Основы теории систем

**Отчет по лабораторной №3**

«Формирование онтологической модели с помощью Protege и Neo4g»

Вариант 15

Выполнил:

Студент1 курса   
Группы ИВТИИбд-12  
Мироненков Константин

Ульяновск,

2025

Задание:

1.Сформировать онтологическую модель структуры логической цепочки «от заводов до магазинов» в инструментах Protege и Neo4j

2.Провести сравнение Protege и Neo4j

3.Сделать заключение о удобстве инструментов

Разработка онтологической модели:

Нужно разработать онтологическую модель структуры цепочки производства от завода до магазина.

Требования:

Минимум 3 класса

у каждого класса хотя бы 1 свойство

минимум 2 типа связи

минимум 2 экземпляра каждого класса

минимум 2 правила логического вывода

Классы:

Завод

Магазин

Товар  
Склад

Отношения:

Хранит

Продает

Производит  
Поставляется в  
Обслуживает  
Имеет получателя

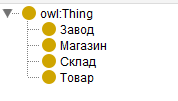
Аксиомы:

Если Завод имеет отношение «производит», то он имеет связь «имеет получателя»

Если Склад имеет отношение «хранит», то он имеет связь «обслуживает»

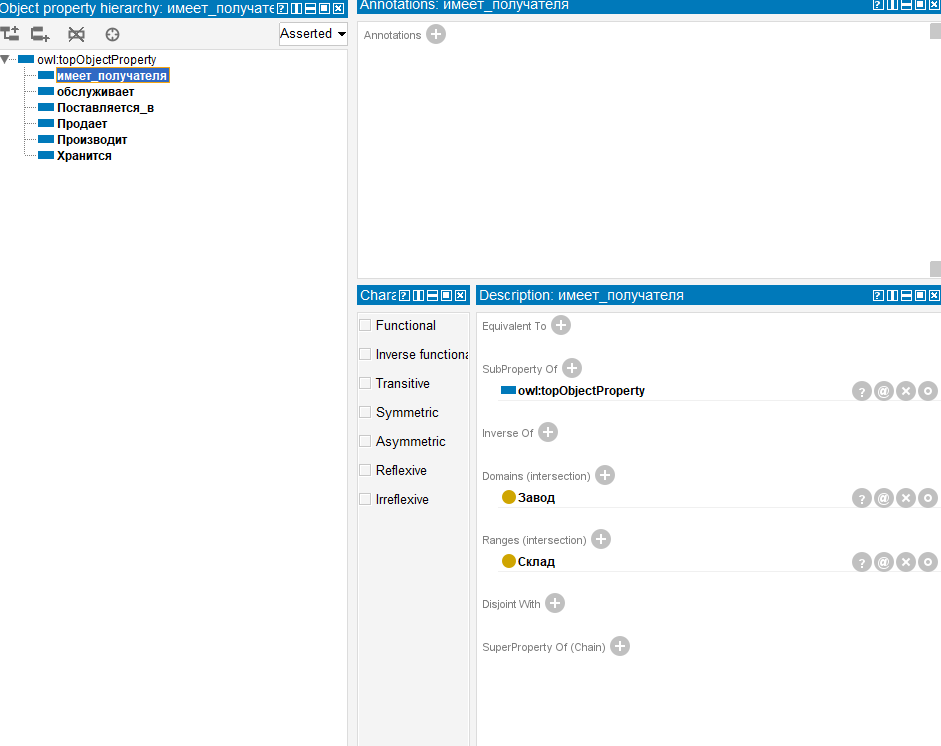
**Protege**

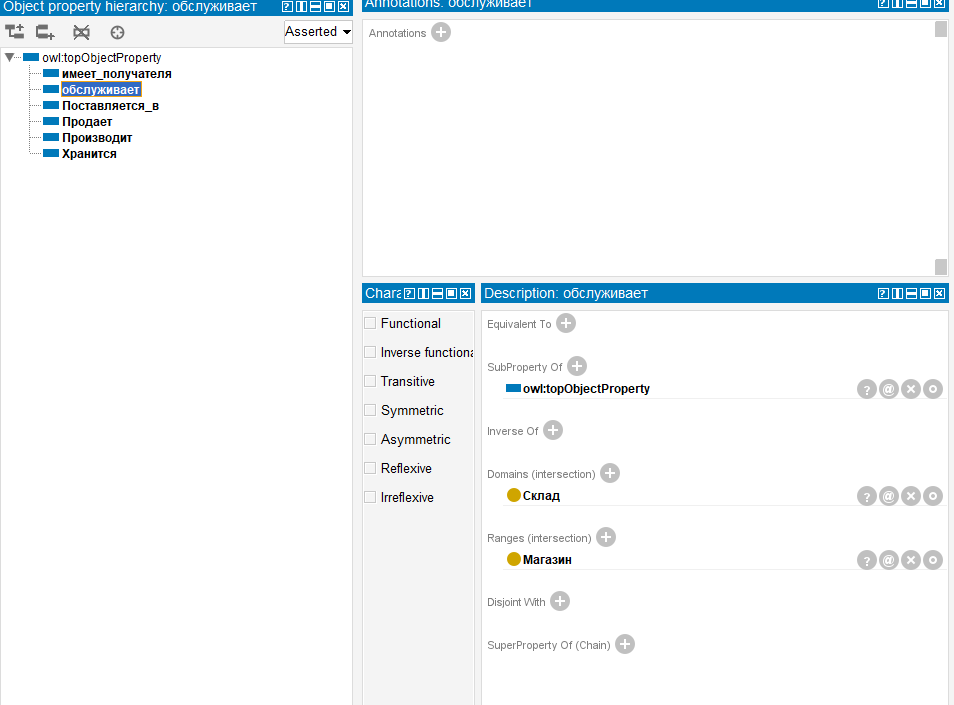
Создание классов:

