table.find()

Поиск записей

```
table.find(string $tableName, array|object $columns = [], array|object $where = [],
array|object $orderBy = [], ?int $limit = null, ?int $offset = null): array
```

Описание:

Ищет записи в таблице и возвращает их в виде массива объектов.

Атрибуты:

Имя	Тип	Описание
tableName	string	Наименование таблицы, в которой будет найдена запись
columns	array | object (json{})	Столбцы которые будут возвращены в ответе. При поиске из основной таблицы указывается только наименование поля, при поиске в связанной таблице указывается в формате: **алиас.поле**. Если нужно тянуть несколько полей с одинаковыми именами, то указываем алиас результирующего поля, например: 'products.id as product_id'
where	array | object (json{})	Условие, которое будет проверяться при поиске
orderBy	array | object (json{})	Устанавливает в какой последовательности будут возвращены выходные значения
limit	int	Устанавливает максимальное количество возвращенных значений
offset	int	Устанавливает смещение

> Чтобы обратится к display field, который автоматически тянется из RELATION в v8 необходимо указать {алиасТаблицы}_{имяПоля}.

> Например, если в RELATION указан алиас **products** и в display field указано поле **name,** то в результирующую выборку это поле попадает под именем **products_name**.

Возвращает:

array - массив объектов (найденных записей).

Примеры:

Получение **всех** регионов (лимит 100):

```
let items = table.find('regions');
let regions = [];
for(let i = 0; i < items.length; i++) { // добавление поочередно всех записей в объект regions
  let item = items[i];
  regions.push({"id": item.id, "name": item.name, "comment": item.comment});
}
```

Получение **трех** значений поля **region** с **id больше двух**:

```
let columns = ['region'];
let where = [["id", ">", 2]];
let orderBy = [];
let limit = 3;
let offset = 0;

let items = table.find('regions', columns, where, orderBy, limit, offset);
```

Получение данных из связанной таблицы **leads**:

```
let columns = ['leads.person_id'];
let where = [["leads.id", ">", 2]];

let items = table.find('orders', columns, where);

let items2 = table.find('orders')
debug(items2[0].products_name)
```

Использование оператора **IN**:

```
 ["stage_code", "IN", ["planned", "active", "submission"]]
```

Версия #4
Ирина Петрова создал 2023-07-13 08:24:58 UTC
Artem Garashko обновил 2025-05-23 07:47:01 UTC