# Задание

Разработать диаграммы сущностных, граничных и управляющих классов для ИС из своего варианта (задача №2).

# Настройка WhiteStarUML

## Включить отображение стереотипов

В меню Model – Profiles..., добавить профиль “UML Standard Profile”.

## Указать стереотип класса или связи

Выделить класс/связь, в окне свойств (Properties, справа внизу) в строчке Stereotype нажать на многоточие. Если в открывшемся окне список доступных стереотипов пуст, то включить отображение стереотипов (см. выше).

## Указать кратность связи

Выделить связь, в окне свойств (Properties, справа внизу) в строках End1.Multiplicity и End2.Multiplicity выбрать из выпадающего списка. Там же в строках End1.Participant и End2. Participant можно увидеть с каким классом соединен конец связи.

## Включить перенос слов в названии класса

Выделить нужный класс(ы), правый клик – Format – Word Wrap Name.

## Скрыть методы у классов-сущностей

Выделить нужный класс(ы), правый клик – Format – Suppress Operations.

# Общие рекомендации по созданию диаграмм классов

## «entities» (Сущностные классы)

1. Каждый класс соответствует таблице БД, названия даются в виде существительных в единственном числе.
2. Атрибуты классов соответствуют полям таблиц.
3. Методы классов не используются.
4. Содержимое таблиц необходимо согласовать с заказчиком и внести в требования к системе
5. Таблицы должны содержать всю необходимую информацию для выполнения всех вариантов использования. Для **каждого варианта использования** необходимо ответить на 2 вопроса: «Какая нужна информация для его выполнения?» и «Где эта информация хранится?». Если ответ – «Нигде», то добавить нужные классы или атрибуты. И наоборот – если есть сущностный класс, то проверить, чтобы для него был вариант использования.
6. Поля, необходимые лишь для связи таблиц, а также кодовые поля, в классах не указываются.
7. На диаграмме могут присутствовать связи-ассоциации (ненаправленные), а также агрегация (aggregation) и композиция (composition).
8. На всех связях, кроме обобщения (generalization), указывается кратность.
9. Связи можно называть глаголами, но необязательно.

## «boundaries» (Граничные классы)

1. Классы соответствуют оконным формам или страницам сайта, с которыми будет взаимодействовать пользователь.
2. Граничные классы тесно связаны с диаграммой вариантов использования. **Каждый вариант использования** должен соответствовать какому-то граничному классу. Может быть один граничный класс для нескольких вариантов использования.
3. **Каждый сущностный класс** должен иметь соответствующий граничный класс для заполнения данных. Исключение – классы, автоматически заполняющиеся системой, если таковые имеются.
4. Основные действия, которые можно выполнить на форме, указываются в виде методов.
5. Основные поля, которые можно заполнить на форме, указываются в виде атрибутов.
6. На диаграмме в основном используются связи – направленные ассоциации. Направление показывает возможность перехода с одной формы на другую.
7. Могут присутствовать связи обобщения, если используется наследование форм, или связи агрегации и композиции, если используются стандартные шаблоны, которые включаются в несколько форм/страниц сайта.
8. Если с данной формы можно перейти на другую форму, то должен быть метод, который это реализует, либо на связи подписывается действие, которое должен выполнить пользователь (нажать на кнопку, выбрать пункт меню и т.п.).

## «controls» (Управляющие классы)

1. На диаграмме указываются различные «менеджеры», которые нужны для связи между граничными и сущностными классами.
2. Обычно здесь показываются «Менеджер СУБД», «Менеджер клиентского приложения», «Менеджер сайта», «Менеджер серверного приложения», «Менеджер HTTP-запросов» и т.п.
3. «Менеджер СУБД» обязательно содержит методы «добавить», «удалить», «изменить», «выполнить запрос».
4. У клиентских менеджеров методы можно не указывать.
5. На диаграмме присутствую связи зависимости (dependence) со стереотипом «call» (вызов).