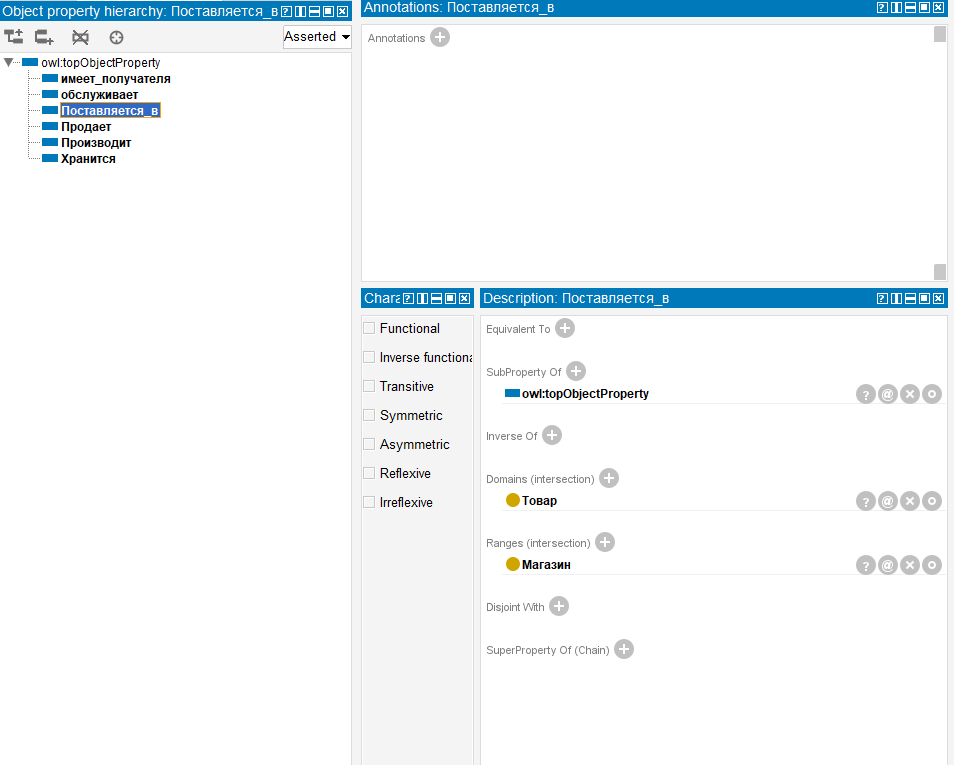


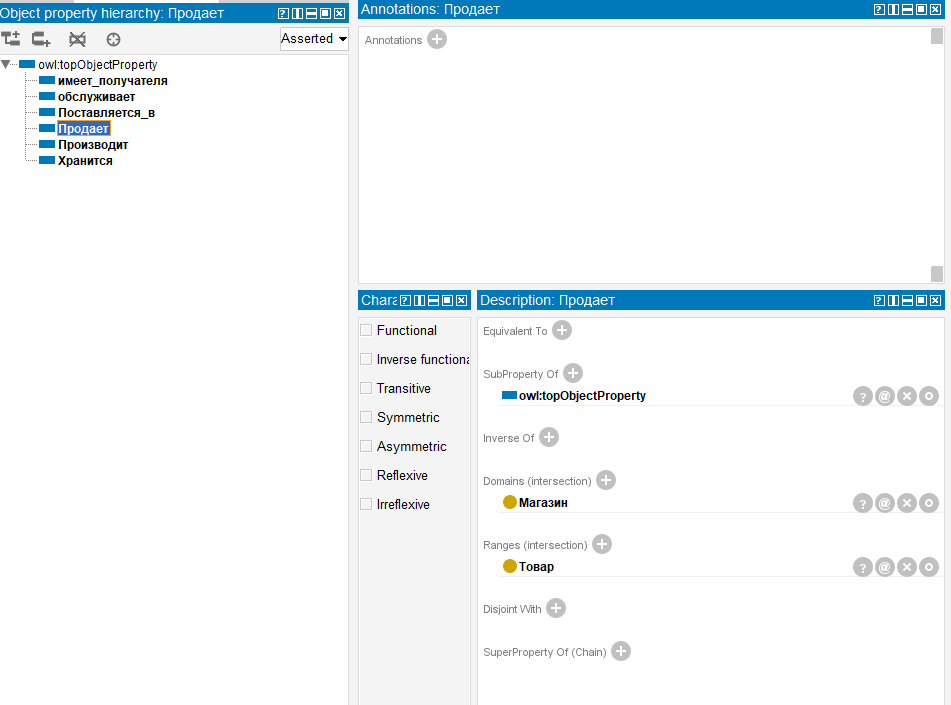
Создание связей:

