

????????? ?????????? ?????

IEntity

При реализации маппинга простых типов данных не требуется каких либо ухищрений. Однако, если в сущности есть свойства типа ссылки на справочник или документ, то необходимо реализовать дополнительную логику.

Пример для сущности Контакт (исходный код урезан для простоты понимания):

```
/// <inheritdoc />
public class ContactDataItem : BaseDataItem
{
    [/// <inheritdoc />
        public override string Path => "app/_clients/_contacts";

        /// <summary>
        /// </summary>
        [JsonProperty(PropertyName = "__name", NullValueHandling = NullValueHandling.Ignore)]
        public string Name { get; set; }

        ...

        /// <summary>
        /// </summary>
        [JsonProperty(PropertyName = "_companies", NullValueHandling = NullValueHandling.Ignore)]
        public List<Guid> Companies { get; set; }

        ...
    }
}
```

Данная реализация позволяет подготовить пакет передачи в ELMA365 в нужном формате.

Лисенер для регистрации изменений элемента сущности (исходный код урезан для простоты понимания):

```
public static Pair<IBaseDataItem, IEntity> SyncContact(IContact item, bool syncLink = false)
{
    [var data = new ContactDataItem();
```

```
if (item == null)
    return new Pair<IBaseDataItem, IEntity>(data, item);

...

E365DataItemManager.Instance.PushItem(typeof(ContactDataItem), item.Uid, data, item,
SR.T("Контакт: {0}", item.Name));

if (!syncLink)
    return new Pair<IBaseDataItem, IEntity>(data, item);

var link = new LinkDataItem(data);

link.Links.Add(new LinkImpl(ContractorListener.SyncContractor(item.Contractor),
    BaseDataItem.GetJsonPropertyName<ContactDataItem>(x => x.Companies)));

E365DataItemManager.Instance.PushItem(typeof(LinkDataItem), link.Uid, link, item, SR.T("Связи
в контакте: {0}", data.Name));

return new Pair<IBaseDataItem, IEntity>(data, item);
}
```

Версия #2

Вербовик Максим Арнольдович создал 2023-02-19 11:35:01 MSK

Вербовик Максим Арнольдович обновил 2023-04-25 14:23:08 MSK