=ЛЕВСИМВ(C2;НАЙТИ(" ";C2;2)-1)\*ЕСЛИ(ПРАВСИМВ(C2;4)="час.";3600;60)

Из 1 таб

Для переноса данных используем SQL-запрос:

INSERT INTO [Client] (FirstName, LastName, Patronymic,Birthday, RegistrationDate, Email, Phone, GenderCode)

SELECT \* FROM client\_a\_import

из 3 таб

INSERT INTO ClientService (ClientID, ServiceID, StartTime)

SELECT c.ID, s.ID, sc.[begin]

FROM serviceclient\_a\_import sc, [Client] c, [Service] s

WHERE sc.[name]=s.Title AND sc.client=c.FirstName

**Стили**

Стили для всех компонентов приложения можно задать в файле App.xaml

<Application.Resources>

<Style TargetType="Button">

<Setter Property="Background" Value="AliceBlue"/>

<Setter Property="FontStyle" Value="Italic"/>

</Style>

<Style TargetType="Control">

<Setter Property="FontFamily" Value="Times New Roman"/>

</Style>

</Application.Resources>

<Application.Resources>

<Style TargetType="Button">

<Setter Property="Background" Value="#ff9c1a"/>

</Style>

</Application.Resources>

FontFamily="Arial Black"

Title="Услуги авто сервиса"

MinHeight="100" MinWidth="300"

Height="450" Width="800">

**Разработка модульных тестов**

Собственно это мы уже проходили, но повторим на этом примере еще раз.

**Создание проекта для модульных тестов**

В контекстном меню **РЕШЕНИЯ** выбираете *Добавить -> Создать проект*

При создании проекта можете воспользоваться фильтрами "языки", "платформа" и "типы проектов". В итоге надо выбрать "Проект модульного теста (.NET Framework)"

Правила формирования имени тестового проекта были в лекции, у меня получилось **CompanyCoreLib.Tests**. Расположение не трогаем, по-умолчанию тестовый проект сохранится в том же решении (рядом с основным проектом)

**Связь с основным проектом**

Дальше нужно добавить связь с основным проектом:

В тестовом проекте находим пункт "ссылки" (раньше назывался "зависимости") и в контекстном меню "добавить ссылку"

Выбрать основной проект (их может быть несколько), у нас он называется **CompanyCoreLib**.

Проверить, добавлен ли проект в зависимости, можно раскрыв "ссылки"

**Написание тестов**

Реализация теста, проверяющего результат работы метода *PopularMonths*:

namespace CompanyCoreLib.Tests

{

[TestClass]

public class AnalyticsTest

{

static Analytics AnalyticsClass = null;

[ClassInitialize]

static public void Init(TestContext tc)

{

AnalyticsClass = new Analytics();

}

[TestMethod]

public void PopularMonths\_Await3ItemsWithSortByDate()

{

List<DateTime> dates = new List<DateTime>() {

new DateTime(2020, 12, 17),

new DateTime(2020, 12, 15),

new DateTime(2020, 11, 17),

new DateTime(2020, 10, 1) };

dates = AnalyticsClass.PopularMonths(dates);

// должно вернуть 3 записи

Assert.AreEqual(dates.Count, 3);

// первой должен быть декабрь

Assert.AreEqual(dates[0], new DateTime(2020, 12, 1));

}

}

}