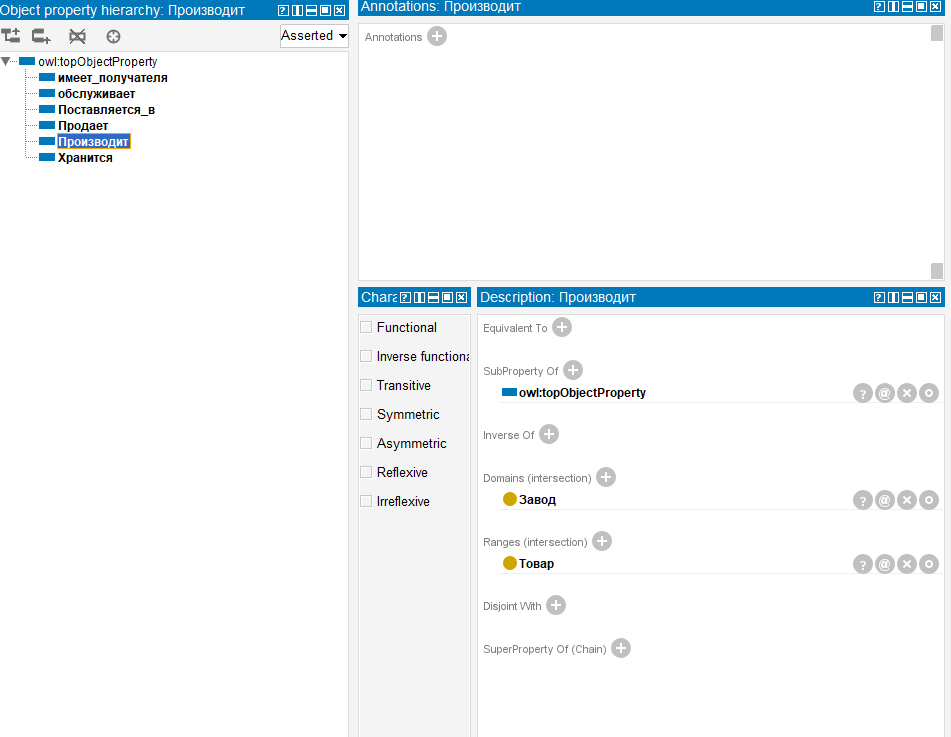
Для каждой связи указать Domen ( кто ) и Ranges(куда)

