ответа HTTP запроса
<code>api.enableContinueCallEndpointOnJsError(): self</code>	Включает вызов эндпоинта при ошибке JS
<code>api.disableContinueCallEndpointOnJsError(): self</code>	Выключает вызов эндпоинта при ошибке JS
<code>api.getIsContinueCallEndpointOnJsError(): bool</code>	Возвращает режим вызова эндпоинта при ошибке JS

## Работа с файлами

### Методы bot

Метод	Описание
<code>bot.getIncomingMessage(bool \$useOutputFilter = false): string</code>	Возвращает текст входящего сообщения или текст подписи к вложению
<code>bot.getWebhookJobId(): int null</code>	Возвращает id вебхука зарегистрированного в очереди Метабот
<code>bot.getWebhookJobRecord(): BotWebhookQueueRecord null</code>	Возвращает запись вебхука зарегистрированного в очереди Метабот
<code>bot.getWebhookJob(): array</code>	Возвращает информацию о вебхуке

<code>bot.<b>getWebhookPayload</b>()</code> : <b>array</b>	Возвращает payload входящего вебхука в виде массива
<code>bot.<b>getLastResultPayload</b>()</code> : <b>array null</b>	Возвращает payload с данными о последнем ответе мессенджера в виде массива
<code>bot.<b>getChannelData</b>(string null \$key = null)</code> : <b>array null mixed</b>	Возвращает настройки канала лида
<code>bot.<b>disableBuffer</b>()</code>	Принудительное отключение буфера исходящих сообщений
<code>bot.<b>enableBuffer</b>()</code>	Принудительное включение буфера исходящих сообщений
<code>bot.<b>flushBuffer</b>()</code>	Принудительное освобождение буфера исходящих сообщений
<code>bot.<b>getAllAttachments</b>(string null \$filesMimeExcludeFilter = 'image video')</code> : <b>array</b>	Возвращает вложения всех видов
<code>bot.<b>getFiles</b>(string null \$mimeIncludeFilter = null, string null \$mimeExcludeFilter = 'image video')</code> : <b>array</b>	Возвращает все вложения, которые распознаны как файл
<code>bot.<b>getImages</b>(string null \$mimeIncludeFilter = 'image')</code> : <b>array</b>	Возвращает все изображения, а также файлы, которые распознаны как изображение
<code>bot.<b>getVideos</b>(string null \$mimeIncludeFilter = 'video')</code> : <b>array</b>	Возвращает все видео, а также файлы, которые распознаны как видео
<code>bot.<b>sendPayload</b>(string null \$endpoint, array object \$payload)</code> : <b>array string mixed</b>	Отправляет payload в мессенджер
<code>bot.<b>sendText</b>(string null mixed \$messageText, array object null \$apiAdditionalParams = null, bool \$isDisableBuffer = true)</code> : <b>bool</b>	Отправляет текст в мессенджер
<code>bot.<b>sendMessage</b>(string null mixed \$messageText, array object null \$buttons = null, array object null \$attachments = null, array object null \$apiAdditionalParams = null, bool \$isDisableBuffer = true, bool \$isUpdateMessageButtons = false)</code> : <b>bool</b>	Отправляет сообщение в мессенджер, сообщение может содержать текст, кнопки меню и вложения

