Инструкция программиста системы «ЖИЛФОНД»

Содержание

Введение 3

Методы авторизации 4

Методы работы с информацией 5

Описание таблиц базы данных zhilfond 8

Описание таблиц базы данных pdn 40

Описание прочих объектов базы данных zhilfond 41

# Введение

Работа с информацией системы «Жилфонд» возможна не только c помощью реализованного веб-приложения, но и с использованием сторонних приложений при помощи Веб-API. <http://msdn.microsoft.com/ru-ru/library/hh833994(v=vs.108).aspx>

# Методы авторизации

***HttpResponseMessage Post(Token item)*** – метод, позволяющий авторизовать пользователя по логину и паролю, полученному из параметра item, создаёт сессию для пользователя в случае если авторизация прошла успешно.

Урл: api/tokensapi/

Параметры урл: нет.

Параметры метода: Token item – объект класса Token, в поля которого записаны логин пользователя и пароль.

[Возвращаемое значение](javascript:void(0)): объект класса HttpResponseMessage.

***HttpResponseMessage*** – класс, входящий в .NET Framework[***http://msdn.microsoft.com/ru-ru/library/system.net.http.httpresponsemessage(v=vs.118).aspx***](http://msdn.microsoft.com/ru-ru/library/system.net.http.httpresponsemessage(v=vs.118).aspx)

***Token*** – класс библиотеки Zhilfond

public class Token

{

public string UserName { get; set; }

public string Password { get; set; }

public string Key { get; set; }

}

}, где

UserName – логин пользователя

Password – пароль пользователя

Key – уникальный код сессии

# Методы работы с информацией

***dynamic Get()*** – метод, позволяющий выбрать значения из форм, связанных с домом отношением «один ко многим», т.е. из форм блока №1.

Блок форм №1

* [Основные данные](http://localhost:7716/Main)
* [Земельные участки](http://localhost:7716/Main)
* [Инженерные системы](http://localhost:7716/Main)
* [Данные по ремонтам](http://localhost:7716/Main)
* [Лифты](http://localhost:7716/Main)
* [Поставщики услуг](http://localhost:7716/Main)
* [Данные по аварийным домам](http://localhost:7716/Main)
* [Лицевые счета](http://localhost:7716/Main)
* [Люди](http://localhost:7716/Main)
* [Финансы](http://localhost:7716/Main)
* [Семья](http://localhost:7716/Main)
* [Документы](http://localhost:7716/Main)
* [Виды услуг](http://localhost:7716/Main)
* [Данные по услугам](http://localhost:7716/Main)

Урл: api/valuesapi/

Параметры урл:

form – имя формы из списка «Блок форм №1»

id\_house – идентификатор дома

id\_parent – идентификатор сущности, связанной с таблицей домов отношением «один ко многим» (например Пассажирский Лифт дома и Грузовой Лифт дома будут иметь разные id\_parent, но одинаковые form и id\_house)

token – идентификатор сессии

Параметры метода: нет.

[Возвращаемое значение](javascript:void(0)): IEnumerable<ZValueV>

***ZValueV*** – класс библиотеки Zhilfond

public class ZValueV

{

public int Id { get; set; }

public int Id\_house { get; set; }

public int Id\_form { get; set; }

public int Id\_field { get; set; }

public string Value { get; set; }

public int Id\_dict { get; set; }

public string Dictname { get; set; }

public string Title { get; set; }

public int Id\_category { get; set; }

public string Category { get; set; }

public int? Id\_parent { get; set; }

public string Warning { get; set;}

}, где

Id – идентификатор поля из таблицы Fields

Id\_house – идентификатор дома из урл

Id\_form – идентификатор формы из таблицы Forms

Id\_field – идентификатор поля из таблицы FieldCategories

Value – значение поля из таблицы ValuesAll\_v

Id\_dict – идентификатор справочника из таблицы Dicts (для установки связи с таблицей Справочников, в случае если тип поля является ссылочным; там же в справочнике прописаны значения для задания ссылочных типов данных)

Dictname – имя ссылочного типа данных из таблицы Dicts (в том числе простейшие типы данных «строковое», «С плавающей точкой»)

Title – название поля из таблицы Fields

Id\_category – идентификатор категории из таблицы Categories

Category – название категории поля из таблицы Categories

Id\_parent – идентификатор из урл

Warning – поле для хранения предупреждения по правилам и формулам

***HttpResponseMessage Post(JObject jsonData)*** – метод, позволяющий добавить данные в систему.

Урл: api/valuesapi/

Параметры урл: нет.

