**Программа для учета компьютеров, оргтехникии, оборудования.**

*Основные функции*:

* Ведение базы компьютеров и оргтехники
* Учет установленного на них программного обеспечения
* Фиксация ремонтов и апгрейдов
* Закрепление объектов за ответственными сотрудниками
* Контроль гарантийных сроков
* Ведение базы поставщиков и прочих партнеров

(Составляющие системы, можно считать, как отдельные проекты, которые должны жить совей жизнью)

Составляющие системы:

* Вкладка «**Устройства**»
* Вкладка «**Отделы**»
* Вкладка «**Организации**»
* Вкладка «**Сотрудники**»

(Это описание основных разделов программы. Необходимо грамотно создать структуру всех классов, интерфейсов. Грамотно продумать наследование и качественное использование интерфейсов, т.к. это залог успешной программы)

Таблицы по умолчанию:

**Устройства:**

1. Главная таблица со списком компьютеров, оргтехники и других устройств и оборудования.
2. Приведена информация о сроках гарантии, ответственных лицах, стоимости, технические характеристики и пр.

Список функций:

1. Добавление техники
2. Удаление техники
3. Редактирование техники
4. Поиск техники по наименованию

*Должна быть связь между на установленное ПО*

**Программное обеспечение:**

1. Таблица с информацией об установленном ПО: стоимость, дата установки и т.д. Находится по умолчанию на вкладке «Устройства». Подчиненная таблица к таблице «Устройства». Т.К. информация о ПО своя для каждого устройства.

Список функций

1. Список ПО (добавление, редактирование, удаление)

**Организации:**

1. Таблица со списком и контактной информацией организаций-поставщиков, партнеров, поставщиков, покупателей и потенциальных клиентов.

Список функций:

1. Добавление / Удаление / Редактирование контактной информации
2. Получение всего список контактов
3. Поиск в списке контактов

**Ремонты:**

1. Таблица с информацией о произведенных ремонтах: стоимость, описание и т.д. Находится по умолчанию на вкладке «Устройства».
2. Подчиненная таблица к таблице «Устройства». Т.К. информация о произведенных ремонтах своя для каждого устройства.

Список функций:

1. Создание / Редактирование / Удаление ремонта
2. Получения полного списка ремонтов
3. Получение полного списка ремонтов на определенной машине

**Отделы:**

1. Таблица со списком отделов в организации, и с контактной информацией по отделам. Каждое устройство из главной таблицы прикреплено к одному из отделов.

Список функций:

1. Добавление / Удаление / Редактирование информации об отделе
2. Получение всего список отделов

**Сотрудники:**

1. Таблица с информацией о сотрудниках организации, в частности, в каком отделе кто работает. Каждое устройство из главной таблицы прикреплено к одному из сотрудников.

Список функций:

1. Добавление / Удаление / Редактирование информации о сотруднике
2. Получение всего список сотрудников

*Лучше сразу грамотно распределить роли между сотрудниками Вашей организации: кто за хранение и обновление какого рода данных будет отвечать. Исходя из этого распределения, кто-то будет заносить в базу информацию о новом, поступившем к Вам оборудовании, кто-то о клиентах, поставщиках и прочих деловых партнерах. Кто-то контролировать информацию о сотрудниках, а кто-то печатать отчеты и исходящие документы. Возможно, все эти действия будет производить и один человек.*

**Общие требования к проекту:**

1. **В качестве БД использовать XML файлы.**
2. **Каждый модуль должен быть отдельной библиотекой, ничем не связанной между собой.**
3. **В случае создания машины/останова/пользователя и т.д. должно происходить редактирование XML файла, в случае отсутствия данного файла создание.**
4. **В проекте должны быть использованы следующий функционал ООП:**
   1. **Наследование**
   2. **Интерфейсы**
   3. **Свойства**
   4. **Возможно Делегаты**
   5. **Коллекции**
   6. **Выбрать где использовать структуры где классы**
   7. **Обработка исключений**
5. В каком случае вы обязаны объявить класс абстрактным?
6. Назовите класс .NET, от которого наследуются все классы?
7. Почему нельзя указать модификатор видимости для методов интерфейса?
8. Можно ли наследовать от нескольких интерфейсов?
9. Какое преимущество использования класса System.Text.StringBuilder перед System.String?
10. Чем отличается event от delegate?
11. Поддерживает ли C# множественное наследование?
12. Кому доступны переменные с модификатором protected на уровне класса?
13. Наследуются ли переменные с модификатором private?
14. Опишите модификатор “protected internal”