虚幻 5.3 零基础入门视频教程-宇宙最简单系列（Unreal Engine5 UE5）

特点：虚幻 5 从入门到起飞　火爆宇宙　超高人气　无脑学

SiKi 老师 www.sikiedu.com

思维导图

https://docs.qq.com/flowchart-addon

介绍和前提

虚幻引擎介绍

1. 什么是游戏引擎
    a. 游戏开发工具集
2. 跨平台
3. 知名游戏作品：
    a. 绝地求生-和平精英
    b. 黑神话·悟空
    c. 仙剑奇侠传
    d. 双人成行
    e. 生化危机
    f. 堡垒之夜

虚幻 5 最新功能

1. Nanite 虚拟多边形几何体
2. Lumen 全局光照解决方案

本课程优点

1. 真正的零基础（没有游戏开发基础的同学）
    a. 会告诉你什么东西需要学习到什么程度
2. 计划出 20 个小时左右，2-4 季左右的教程讲解

a. 目标：虚幻完整入门课程

开发学习方法

1-一定要跟着练习（在理解的基础上练习）

2-边学边忘是正常的（学到后面，忘记了就回头查，不需要可以记忆）

a-利用 chatgpt 查询

3-不要过于在乎细节（不求甚解）

a-虚幻里面的设置很多，属性很多，编辑界面属性也很多

b-忽略细节，关注大局

学习技巧

1. 如何解决问题-谷歌和百度
   示例：如何显示隐藏项目
2. 观看速度可以自行调整
3. 学习笔记（网易云笔记、印象笔记）
   a. 不是为了复习，是为了辅助达到整理知识
4. 借助大规模语言模型进行学习和查询
   a. 文心一言、通义千问
   b. chatgpt
5. 一键三连 学习效率加成

常见问题答疑

1. 游戏开发选择 Unreal 还是 Unity；
   a. Unreal 渲染效果好有完善的蓝图支持；Unity 开发快捷
   b. Unreal 学习成本和开发成本相对来说比 Unity 高
   c. 招聘岗位数量和市场占有率 Unity 有 48%；虚幻有 13%
      https://www.mycaijing.com/article/detail/503534?source_id=40#:~:text=%E7%AC%AC%E4%B8%89%E6%96%B9%E5%B8%82%E5%9C%BA%E8%B0%83%E7%A0%94%E6%9C%BA%E6%9E%84,%E5%BC%95%E6%93%8E%E5%B8%82%E5%9C%BA%E7%9A%84%E5%8F%8C%E5%AF%A1%E5%A4%B4%E3%80%82
   d. 招聘竞争难度，虚幻竞争会少一些
   e. 未来发展预测：两个引擎会继续保持各自的优势，和目前的市场占有率目前来看不会有太大的变化
   f. 学习建议：选择其中一个深入学习，然后未来有时间了在对其他引擎进行学习，达到略懂

2. 学习 Unreal 是否一定需要学习 C++;
3. 游戏开发是否要求英语要好，是否要求数学要好;

适宜人群

a. 想做独立游戏的
b. 想找工作的
c. 游戏美术、策划
d. 工业、家装设计师

虚幻下载和安装

配置参考

官网：https://www.unrealengine.com/zh-CN/download

老师建议：5K 左右或以上的游戏本（1T 以上的固态硬盘）

下载安装步骤

1. 官网（搜索关键词：虚幻 Epic） unrealengine.com
2. Launcher 设置
3. 安装选项设置
4. 目录在英文目录下

版本：

UE5.3.2

【可选】C 盘空间释放

1. 缓存路径修改
   默认缓存路径：*C:\Users\souke\AppData\Local\UnrealEngine\5.3\DerivedDataCache*
   修改方式：
   a. 打开对应引擎版本文件目录：*C:\Program Files\Epic Games\UE_5.3\Engine\Config* 中的 BaseEngine 文件
   b. 查找条目 *InstalledDerivedDataBackendGraph* 或 *%ENGINEVERSIONAGNOSTICUSERDIR%DerivedDataCache*
   c. 修改 path 为 "%GAMEDIR%DerivedDataCache"