<p><b>bot.sendButtons</b>(string null mixed \$messageText, array object \$buttons, array object null \$apiAdditionalParams = null, bool \$isDisableBuffer = true, bool \$isUpdateMessageButtons = false): <b>bool</b></p>	Отправляет кнопки с сообщением в мессенджер
<p><b>bot.getTelegramFileInfoById</b>(string null \$fileId, bool \$isShowOriginalInfo = false): <b>array null</b></p>	Возвращает информацию о файле в Telegram по его id
<p><b>bot.sendTelegramAttachmentAndGetFileId</b>(string null \$messageText, array object null \$attachment, array object null \$buttons = null, array object null \$apiAdditionalParams = null): <b>string null</b></p>	Отправляет файл в телеграм и возвращает его id
<p><b>bot.editTelegramInlineKeyboard</b>(int string \$chatId, int string \$messageId, string null string \$inlineMessageId, array object null \$inlineKeyboardMarkup): <b>array null</b></p>	Редактирует кнопки меню уже отправленного в Telegram сообщения
<p><b>bot.removeTelegramInlineKeyboard</b>(int string \$chatId, int string \$messageId, string null \$inlineMessageId = null): <b>array null</b></p>	
<p><b>bot.getTelegramLastMessageId</b>(): <b>int null</b></p>	Возвращает ID последнего сообщения, отправленного в Telegram
<p><b>bot.setCdnConfig</b>(array object null \$cdnConfig = null): <b>bool</b></p>	Устанавливает конфиг для доступа к CDN
<p><b>bot.getCdnConfig</b>(): <b>array null</b></p>	Возвращает текущий конфиг CDN
<p><b>bot.getFileInfoByUrl</b>(string \$fileUrl): <b>array</b></p>	Возвращает информацию о файле по URL
<p><b>bot.downloadFileFromUrl</b>(string \$fileUrl): <b>string array</b></p>	Скачивает файл с URL и загружает в хранилище файлов Метабот для вашего бизнеса
<p><b>bot.downloadTmpFileFromUrl</b>(string \$fileUrl): <b>string array</b></p>	Скачивает файл с URL и загружает во временное хранилище файлов Метабот для вашего бизнеса
<p><b>bot.removeFile</b>(string \$fileUrl): <b>string null</b></p>	Удаляет файл по URL из хранилища файлов Метабот для вашего бизнеса
<p><b>bot.removeTmpFile</b>(string \$fileUrl): <b>string null</b></p>	Удаляет файл по URL из временного хранилища файлов Метабот для вашего бизнеса

<code>bot.uploadFileToCdnAndGetNewUrl(string \$fileUrl, array object null \$cdnConfig = null): array</code>	Заливает файл на CDN и возвращает новую ссылку
<code>bot.removeFileFromCdn(string \$fileUrl, \$cdnConfig = null)</code>	Удаляет файл из CDN

## Методы lead

Метод	Описание
<code>lead.getMessenger(): int</code>	Возвращает обозначение текущего мессенджера канала
<code>lead.getMessengerCode(): string</code>	Возвращает буквенное обозначение текущего мессенджера канала
<code>lead.getIdentification(): string null</code>	Возвращает идентификатор лида во внешней системе
<code>lead.getIdentificationEntity(): string null</code>	Возвращает дополнительный идентификатор лида во внешней системе

## Работа с GPT

### Методы bot

Метод	Описание
<code>bot.run(array object null \$jobParams = []): bool</code>	Запускает скрипт в рамках текущего потока бота
<code>bot.stop(): bool</code>	Прерывает текущий скрипт
<code>bot.throw(string \$message = "", int \$code = 0): bool</code>	Создает искусственное падение скрипта
<code>bot.disableRepeatMessageText(): bool</code>	Отключает вывод сообщения для команды конструктора "Повтор" для текущего вебхука/задания
<code>bot.enableRepeatMessageText(): bool</code>	Включает вывод сообщения для команды конструктора "Повтор" для текущего вебхука/задания

