

Получение информации о регистре пациента по коду

- Запрос
 - Адрес запроса
 - Параметры запроса
- Ответ сервера
 - Описание полей
 - Возможные ошибки

Запрос

Адрес запроса

GET: /api/register/byCode

Параметры запроса

Параметр	Тип	Описание	По умолчанию	Обязательный	Комментарий
registerCode	query	Код регистра		+	
personGuid	query	гид персоны		+/-	Набор данных для поиска 1
mkabGuid	query	гид МКАБа персоны		+/-	Набор данных для поиска 2
surname	query	Фамилия персоны		+/-	Набор данных для поиска 3
name	query	Имя персоны		+/-	Набор данных для поиска 3
patronymic	query	Отчество персоны		+/-	Набор данных для поиска 3
birthday	query	Дата рождения персоны		+/-	Набор данных для поиска 3
snils	query	Снилс персоны		+/-	Набор данных для поиска 3
policySeries	query	Серия полиса персоны		+/-	Набор данных для поиска 3
policyNumber	query	Номер полиса персоны		+/-	Набор данных для поиска 3

Ответ сервера

В случае успешного выполнения сервер вернет статус 200
Модель: [Информация о нахождении в регистре - RegisterItem](#)

Описание полей

Название	Тип данных	Описание	Комментарий
RegisterMembershipGuid	Guid	Уникальный идентификатор	
RegisterMembershipNum	string	Внешний идентификатор	
Register	BaseHandbookModel	Вид регистра	
Status	BaseHandbookModel	Статус	
Condition	BaseHandbookModel	Состояние	
ExclusionReason	BaseHandbookModel	Причина исключения	
Diagnosis	array of BaseHandbookModel	Диагнозы	
Lpu	BaseHandbookModel	Информация об ЛПУ	
LastUpdated	DateTime	Дата последнего обновления записи	
DateBegin	DateTime	Дата начала действия	
DateEnd	DateTime	Дата окончания действия	

DocPRVD	Guid	Идентификатор должности врача, включившего в регистр	
DocFIO	string	ФИО врача	
LpuGuid	Guid	Идентификатор ЛПУ	
Reason	string	Причина включения в регистр	
Note	string	Примечание	
RejectionReason	string	Причина отказа	
HasSocialServices	bool	Признак получения набора соцслуж	

Пример модели в формате JSON

```
{
  "registerMembershipGuid": "cbcd35f3-c016-453b-b80c-c3ee56abb4dc",
  "register": {
    "isNeedApprove": false,
    "isDoctorCanAdd": true,
    "isAutoInclude": false,
    "code": "covid19",
    "name": " COVID-19"
  },
  "status": {
    "code": "3",
    "name": ""
  },
  "condition": {
    "code": "",
    "name": ""
  },
  "exclusionReason": {
    "code": "",
    "name": ""
  },
  "deathCause": {
    "code": "",
    "name": ""
  },
  "diagnosis": [
    {
      "code": "U07.1",
      "name": "COVID-19, COVID-19"
    }
  ],
  "lpu": {
    "code": "CDD95115-0D38-42A5-97D6-2EBA34329A36",
    "name": " . ( \" \")"
  },
  "lastUpdated": "2022-04-15T12:23:01",
  "registerMembershipNum": "",
  "rejectionReason": "",
  "hasSocialServices": false,
  "dateBegin": "2022-02-10T11:30:05",
  "dateEnd": "2222-01-01T00:00:00",
  "docPRVD": "0abc0ed3-9947-4791-95d8-83232401be00",
  "docFIO": " ",
  "lpuGuid": "cdd95115-0d38-42a5-97d6-2eba34329a36",
  "reason": "",
  "note": ""
}
```

Возможные ошибки

Код ответа сервера	Код сообщения	Сообщение	Тип ошибки
400	4000	Ошибка связи с учреждением	Validation

404	4000	Запись не найдена	Error
-----	------	-------------------	-------

В случае возникновения ошибок будет возвращен [стандартный ответ сервера](#);

В случае возникновения ошибок валидации сервер вернет стандартное [сообщение валидации](#).

Примечание

Метод используется без проверки авторизации