1. Рассмотри основной способ логирования в UE – макрос UE\_LOG. Выведем основные типы в консоль и посмотрим, как эта консоль открывается через editor.

2. Макрос UE\_LOG() имеет несколько параметров:

CategoryName - макрос категории логирования, это некоторая метаинформация (по сути это дополнительная строчка, которая добавляется к выводу – по ней можно фильтровать логи и определять, из какого места программы у нас печатается сообщение) – сейчас воспользуемся встроенной категорией LogTemp.

Verbosity или уровень логирования – их также несколько, сейчас возьмем Display (так же распространены Warning и Error).

строчка вывода (строковый литерал) – сообщение, которое будет выведено с ошибкой. Однако просто так передать строчку в кавычках нельзя, нужно обернуть ее в макрос TEXT(). Данный макрос является оберткой над строкой и он возвращает расширенную строку (можно передавать туда различные символы, иероглифы и т.п. и все будет хорошо отображаться).

UE\_LOG(LogTemp, Display, TEXT("Hello, Unreal!"));

UE\_LOG(LogTemp, Warning, TEXT("Hello, Unreal!"));

UE\_LOG(LogTemp, Error, TEXT("Hello, Unreal!"));

3. Полученные логи ищем в Window->DeveloperTools>OutputLog.

4. Как залогировать основные типы?

// Called when the game starts or when spawned

void ABaseGeometryActor::BeginPlay()

{

Super::BeginPlay();

UE\_LOG(LogTemp, Display, TEXT("Hello, Unreal!"));

UE\_LOG(LogTemp, Warning, TEXT("Hello, Unreal!"));

UE\_LOG(LogTemp, Error, TEXT("Hello, Unreal!"));

int WeaponsNum = 4;

int KillsNum = 7;

float Health = 34.435f;

bool IsDead = false;

bool HasWeapon = true;

UE\_LOG(LogTemp, Display, TEXT("Weapons num: %d, kills num: %i"), WeaponsNum, KillsNum);

UE\_LOG(LogTemp, Display, TEXT("Health: %.2f"), Health);

UE\_LOG(LogTemp, Display, TEXT("IsDead: %d"), IsDead);

UE\_LOG(LogTemp, Display, TEXT("HasWeapon: %d"), static\_cast<int>(HasWeapon));

}

Форматировать вывод можно через типичные спецификаторы формата. Обращаю внимание, что перечисляемые аргументы для вставки должны быть за пределами макроса TEXT().

Перенесем данный функционал в функцию printTypes() класса

// ...

BaseGeomteryActor:

private:

void printTypes();

// ...

void ABaseGeometryActor::printTypes()

{

int WeaponsNum = 4;

int KillsNum = 7;

float Health = 34.435f;

bool IsDead = false;

bool HasWeapon = true;

UE\_LOG(LogTemp, Display, TEXT("Weapons num: %d, kills num: %i"), WeaponsNum, KillsNum);

UE\_LOG(LogTemp, Display, TEXT("Health: %.2f"), Health);

UE\_LOG(LogTemp, Display, TEXT("IsDead: %d"), IsDead);

UE\_LOG(LogTemp, Display, TEXT("HasWeapon: %d"), static\_cast<int>(HasWeapon));

}

Файлы с логами можно найти в папке проекта в папке Saved.

5. Спецификаторы формата

Изображение выглядит как стол

Автоматически созданное описание