1. UML (Unified Modeling Language) - это язык графического моделирования, используемый для описания, визуализации, проектирования и документирования программных систем.
2. Типы диаграмм UML:

* Диаграмма классов
* Диаграмма объектов
* Диаграмма компонентов
* Диаграмма развертывания
* Диаграмма пакетов
* Диаграмма состояний
* Диаграмма последовательности
* Диаграмма деятельности
* Диаграмма прецедентов

1. На диаграмме классов могут быть следующие отношения между классами:

* Наследование (отношение "является")
* Ассоциация (отношение "имеет")
* Агрегация (отношение "часть-целое")
* Композиция (особый случай агрегации, когда часть не может существовать без целого)
* Реализация (отношение "реализует интерфейс")

1. Абстрактный класс на диаграмме классов обозначается курсивом над названием класса.
2. Интерфейс на диаграмме классов обозначается штриховой рамкой и именем класса в угловых скобках.
3. Доступность членов класса (публичный, приватный, защищенный и т.д.) на диаграмме классов обозначается специальными символами перед названием члена класса. Например, плюс перед названием метода означает, что метод публичный.
4. Агрегация - это отношение "часть-целое", когда часть может существовать и без целого. На диаграмме классов агрегация обозначается в виде прямой с ромбом на конце..
5. Ассоциация - это отношение "имеет", когда один класс связан с другим классом. На диаграмме классов ассоциация обозначается линией, соединяющей классы, с указанием типа связи и ее мультиплицированности.
6. На диаграмме последовательности используются следующие обозначения:

* Объекты, между которыми происходит взаимодействие, обозначаются вертикальными линиями с названием объекта вверху.
* Сообщения, передаваемые между объектами, обозначаются стрелками, соед

1. используется для визуального представления взаимодействия между объектами или компонентами системы во времени. Она показывает последовательность выполнения операций или вызовов методов, а также обмен сообщениями между объектами.
2. Диаграмма использования (Use Case) используется для описания функциональности системы с точки зрения ее пользователей. Она показывает, как система взаимодействует с актерами (пользователями), какие задачи они могут выполнять, и как система должна реагировать на их действия.