Параметры метода: JObject jsonData – объект класса JObject, в поля которого записаны данные для добавления в БД, где

jsonData.Token – идентификатор сессии пользователя,

jsonData.Xml – данные для вставки в БД в формате XML (формат XML-документа предоставляется разработчиками)

jsonData.Signature – цифровая подпись данных

jsonData.SourceType – зарезервированное поле для импортирования файлов, не обязательно к заполнению

jsonData.Id\_source – зарезервированное поле для импортирования файлов, не обязательно к заполнению

# Описание таблиц базы данных zhilfond

***TABLE "Attachments"***

//прикреплённые файлы домов

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_house" integer, // код дома

"Filename" character varying(255) NOT NULL, // имя файла

"Data" bytea, // содержимое файла

"Order" integer, // поле сортировки

"Archive" boolean, // флаг: отображать файл или только хранить в бд

)

***TABLE "Audit"***

//аудит изменений, сделанных пользователями

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_house" integer, // код дома

"Id\_user" integer, // код пользователя

"OldVal" character varying(1024), // предыдущее значение поля

"NewVal" character varying(1024), // новое значение поля

"WhenDateTime" timestamp without time zone NOT NULL, // дата и время внесения правки

"Id\_form" integer NOT NULL, // код формы, в которую вносится правка

"Id\_field" integer NOT NULL, // код поля, в которое вносится правка

)

***TABLE "Categories"***

// категории полей

(

"Id" serial NOT NULL, // идентификатор таблицы

"Title" character varying(255), // название категории

"Order" integer NOT NULL, // поле сортировки

)

***TABLE "Dicts"***

// типы полей (справочники + простые типы данных)

(

"Id" serial NOT NULL, // идентификатор таблицы,

//до 1000 – справочники, больше 1000 – простые типы данных

"Title" character varying(255) NOT NULL, // название справочника

)

***TABLE "DictsValues"***

//значения справочников

(

"Id" integer NOT NULL, // идентификатор таблицы

"Id\_dict" integer NOT NULL, // идентификатор словаря

"Value" character varying(1024), // значение справочника

)

***TABLE "FieldCategories"***

// таблица соответствия полей и категорий

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_category" integer NOT NULL, // код категории

"Id\_field" integer NOT NULL, // код поля

)

***TABLE "Fields"***

// поля форм

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_form" integer, // код формы

"Title" character varying(1024), // название поля

"Id\_dict" integer, // код типа данных поля

"Order" integer, // поле сортировки

"Required" boolean NOT NULL, // флаг: обязательное поле к заполнению или нет

"Personal" boolean NOT NULL, // флаг: относится поле к персональным данным или нет

)

***TABLE "Forms"***

// формы

(

"Id" serial NOT NULL, // идентификатор таблицы

"Title" character varying(255) NOT NULL, название формы

"Order" integer, // поле сортировки

)

***TABLE "Formulas"***

// формулы для расчёта полей

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_field" integer, // код поля

"Title" character varying(1024) NOT NULL, // название формулы

"Predicate" character varying(4096) NOT NULL, // формула поля

)

***TABLE "GenReports"***

// сформированные отчёты

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_user" integer NOT NULL, // код пользователя, сгенерировавшего отчёт

"Filename" character varying(1024) NOT NULL, // имя файла

"Data" bytea NOT NULL, // содержимое файла

"Created" timestamp without time zone, // дата и время генерации отчёта

"Type" character varying(50) NOT NULL, // тип сформированного отчёта

"Signature" character varying(128) NOT NULL, // ЭЦП сформированного отчёта

)

***TABLE "GroupHouses"***

// таблица соответствия группы домов и домов

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_group" integer NOT NULL, // код группы домов

"Id\_house" integer, // код дома

)

***TABLE "GroupRepTemplates"***

// шаблоны генератора отчётов

(

"Id" serial NOT NULL, // идентификатор таблицы

template\_name character varying(100) NOT NULL, // название шаблона отчёта

)

