

Эссе:

1. Виды диаграмм UML
2. Какие связи между классами можно изобразить на UML диаграмме класса
3. Что такое стереотип

UML (Unified Modeling Language) - это стандартизированный язык моделирования, который используется для описания, проектирования и документирования программных систем.

### 1) Виды диаграмм UML:

**Структурные** диаграммы моделируют структуру системы, то есть ее статические компоненты и отношения между ними. Структурные диаграммы включают:

**1. Диаграмма классов (Class Diagram):** Это одна из наиболее распространенных структурных диаграмм. Она моделирует структуру классов в системе, то есть, какие классы существуют, какие у них атрибуты (поля) и методы, а также как они связаны друг с другом. Отношения между классами, такие как ассоциация, наследование и агрегация, могут быть также представлены на этой диаграмме.

**2. Диаграмма компонентов (Component Diagram):** Эта диаграмма моделирует физическую структуру системы и компоненты, из которых она состоит. Компоненты могут представляться как библиотеки, модули или сервисы, и на диаграмме отображаются их взаимосвязи и зависимости.

**3. Диаграмма объектов (Object Diagram):** Данная диаграмма позволяет показать конкретные экземпляры объектов и их отношения в определенный момент времени. Она может быть полезной для иллюстрации состояния системы в конкретные моменты времени.

**4. Диаграмма развертывания (Deployment Diagram):** Эта диаграмма моделирует развертывание компонентов системы на аппаратном оборудовании и сетях. Она показывает, на каких серверах и устройствах выполняются компоненты системы и как они связаны с сетями.

**Поведенческие диаграммы** моделируют динамическое поведение системы, то есть как система взаимодействует с объектами и какие процессы в ней происходят. Поведенческие диаграммы включают:

**1. Диаграмма вариаций (Use Case Diagram):** Эта диаграмма описывает функциональность системы с точки зрения ее пользователей (актеров) и сценариев использования. Она помогает понять, какие действия пользователей можно моделировать в системе.

**2. Диаграмма последовательности (Sequence Diagram):** Позволяет моделировать взаимодействие между объектами или компонентами системы во времени. На диаграмме отображается последовательность вызовов методов и передача сообщений между объектами.

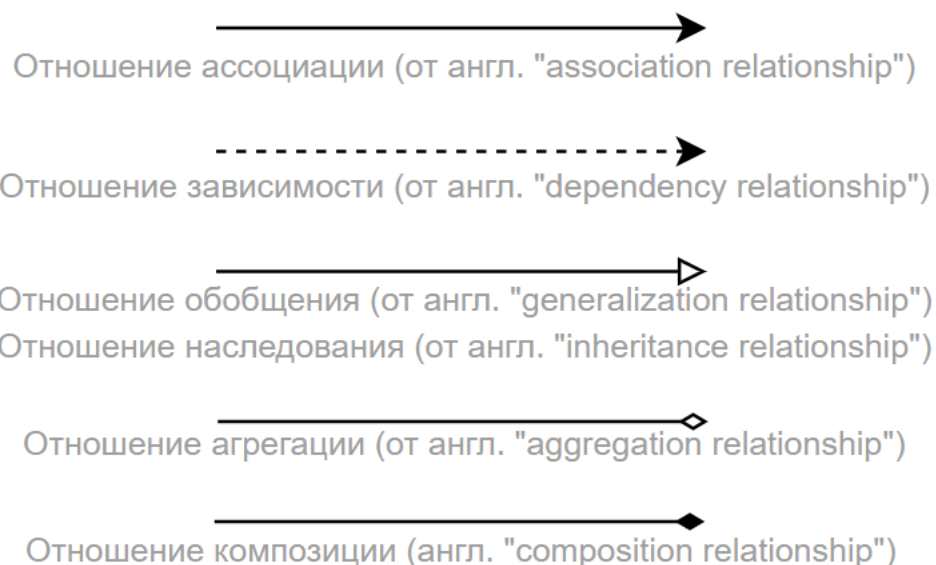
**3. Диаграмма состояний (State Diagram):** Используется для моделирования состояний объектов и переходов между ними в ответ на события. Это полезно при проектировании систем с объектами, которые могут находиться в разных состояниях.

**4. Диаграмма активности (Activity Diagram):** Представляет собой графическое представление алгоритмов или процессов в системе. Она описывает, как система выполняет действия и какие решения принимаются на каждом этапе.

**5. Диаграмма коммуникации (Communication Diagram):** Подобна диаграмме последовательности, но с упором на коммуникацию между объектами. Она показывает объекты и сообщения, которые передаются между ними в рамках определенной функциональности.

Классификация диаграмм на структурные и поведенческие помогает организовать моделирование различных аспектов программных систем.

## 2) Какие связи между классами можно изобразить на UML диаграмме класса



**1. Ассоциация (Association):** Это базовый тип связи, который показывает, что классы имеют некоторую связь или отношение друг к другу. Он обозначается линией, соединяющей классы, и может иметь множество различных атрибутов, таких как множественность (например, один-к-одному, один-ко-многим).

**2. Зависимость (Dependency):** Этот тип связи показывает, что один класс зависит от другого класса, например, при использовании методов другого класса. Он обозначается пунктирной стрелкой, указывающей на зависимый класс.

**3. Обобщение (Generalization):** Этот тип связи используется для представления наследования, когда один класс (подкласс или производный класс) наследует атрибуты и методы другого класса (базового класса или суперкласса). Он обозначается линией с треугольником, указывающим на суперкласс.

**4. Агрегация (Aggregation):** Это тип связи, который указывает, что один класс является частью другого класса. Это обозначается стрелкой, указывающей на агрегирующий класс, и может использоваться, например, для представления составных объектов.

**5. Композиция (Composition):** Этот тип связи аналогичен агрегации, но с более сильной связью. Когда объект составной, его существование зависит от объекта, который его содержит. Это обозначается стрелкой, с закрашенной головой стрелки.

### 3) Что такое стереотип

**Стереотипы в UML** - это механизм расширения языка, который позволяет добавлять пользовательские атрибуты, операции и аннотации к элементам модели для более детального описания их семантики или поведения. Стереотипы помогают более точно определить, как элементы модели должны использоваться и какие дополнительные свойства они имеют. Стереотипы часто представляют собой имена или метки, заключенные в угловые скобки, например, «<<web-сервис>>» или «<<singleton>>».