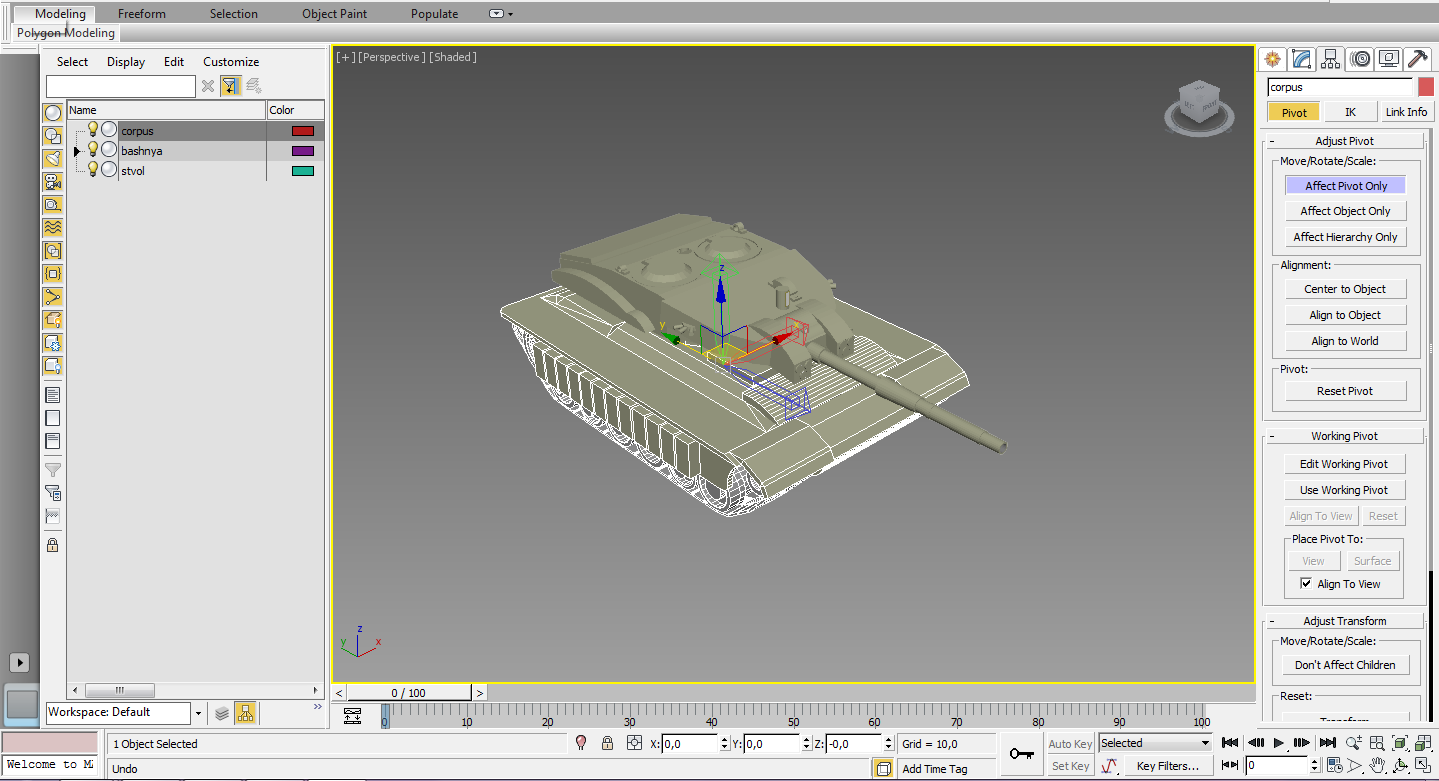
**Тема 13. Управление 3D-моделью танка в среде Unity**

1. Отредактировать в **3ds MAX** созданную по **Теме 9** на основе примитивов модель танка из трех составляющих частей: корпус, башня, ствол и дать им по отдельности соответствующие имена.
2. При создании танка необходимо изменить направление осей координат точки привязки **Pivot** всех его частей в соответствии с рисунком для корпуса и аналогично для башни и ствола, чтобы направление оси **Y** совпадало с осями в редакторе **Unity** (подробнее см. **Лекцию 13**).