<code>bot.isEnabledRepeatMessageText(): bool</code>	Проверяет включен или нет вывод сообщения для команды конструктора "Повтор" для текущего вебхука/задания
<code>bot.hideRepeatMessageButtons(): bool</code>	Отключает вывод меню для команды конструктора "Повтор" для текущего вебхука/задания
<code>bot.showRepeatMessageButtons(): bool</code>	Включает вывод меню для команды конструктора "Повтор" для текущего вебхука/задания
<code>bot.isShowRepeatMessageButtons(): bool</code>	Проверяет включен или нет вывод меню для команды конструктора "Повтор" для текущего вебхука/задания
<code>bot.getConversationQuestion(): OverridenQuestion null</code>	Возвращает текущее состояние беседы
<code>bot.getCurrentScriptId(): int null</code>	Возвращает ID текущего скрипта
<code>bot.getCurrentScriptCode(): string null</code>	Возвращает код текущего скрипта
<code>bot.getCurrentCommandId(): int null</code>	Возвращает код текущей команды
<code>bot.getConversationButtons(): array</code>	Возвращает текущие кнопки
<code>bot.getMenuItemsForScript(?string \$scriptId): array</code>	Возвращает все пункты меню скрипта
<code>bot.getMenuItemsForCurrentScript(): array</code>	Возвращает все пункты меню текущего скрипта
<code>bot.getButtonsForScript(?string \$scriptId, ?int \$scriptId = null, bool \$withRunScriptId = true): array</code>	Возвращает все кнопки скрипта
<code>bot.getButtonsForCurrentScript(bool \$withRunScriptId = true): array</code>	Возвращает все кнопки текущего скрипта
<code>bot.findButtonByIncomingMessage(array object \$buttons, ?string \$overrideIncomingMessage = null, bool \$isLogFoundButton = true): array null</code>	Находит кнопку которую "нажимает пользователь" по входящему сообщению
<code>bot.getScriptIdFromButtonsByIncomingMessage (array object \$buttons, ?string \$overrideIncomingMessage = null, bool \$isLogFoundButton = true): int null</code>	Возвращает id скрипта по кнопке которую "нажимает пользователь"

<b>bot.getScriptCodeFromButtonsByIncomingMessage</b> (array object \$buttons, ?string \$overrideIncomingMessage = null, bool \$isLogFoundButton = true): <b>string null</b>	Возвращает код скрипта по кнопке которую "нажимает пользователь"
--	--

## Методы api

Метод	Описание
<b>api.getLastErrorBody</b> (bool \$isAutoConvertToJson = true): <b>string array null</b>	Возвращает содержимое API ответа, если внешнее API вернуло код ошибки
<b>api.enableContinueCallEndpointOnJsError</b> (): <b>bool</b>	Игнорирует ошибку в JS Body
<b>api.disableContinueCallEndpointOnJsError</b> (): <b>bool</b>	Отключает игнорирование ошибки в JS Body
<b>api.getIsContinueCallEndpointOnJsError</b> (): <b>bool</b>	Проверяет вкл. или выключено игнорирование ошибки в JS Body перед запуском внешнего api
<b>api.callApiOnlyByOkFlag</b> (bool \$value = true): <b>bool</b>	Выполняет обращение по api, только если JS валиден
<b>api.onlyForOk</b> (bool \$value = true): <b>bool</b>	
<b>api.isCallApiOnlyByOkFlag</b> (): <b>bool</b>	Проверяет включен или нет флаг isCallApiOnlyByOkFlag
<b>api.isOnlyForOk</b> (): <b>bool</b>	
<b>api.ok</b> (bool \$value = true): <b>bool</b>	Указывает что JS валиден, для того чтобы выполнялся запуск внешнего api
<b>api.isOk</b> (): <b>bool</b>	Проверяет валиден ли JS

## Работа с отладкой

Метод	Описание
<b>bot.sysLogInfo</b> (\$data, bool \$isSendMessageToLead = false): <b>bool</b>	Добавляет в лог системы информацию
<b>bot.sysLogError</b> (\$data, bool \$isSendMessageToLead = false): <b>bool</b>	Добавляет в лог системы информацию об ошибке
<b>bot.sysLogInfoToAdmin</b> (\$data): <b>bool</b>	Добавляет в лог системы информацию для админа
<b>bot.sysLogErrorToAdmin</b> (\$data): <b>bool</b>	Добавляет в лог системы информацию об ошибке для админа

`bot.debug($data, bool $isStopBot = false): bool`

Выводит информацию в мессенджер

Версия #25

Ирина Петрова создал 28 February 2024 09:20:33

Ирина Петрова обновил 26 March 2025 08:03:18