***TABLE "Groups"***

// группы домов

(

"Id" serial NOT NULL, // идентификатор таблицы

"Title" character varying(255) NOT NULL, // имя группы

"Id\_user" integer, // код пользователя, который будет видетб эти группы домов

)

***TABLE "Houses"***

// таблица для вывода информации по дому на главной странице

(

"Id" serial NOT NULL, // идентификатор таблицы

id\_location integer, // код населенного пункта

id\_raion integer, // код района

id\_street integer, // код улицы

"number" integer, // номер дома

letter character varying(3), // литера дома

latitude real, // географическая широта, на которой расположен дом

longitude real, // географическая долгота, на которой расположен дом

id\_tdoma integer, // код типа дома

seriya character varying(50), // серия, тип проекта

date\_exp timestamp without time zone, // дата ввода в эксплуатацию

etaz\_max integer, // количество этажей, наибольшее

kolmunkv integer, // количество муниципальных квартир

ploo integer, // всего общая площадь квартир

kolzp integer, // зарегистрированно постоянно

id\_fupr integer, // код формы управления

id\_oorg integer, // код управляющей / обслуживающей организации

)

***TABLE "ImpFiles"***

// Импортированные файлы

(

"Id" serial NOT NULL, // идентификатор таблицы

"Filename" character varying(1024) NOT NULL, // имя файла

"Created" timestamp without time zone NOT NULL, // дата и время импорта файла

"Id\_user" integer NOT NULL, // код пользователя, импортирующего файл

)

***TABLE "Keys"***

// публичные ключи пользователей для подписания данных

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_User" integer, // код пользователя-владельца ключа

"KeyValue" character varying(4096), // содержимое ключа

"Date\_Start" timestamp without time zone, // дата начала действия ключа

"Date\_Finish" timestamp without time zone, // дата окончания действия ключа

)

***TABLE "Messages"***

// сообщения по домам

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_house" integer, // код дома

"Text" character varying(4096), // текст сообщения

"Id\_user" integer, // код пользователя-автора сообщения

"Created" timestamp without time zone, // дата и время создания сообщения

)

***TABLE "PrivateKeys"***

// приватные ключи для подписания данных

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_user" integer NOT NULL, // код пользователя

"KeyValue" character varying(4096) NOT NULL, // содержимое ключа

)

***TABLE "ReportParams"***

// настройки шаблона отчёта из генератора отчётов

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_group" integer, // код шаблона отчёта

"Id\_form" integer, // код формы

"Id\_field" integer, // код поля

"Chosen" boolean, // флаг: выводить поле в отчёте или нет

"Id\_category" integer, // код соответствия категории полю

)

***TABLE "ReportTemplates"***

// шаблоны типовых отчётов (групповые + для одного дома)

(

"Id" integer NOT NULL, // идентификатор таблицы

"Filename" character varying NOT NULL, // название файла шаблона отчёта

"Data" bytea NOT NULL, // содержимое файла шаблона отчёта

"Reportname" character varying(255), // название отчёта

"Reporttype" integer, // идентификатор типа отчёта:

// 1 – для одного дома, 2 – для группы домов

)

***TABLE "Results"***

// результаты обновления данных о домах

(

"Id" serial NOT NULL, // идентификатор таблицы

"Created" time without time zone NOT NULL, // дата и время внесения изменения

"Id\_user" integer NOT NULL, // код пользователя – автора изменений

"Status" character varying(20) NOT NULL, // результат правки: Ок или Ошибка

"Error" character varying(1024), // текст ошибки

"Element" character varying(1024) NOT NULL, // обновляемое поле, формат данных:

// [Форма].[Родительский элемент (если есть)].[Поле]

"SourceType" character varying(20), // тип обновления данных:

// import – импорт из файла, manual - вручную

"Id\_source" integer, // идентификатор файла в случае, если SourceType = import

"House" character varying(1024) NOT NULL, // дом, информация о котором обновляется

)

***TABLE "Rights"***

// уровни доступа

(

"Id" serial NOT NULL, // идентификатор таблицы

"Title" character varying(255), // наименование уровня доступа

)

***TABLE "Roles"***

// роли пользователей

(

"Id" serial NOT NULL, // идентификатор таблицы

"Title" character varying(255) NOT NULL, // название роли

)

