API DB Wrapper.

API создания схемы БД.

Две основные части:

1. Регистрация сущностей
2. Генерация объектов

Сущности.

Описание сущностей базы и связей между ними.

API objects

## Scheme

|  |  |
| --- | --- |
| Parameters | |
| Table: | xxweb\_db\_schemes\_t |
| Object: | xxweb\_db\_scheme\_typ |
| Collection: | - |
| Object view | xxweb\_db\_schemes\_v |

### Description scheme

|  |  |  |
| --- | --- | --- |
| Id |  |  |
| Name |  |  |
| Dev\_code |  |  |
| Owner |  |  |
| Version |  |  |
| Entities | Collection entity | xxweb\_db\_entities\_typ |
| Db\_objects | Collection object |  |
| Creation\_date |  |  |
| Last\_update\_date |  |  |
| Is\_deleted |  |  |
| Field\_tmp | Не хранимое поле. Используется только в процессе обработки |  |
| Indexes\_tmp | Не хранимая коллекция индесков |  |

## Entity

|  |  |
| --- | --- |
| Parameters | |
| Table: | xxweb\_db\_entities\_t |
| Object: | xxweb\_db\_entity\_typ |
| Collection: | xxweb\_db\_entities\_typ |
| Object view | xxweb\_db\_entities\_v |

### Description entity

|  |  |  |
| --- | --- | --- |
| id |  |  |
| Scheme\_id |  |  |
| Idx | Номер сущности в схеме (принципиален при создании объектов) |  |
| Name |  |  |
| EntryName |  |  |
| Version |  |  |
| Fields | Collection fields |  |
| Keys |  |  |
| Indexes |  |  |
| Contents | Коллекция значений (для создания простого справочника) |  |
| Creation\_date |  |  |
| Last\_update\_date |  |  |
| Is\_deleted |  |  |

## Objects (объекты БД)

|  |  |
| --- | --- |
| Parameters | |
| Table: | xxweb\_db\_objects\_t |
| Object: | xxweb\_db\_object\_typ |
| Collection: | xxweb\_db\_objects\_typ |
| Object view | xxweb\_db\_objects\_v |

### Description object

|  |  |  |
| --- | --- | --- |
| id |  |  |
| Scheme\_id |  |  |
| Idx | Номер сущности в схеме (принципиален при создании объектов) |  |
| Owner |  |  |
| Type |  |  |
| Object Type |  |  |
| Name |  |  |
| Ddl |  |  |
| Status |  |  |

## Indexes

|  |  |
| --- | --- |
| Parameters | |
| Table: | xxweb\_db\_indexes\_t |
| Object: | xxweb\_db\_index\_typ |
| Collection: | xxweb\_db\_indexes\_typ |
| Object view | xxweb\_db\_indexes\_v |

### Description index

|  |  |  |
| --- | --- | --- |
| id |  |  |
| entity\_id |  |  |
| Idx | Номер сущности в схеме (принципиален при создании объектов) |  |
| Version |  |  |
| Name |  |  |
| Fields |  |  |
| Type |  |  |
| Is\_deleted |  |  |

Fields

|  |  |
| --- | --- |
| Parameters | |
| Table: | xxweb\_db\_fields\_t |
| Object: | xxweb\_db\_field\_typ |
| Collection: | xxweb\_db\_fields\_typ |
| Object view | xxweb\_db\_fields\_v |

### Description field

|  |  |  |
| --- | --- | --- |
| id |  |  |
| entity\_id |  |  |
| Idx | Номер сущности в схеме (принципиален при создании объектов) |  |
| Version |  |  |
| Name |  |  |
| Type |  |  |
| Length |  |  |
| Accuracy | Точность числа |  |
| Is\_Unique | Y\N |  |
| Is\_Sequence | Y\N |  |
| Is\_PK | Y\N |  |
| Is\_FK | Y\N |  |
| Is\_Bool | Y\N |  |
| Is\_Indexed | Y\N |  |
| Default\_value |  |  |
| Is\_Delete | Y\N |  |

## Relationships

|  |  |
| --- | --- |
| Parameters | |
| Table: | xxweb\_db\_relationships\_t |
| Object: | xxweb\_db\_relationship\_typ |
| Collection: | xxweb\_db\_relationships\_typ |
| Object view | xxweb\_db\_relationships\_v |

### Description relationship

|  |  |  |
| --- | --- | --- |
| Id |  |  |
| Entity\_id |  |  |
| Idx |  |  |
| Type |  |  |
| source\_field\_id |  |  |
| target\_entity\_id |  |  |
| target\_field\_id |  |  |
| collect\_name |  |  |
| Updated |  |  |
| Deleted |  |  |
| is\_deleted |  |  |

Генерация объектов

Создание объектов, по описанным связям.

Методы работы с БД

Метод **load**

Принимает xml, парсит его, возвращает готовый объект.

Структура XML:

<content>

<Object1>

<Attribute1>

…

<AttributeN>

</Object1>

</content>

Object: <ObjectName>content</ObjectName>. ObjectName - имя записи (entry\_name). content – содержимое объекта (поля). Для идентификации объекта, в его содержимом должен быть тег с именем поля первичного ключа.

Attribute – атрибут объекта. Может представлять из себя поле, объект, массив объектов.

Если Attribute отсутствует – берется значение из базы. Если атрибут есть (даже пустой) – берется его значение.

Attribute поле: <FieldName>Value</FieldName>. FieldName – название поля, так, как задано при описании схемы.

Attribute объект: аналогично Object.

Attribute массив объектов: <CollectionName><ObjectName>content</ObjectName></CollectionName>.