**2. UHealthComponent**

1. Создаем наш новый компонент, от кого наследуем, как назвали, как сразу прибрались в автоматически сгенерированном коде и почему?

2. Добавляем переменную для здоровья, геттер, еще одну переменную для максимального здоровья (какой определяем UPROPERTY у последней?), где определяем присваивание нашей переменной здоровья переменной максимального здоровья?

3. Создаем компонент у персонажа, в чем его особенность?

4. Как сделать так, чтобы значение здоровья отображалось в текстовом поле над персонажем? Компонент и все остальное

5. Макрос проверки на нулевые указатели, как работает и в каких версиях билдов?

6. Заходим в блюпринт, настраиваем наш текст

7. Как проверили работу макроса check() и как он, собственно, работает?

1. Начнем с создания компонента в UE, который будет контролировать здоровье нашего персонажа.

В данном курсе мы будем **плотно** работать с компонентами и стараться **инкапсулировать все поведение в данные компоненты, чтобы не раздувать основной класс персонажа**.

В папке Components создаем класс, наследующийся от ActorComponent. Назовем STUHealthComponent.

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Первым делом изменим обновление нашего компонента на каждом тике игрового цикла bCanEverTick на false.

В данном классе тик не понадобится, так как всю логику будем обрабатывать на ивентах, поэтому в целях оптимизации убираем функцию TickComponent().

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

2. Добавим самую главную переменную, отвечающую за здоровье нашего персонажа (текущее), а также геттер для нее.

Изображение выглядит как текст

Автоматически созданное описание

А также добавим protected-переменную, отвечающую за максимальное здоровье персонажа. Сделаем ее доступной в блюпринтах и ограничим значениями от 0 до 1000.

UPROPERTY(EditDefaultsOnly, BlueprintReadWrite, Category = "Health", meta = (ClampMin = "0.0", ClampMax = "1000.0"))

float MaxHealth = 100.0f;

В BeginPlay() присваиваем Health значение MaxHealth (то есть каждый раз, когда игра начинается или наш персонаж спавнится).

3. Создаем данный компонент у нашего персонажа STUBaseCharacter:

Изображение выглядит как текст

Автоматически созданное описание

Как обычно создаем его в конструкторе персонажа, однако отличие в том, что данный компонент чисто логический и его не надо привязывать к нашему корневому элементу.

Изображение выглядит как текст

Автоматически созданное описание

4. Теперь давайте сделаем так, чтобы значение здоровья отображалось в текстовом поле над персонажем. Для этого добавляем TextRenderComponent нашему персонажу и в этот раз привязываем его к нашему корневому элементу:

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

Так же в функции Tick нашего персонажа будем передавать из нашего компонента здоровья значение здоровья в текстовую компоненту:

Изображение выглядит как текст

Автоматически созданное описание

5. Заметим, что мы не проверяем данные компоненты на нулевые указатели, так как подразумевается, что они не могут быть таковыми, если у нас создан объект, однако в UE есть макрос check() для проверки, который сгенерирует ошибку, если данные компоненты нулевые:

Изображение выглядит как текст

Автоматически созданное описание

Данный макрос работает только в Development и Debug билдах, в Shipping версии они просто игнорируются.

6. В блюпринте видим, что наши компоненты создались. Компонент текста необходимо отцентровать, поднять, изменить цвет и развернуть на 180 градусов (так как камера у нас ЗА персонажем).



7. Для проверки макроса check() закомментируем создание HealthComponent, запустим игру снова и сразу словим ошибку:

Изображение выглядит как текст

Автоматически созданное описание