Вторая лекция

Мои заметки по второй лекции курса разработки игр на Unity

Keywords:

- RigidBody
- Collider
- Joint
- Animation
- Animation Controller
- Prefab
- Extra

RigidBody

- Двигать gameobject лучше через rigidbody, а не через transform, т.к. через transform будут проблемы с физикой (силы, скорости) - будут разрывы в движении
- Кэшировать rigidbody в Awake
- isKinematic, если активирован, то gameobject не двигается из-за движка, только тогда, когда мы сами того захотим программно

Collider

- `MeshCollider` лучше использовать для окружения и делать static, где возможно
- Static gameobject обсчитывается каждый `fixed update`
- `isTrigger`, если активирован, то `collider` не воспринимается, как твердое тело, т.е. проходит через другие коллайдеры без столкновений
- Если хотим двигать `collider`, то желательно вместе с этим двигать `rigidbody`, т.к. они обсчитываются взаимно и при отдельном обсчете возрастает нагрузка

Joint

- Виды joints
  - `Hinge Joint` - вращение вокруг точки. Например, вращение на шарнире или вращение двери на петлях
  - `Spring Joint` - изменение расстояния. Например, пружина или армотизация у колес машины
  - `Fixed Joint` - нет вращения или изменения расстояния между gameobjects
- `BreakForce`/`BreakTorque` - сила/момент силы, после которого `joint` разрушается
- Для Joints нужен `rigidbody`
- Если не хотим, чтобы gameobjects сталкивались, то ставим галку на `isCollisionEnabled`, но обычно этот флаг не нужен
- Обращать внимание на точки соединения!
- enter - точка соединения
- Два joint лучше не делать. Для этого есть configurant joint
- `configurant joint` - complex joint

- У joints есть ограничения по `limits`, но нужно поставить галку useLimits

Animation

- Импортированный Animation Clip доступен только для чтения
- Есть два режима создания animation clip из unity
  - фиксированными ключами с нулевой производной
  - кривой с ключами, в которых задана производная
- скиннинг - процесс натягивания весов костей на опорные точки скелета

Animation Controller

- Руководит за переход анимационных клипов
- Можно указать время на переход `hasExitTime`
- В блок `AnyState` входит каждый кадр
- Из `AnyState` переходить в другие клипы только по условию
- Если событие пришло из `AnyState` на переход в клип, который уже идет, то он продолжит идти с нуля
- `BlendTree` - смешивание анимаций
- Анимационные события. Например, вставка гранаты в руку, когда замахнулись гранатой

Extra

- `var` лучше использовать вместо явного присвоения типа
- Ввод лучше делать через GetAxis