1. Отредактировать поверхности отдельных частей модели танка для придания схожести с реальным современным танком. Наложить соответствующие текстуры на объекты, составляющие модель танка.
2. Экспортировать в среде **3ds MAX** модель танка в формате **3DS** (или **FBX)** для последующего импорта в среду **Unity** командой **Export>**, а также сохранить исходник в формате **max** для возможной коррекции исходной модели танка.
3. Создать в среде **Unity** холмистую поверхность **Terrain** с плоской частью в центре и холмами (горами) по краям.
4. Импортировать модель танка в среду **Unity,** скопировав файл с танком в формате **3DS** в папку **Assets** проекта**,** при этом в **Assets** будет создан объект в виде префаба танка**,** и затем разместить его перетаскиванием префаба танка на сцену в начале плоской части территории.

*Если импортированный танк не соответствует размерам созданного* ***Terrain****, необходимо отредактировать значение* ***Scale Factor*** *танка.*

*Если текстура объекта легла неправильно (например, повторяется много раз), необходимо изменить значение* ***Tilling*** *в материале текстуры.*

1. Создать иерархию подчиненности частей танка в **Unity** наложением в окне **Hierarchy** при перетаскивании мышью объектов: ствол на башню, башню на корпус в соответствии с теми именами, которые были присвоены этим объектам в **3ds MAX** перед импортом в среду **Unity**. При этом можно убрать из иерархии объект – весь танк, т.е. имя префаба танка, оставив только составляющие его компоненты: корпус, башню и ствол.
2. Добавить танку в Инспекторе компонент **Rigidbody** и задать ему массу *(****Mass****)* около 1000.
3. Добавить скрипт как компонент в модель танка на верхний объект иерархии, в нашем случае это его корпус и создать в нем программный код для управления движением танка и вращением его отдельных компонент, используя материалы **Лекции 13**.
4. Запустите приложение, нажав кнопку **Play,** и проверьте работу созданного сценария управления танком. Для реалистичного передвижения возможно будет необходимо отрегулировать скорость и массу танка.
5. (\*. Дальнейшие пункты задания выполнить факультативно) Добавьте на сцену в горизонтальной части поверхности напротив танка три объекта окружения: тонкую высокую стенку (Box), вытянутые цилиндр (Cylinder) и капсулу (Capsule). Добавьте каждому из объектов компоненту **Rigidbody** и раскрасьте их разными цветами или наложите текстуры.
6. Перед «стенкой» разместите триггерный коллайдер и добавьте для него скрипт на основе функции **OnTriggerStay**(), который заставляет вращаться стенку вокруг вертикальной оси, если в триггере находится танк (триггер сделайте невидимым – выключить его компоненту **MeshRenderer**).
7. Разместить на сцене два точечных источника света типа **Point** красного и синего цвета с начальной интенсивностью **Intensity** равной 0 и увеличенной **Range** областью действия равной примерно 100.
8. Разместите по краям сцены два триггерных коллайдера и создайте для каждого из них по скрипту с использованием двух функций **OnTriggerEnter()** и **OnTriggerExit()**, которые при входе в них танка будут включать красный или синий цвета дополнительной освещенности (задайте высокую интенсивность равной, например, 100) от размещенных на сцене источников света и соответственно выключать при выходе танка из соответствующих триггерных коллайдеров, создавая тем самым вспышки красного или синего цвета на всей сцене.
9. Добавьте для танка скрипт, который генерирует падение «бомб» на танк принажатии на определенную клавишу. Используйте для генерации примитив **Capsule**. Задайте для генерируемых объектов соответствующий цвет. Для расчета области падения выберите площадку размером вдвое больше размеров танка с разбросом случайным образом с помощью конструкции **Random.Range()** ицентром вблизи танка (ориентируйтесь на координаты центра танка на сцене в компоненте Инспектора **Transform/Position**).
10. Запустите игру и, генерируя нажатием заданной клавиши падение «бомб», попробуйте объехать зону бомбежки.