2. 保管库（下载的资源、项目）缓存位置
   通过启动器设置

建议运行和文件路径位于固态硬盘

创建工程注意事项

1. 路径和名字中不要有中文
2. 后续开发中都尽量避免使用中文（除非是要显示中文提示的情况）

引擎基本知识

虚幻引擎界面

1. 基本窗口
    a. 关卡窗口（Map 窗口）
    b. 大纲窗口
    c. 细节窗口
    d. 内容浏览器（Ctrl+空格）
2. 窗口分类
    a. 彩色窗口
    b. 黑白窗口
3. 窗口操作
    a. 移动
    b. 隐藏选项卡
    c. 停靠侧边栏
4. 窗口的布局
    a. 修改
    b. 保存和加载

学习提示
我们讲解中有很不同的操作方式和快捷键，这些不需要全部掌握，只需要记得一个当前自己最适合的就可以。

虚幻设置

1. 编辑-编辑器偏好设置
    a. 语言设置
2. 编辑-项目设置
3. 设置-世界场景设置
    a. 另外打开方式：窗口-世界场景设置
    b. 概念：一个地图，就是一个关卡，就是一个世界。Map=Level=World

内容浏览器

1. 打开快捷键：Ctrl + 空格

2. 保存所有
3. 加载初学者内容包
4. 资源的导入、导出和迁移
5. 资源搜索-滤波-集合-收藏夹
6. 资产搜索：Ctrl+P

游戏运行和退出

1. ESC 退出运行
2. F11 切换是否全屏

视野操作

场景漫游

1. 鼠标右键+WSAD QE
2. 移动速度调整：鼠标滑轮 右上角选项

什么是 Map-关卡

1. 新建
    a. Ctrl+S
    b. 文件-》保存
    c. 保存按钮
2. Map Actor Component
3. 游戏的运行
    a. 运行
    b. 停止
    c. 弹出
    d. 暂停

关卡的相关显示设置：

什么是 Actor-演员

游戏场景的组成元素

什么是组件 Component

1. 内置组件
2. 自定义组件
    a. 蓝图组件
    b. C++组件

组成关系

一个游戏（项目）由多个 Map 组成，每个 Map 中存在多个 Actor，不同的 Actor 由不同的 Component 组成，不同的 Actor 为什么外观和作用不一样主要是由于它身上的组件造成的。