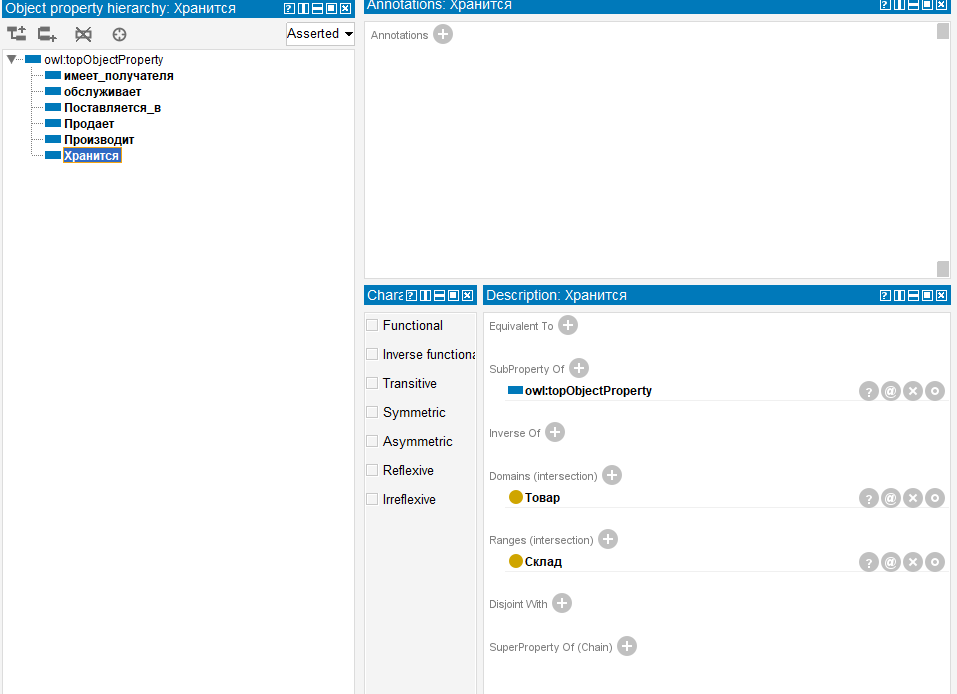










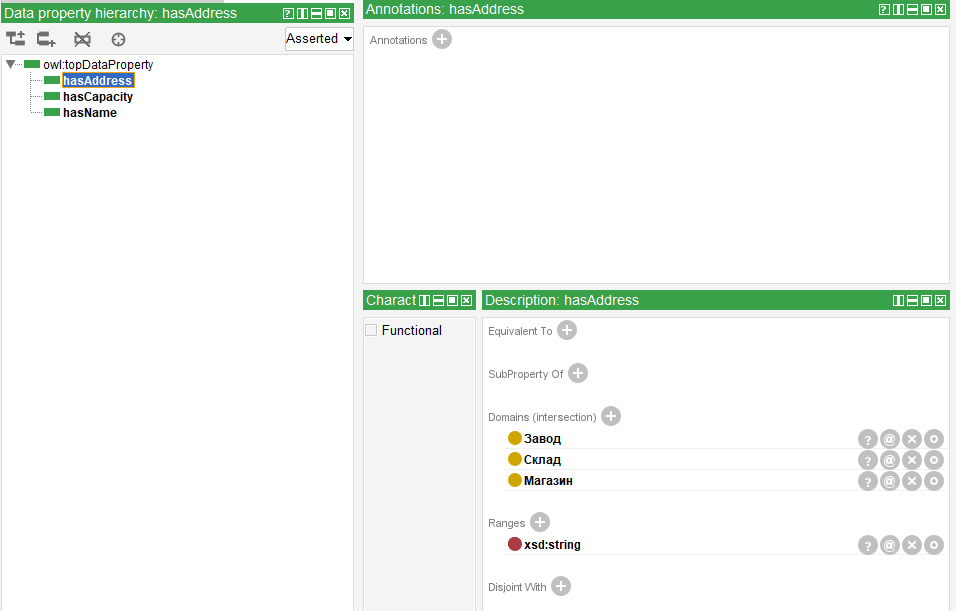


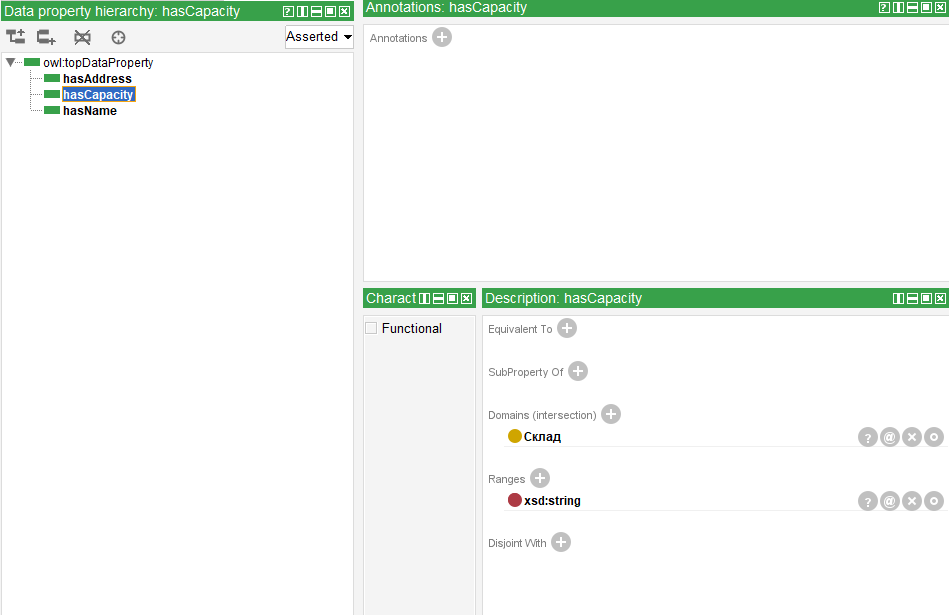
Создание свойств:

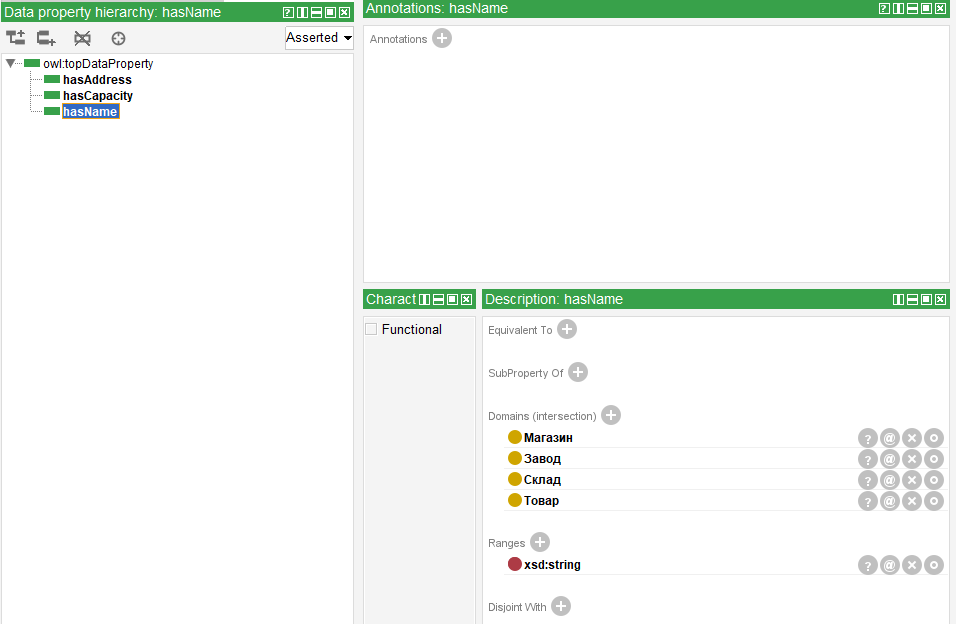
Создаем 3 свойства   
Адрес для заводов, складов и магазинов

Вместимость для складов

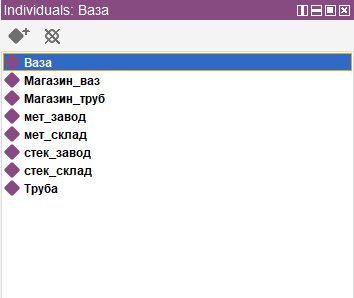
Название для всех класов







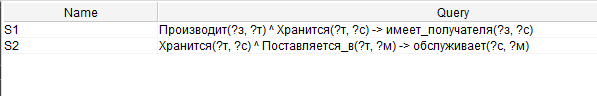
Создание экземпляров:

