EngDictionary

Solution-а ще има следните проекти:

1. EngDictionary.Data – проект, който ще съдържа всички entities (обекти - думи)
2. EngDictionary.Data.Tests – тестов проект
3. EngDictionary.Engine – проект, който ще съдържа всички операции с думите (създаване, редактиране, изтриване на думи, справки на думи, четене и запис на думи в XML файл.)
4. EngDictionary.Engine.Test – тестов проект
5. EngDictionary.Client – проект, който ще съдържа User Interface, който ще работи с данните на речника.
6. EngDictionary.Client.Tests – тестов проект

EngDictionary.Data

Задание:

Проекта да съдържа следните Entities:

* **Verb** (Глагол)
  + **Id** of type Guid (read only – означава, има само get, няма set)
  + **Text** of type string
  + **Translation** of type string
  + **IsRegular** of type bool
  + **PastSimple** of type string
  + **PastParticiple** of type string
* **Subjective** (Съществително)
  + **Id** of type Guid (read only – означава, има само get, няма set)
  + **Text** of type string
  + **Translation** of type string
  + **Plural** of type string
* **Adjective** (Прилагателно)
  + **Id** of type guid (read only – означава, има само get, няма set)
  + **Text** of type string
  + **Translation** of type string
  + **Comparative** of type string
  + **Superlative** of type string
* **Word** (Обща дума – наречие, съюзи, предлози, т.н.)
  + **Id** of type Guid (read only – означава, има само get, няма set)
  + **Text** of type string
  + **Meaning** of type string

Примери:

Verb

|  |  |  |
| --- | --- | --- |
| Property Name | Type | Example |
| Id | Guid | EA129E5F-F2DF-4698-AEC1-97D3A8EDA372 |
| Text | string | Work |
| Translation | string | Работя |
| IsRegular | bool | No (0) |
| PastSimple | string | Worked |
| PastParticiple | string | Worked |

|  |  |  |
| --- | --- | --- |
| Property Name | Type | Example |
| Id | Guid | 23677258-7242-4E8E-9F86-A6310A0B2215 |
| Text | string | Swim |
| Translation | string | Плувам |
| IsRegular | bool | Yes (1) |
| PastSimple | string | Swam |
| PastParticiple | string | Swum |

Subject

|  |  |  |
| --- | --- | --- |
| Property Name | Type | Example |
| Id | Guid | 4244B4DB-3443-493E-B155-890EA70C344A |
| Text | string | Table |
| Translation | string | Маса |
| Plural | string | Tables |

Adjective

|  |  |  |
| --- | --- | --- |
| Property Name | Type | Example |
| Id | Guid | 5E251EBE-618C-47B4-8B92-F0607E339B5E |
| Text | string | Big |
| Translation | string | Голям |
| Comparative | String | Bigger |
| Superlative | string | Biggest |

Word

|  |  |  |
| --- | --- | --- |
| Property Name | Type | Example |
| Id | Guid | 95E3508C-40CC-4699-B663-1DDA24E24D87 |
| Text | string | For |
| Translation | string | За |

|  |  |  |
| --- | --- | --- |
| Property Name | Type | Example |
| Id | Guid | 80EF9A6A-87B4-44B3-859D-6170733DE82D |
| Text | string | Monday |
| Translation | string | Понеделник |

Да се направи архитектура на проекта.

* Да се направи interface, който да съдържа всички общи характеристики на всички видове думи. Interface-a да е с име IWord
* Да се направи абстрактен клас, който да съдържа всички общи черти на всички Entities. Класът да се казва EntityBase и са имплементира interface-a IWord. Да има 2 конструктора:

EntityBase() – без параметри, в който Id да се инициализира с произволна Guid стойност.

EntityBase(Guid id) – с параметър Guid, с който да се инициализира пропъртито Id. За генериране на произволен Guid използвай Guid.NewGuid();

* За всеки вид дума да се направи интерфейс и клас. Интерфейса на думата да наследява интерфейса IWord, класа на думата да наследява и двете – абстрактния клас EntityBase и интерфейса, който е направен за тази дума.
* Всички интерфейси да са public. Да се намират под проекта.
* Всички класове, включително и абстрактния да се намират в папка Implementation и да са internal.

Tips:

Създаване на пропърти в клас:

Това е традиционния начин за създаване на пропърти, като се изпозлва private member:

private string \_name;

public string Name

{

get { return \_name; }

set { \_name = value;

}

По-новия вариант за създаване на пропърти е следния:

public string Name { get; set; }

Това е същото като горния начин, но много по-опростено. Използва се в повечето случаи. Вътре и извън класа ползваш директно Name. Направи пропъртитата на класовете по този начин.