***TABLE "RolesForms"***

// таблица соответствия форм и категорий по уровням доступа из таблицы ролей

(

"Id" integer NOT NULL, // идентификатор таблицы

"Id\_Role" integer, // код роли

"Id\_Form" integer, // код формы

"Id\_Cat" integer, // код категории

"Id\_Right" integer, // код уровня доступа

)

***TABLE "RolesHouses"***

// таблица соответствия домов по уровням доступа из таблицы ролей

(

"Id" integer NOT NULL, // идентификатор таблицы

"Id\_Role" integer NOT NULL, // код роли

"Id\_House" integer, // код дома

"Id\_Right" integer, // код уровня доступа

)

***TABLE "RuleOperations"***

// логические операторы сравнения для генерации правил

(

"Id" serial NOT NULL, // идентификатор таблицы

"Operation" character varying(10) NOT NULL, // логический оператор сравнения

)

***TABLE "Rules"***

// правила полей

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_operation" integer NOT NULL, // код логического оператора

"Title" character varying(1024), // название правила для общего понимания правила

"Predicate" character varying(1024), // операнд для сравнения

"Id\_field" integer, // код поля

)

***TABLE "Sessions"***

// сессии пользователей

(

"Id" serial NOT NULL, // идентификатор таблицы

"Token" character varying(100) NOT NULL, // токен, идентифицирующий пользователя

"CreatedTime" timestamp without time zone, // время начала сессии

"UserId" integer NOT NULL, // код пользователя – хозяина сессии

)

***TABLE "Settings"***

// настройки подсказки Яндекс-карты

(

"Id" serial NOT NULL, // идентификатор таблицы

"BaloonTemplate" text, // текст подсказки Яндекс-карты

)

***TABLE "UnreadMessages"***

// непрочитанные сообщения пользователя по домам

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_user" integer, // код пользователя

"Id\_message" integer, // код сообщения

)

***TABLE "Users"***

(

"Id" integer NOT NULL DEFAULT, // идентификатор таблицы

"Username" character varying(255) NOT NULL, // ФИО пользователя

"Password" character varying(100) NOT NULL, // пароль пользователя

"Active" boolean, // флаг активности пользователя: архивный или нет

"Login" character varying(255), // логин пользователя

"Id\_Role" integer, // код роли

)

***TABLE "Values"***

// значения полей по домам и родительским элементам

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_house" integer, // код дома

"Id\_form" integer NOT NULL, // код формы

"Id\_field" integer NOT NULL, // код поля

"Value" character varying(4096), // значения поля

"Id\_parent" integer, // код родительского элемента, если есть, в противном случае - 0

"Warning" character varying(1024), // текст предупреждения после валидации правил

)

# Описание таблиц базы данных pdn

***TABLE "Values"***

// значения по домам и родительским элементам, структура и смысл аналогичны таблице Values из бд zhilfond, только в этой таблице – персональные данные системы, требующие большей защищенности и ограничений по доступу к информации

(

"Id" serial NOT NULL, // идентификатор таблицы

"Id\_house" integer NOT NULL, // код дома

"Id\_form" integer NOT NULL, // код формы

"Id\_field" integer NOT NULL, // код поля

"Value" character varying(4096), // значения поля

"Id\_parent" integer, // код родительского элемента, если есть, в противном случае - 0

"Warning" character varying(1024), // текст предупреждения после валидации правил

)

# Описание прочих объектов базы данных zhilfond

Кроме таблиц, в базе данных zhilfond существуют представления, цель которых – удобный доступ к часто запрашиваемым данным. Детально можно с ними ознакомиться, посмотрев запросы для создания представлений в базе данных. Так, в системе реализованы представления:

Audit\_v – для просмотра человеко-понятной информации по аудиту;

DictValues\_v – для просмотра человеко-понятной информации по справочникам и их значениям;

Houses\_v – для просмотра человеко-понятной информации по домам;

ValuesAll\_v – для просмотра значений полей с дополнением значений персональных данных;

ValuesPdn\_v – для выборки значений персональных данных;

Values\_v – для просмотра человеко-понятной информации с учётом содержимого справочников.

Для организации взаимодействия с базой данных персональных данных pdn в базе данных zhilfond реализованы триггерные функции insert\_check\_pdn и update\_check\_pdn для вставки и обновления полей соответственно.