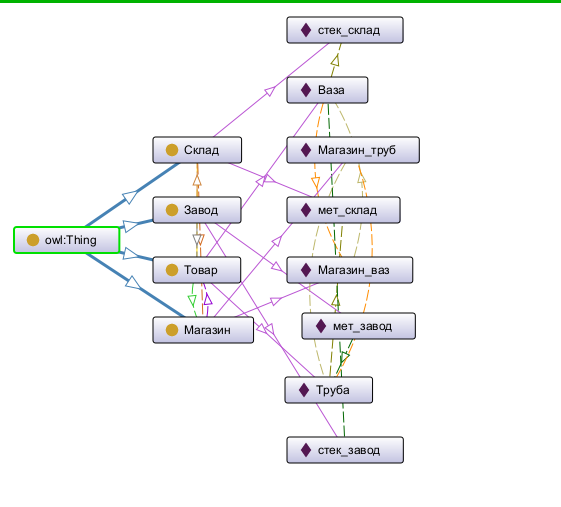


В каждом экземпляре указать тип элемента (Для вазы это Товар , Для мет\_склада это Склад ) и тд

Так же указать связи (Магазин\_ваз = продает-вазы) и тд

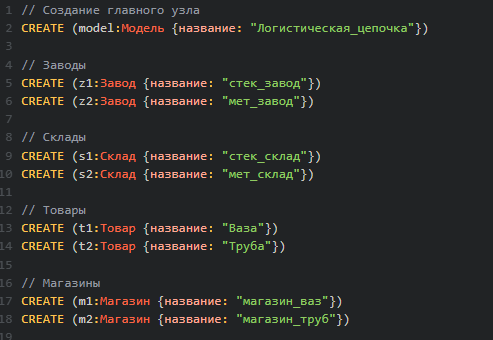
Правила вывода:



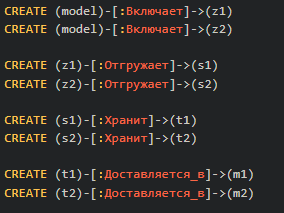
Итог:  


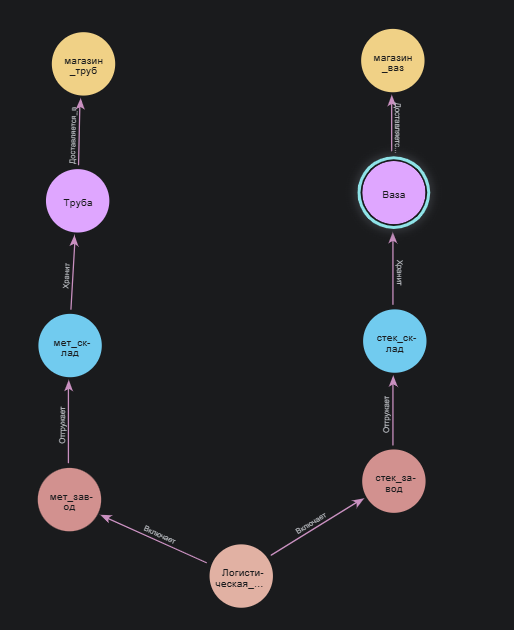
**Neo4j**

Создаем классы:



Создаем связи:



Итог:  


**Вывод**

1. **Protege**- Более структурированный и обширный инструмент с удобным интерфейсом, не нужны знания программирования

Выберите Protege если ваша цель :

разработать концептуальную модель (онтологию).

Вы цените наглядность и удобство графического интерфейса.

Вы не хотите погружаться в тонкости программирования.

1. **Neo4j** — оптимальное решение для хранения и обработки больших объемов графовых данных с динамической структурой, а также для простой интеграции в готовые приложения.

Выберите Neo4j  если ваша цель :

Вам нужно работать с огромными наборами данных, представленных в виде графа.

Ваши данные изменчивы и не укладываются в жесткую схему.

Вы планируете встраивать базу данных в приложение с помощью популярных языков программирования.