Actor 的操作

视野控制
选择控制
变换控制

基本模型的添加和模型变换操作

1. 基本模型的添加
    a. 基本几何体

2. 模型的操作

a. 视野

i. F 双击 聚焦

ii. Alt+鼠标左键 围绕焦点旋转

b. 工具选择

i. 右上角

ii. QWER

c. 移动 - 旋转 - 缩放

i. 单位 厘米 度 倍

ii. 移动 单个轴 单个面 End 快捷键

1. Shift+移动

iii. 旋转 单个轴

iv. 缩放 单个轴 单个面 整体

世界坐标系和本地坐标系（世界空间和本地（局部）空间）

上下左右和坐标轴的对应关系

控制台

1. 打开方式

a. 下方

b. “~”打开

2-屏幕调试信息

a. DisableAllScreenMessages

Actor 操作 - 复制、删除、禁用、归类、父子关系

复制

1. Alt+移动（注意选择的是组件还是Actor）
2. Ctrl + D
3. 右键

删除

1. Del Backspace
2. 右键

物体选择

Ctrl 加选和减选

Shift 批量选择

其他移动操作

1. 按下 v 顶点吸附
2. 鼠标中间按下可以临时移动中心点

中心点：计算这个模型位置的点（一般位于物体的底部中心，变换物体的时候的点，可以在建模软件中修改。）

资源的添加

1. 工程内置资源（初学者内容包）
2. 虚幻商城
3. Quixel Bridge（要登录虚幻账号）
4. 导入外部资源

外部模型导入

模型软件

Blender 3DMax Maya

格式

.fbx .3ds .maya .blend

网站

free3d.com

大纲窗口

1. 通过文件夹进行分类
2. 隐藏和显示
3. F2 重命名
4. 父子关系
5. Ctrl+G Shift+G 组合和取消组合

基本模型、静态网格组件和静态网格

作用：控制物体渲染的样子
模型：
1. 基本模型

静态网格组件基本属性
控制显示的
1. 静态网格
2. 材质
控制物理的
1. 移动性：静态、可移动
2. 模拟物理
3. 质量

几何体-笔刷

1. Brush Type：Additive Subtractive
2. 表面属性和画刷设置
3. 笔刷编辑模式

材质

作用：控制模型的外观的

节点介绍

1-主节点和其他节点

2-左边进 右边出

3-Alt 删除连接

4-Ctrl 移动连接

常用的节点：

1+左击　一个值　　速度、亮度

2+左击　两个值　　二维的方向（向量）、二维的坐标

3+左击　三个值 x y z　　三维的方向、三维的坐标、颜色值 RGB

4+左击　四个值 R G B A　　带透明度的颜色值

主节点

1. 基础颜色
2. Metallic（金属度）
3. 粗糙度

Light 光源

光源类型

1. PointLight-点光源（灯泡、蜡烛）
2. SpotLight-聚光灯（射灯、手电筒）
3. RectangularLight-矩形光源
4. SkyLight 天空光照（天空中的漫反射，天空中的环境光）

环境光照混合器

位置：窗口-环境光照混合器

1. 天空光照 SkyLight - 环境光 - 环境漫反射
2. 大气光源 DirectionalLight - 直射光 - 太阳
3. 天空大气 SkyAtmosphere

4. 体积云 VolumetricCloud
5. 高度雾 ExponentialHeightFog

PlayerStart

作用：设置主角的生成位置

1. 通过 PlayerStart 设置生成位置
2. “右键 - 从此处运行”
3. 设置生成点
    a. 当前相机位置
    b. 默认玩家出生点

组件

列表

1. 静态网格组件
2. 光源组件

分类

1. 场景组件（Scene Component）
    a. 对场景有视觉影响的组件
2. 非场景组件（Non-Scene Component）
    a. 功能性组件

体积

描述：一个 3D 的空间

1. TriggerVolume
    a. 触发体积：检测进入了某个区域。
2. PainCausingVolume
    a. 伤害施加体积
3. KillZVolume

4. PhysicsVolume
5. PostProcessVolume

项目设置

1. 编辑器开始地图
2. 游戏默认地图

蓝图 - 蓝图类 - BluePrint（Class）

蓝图类是什么？

概念：蓝图也叫做蓝图类

蓝图是一个游戏逻辑的可视化的编辑工具。

蓝图就是编写游戏逻辑的

什么是类？

关于蓝图类的理解

在蓝图中的学习，里面某些概念是直接引用了 C++语法中的概念，所有如果有编程基础更有助于理解蓝图，因为每一个蓝图本质上就是一个类。

不过没有 C++编程基础也没有关系，忽略哪些 C++中的概念比如（类、继承），重点学习怎么使用即可。

对于要深入学习的童鞋，还是要做好学习 C++的准备哦。（虚幻）

蓝图分类

1. 基于关卡的蓝图-关卡蓝图（一个关卡只有一份）
2. 基于 Actor 的蓝图-Actor 蓝图（有多少个 Actor 就有多少份蓝图）
3. 基于 Component 的蓝图-组件蓝图（有多少个 Actor 上有这个组件，就有多少份蓝图）

一般情况下这个蓝图类中的逻辑是控制谁的，那么就把这个蓝图类放在哪里。

游戏逻辑写在哪里，主要是分析这个游戏逻辑是跟谁相关的，那么就放在哪个蓝图类里面。

蓝图不是一个新的功能什么的，只是游戏逻辑的一种表现方式。游戏逻辑可以通过编写 C++代

码方式表达，也可以通过蓝图，这种可视化编辑的方式。

基于 Actor 的蓝图

理解：创建一个 Actor 的模版（创建一个可复用，带有蓝图游戏逻辑的模版）

1. 根据一个 Actor 创建
2. 根据多个 Actor 创建

为什么蓝图可以基于 Actor，也可以基于 Component 来创建。

因为本质上 Actor 和 Component 也是一个东西，都是带有一定游戏动能（游戏逻辑的）C++类。

虚幻 5 基本架构思维导图

https://docs.qq.com/mind-addon

蓝图编辑操作

蓝图节点基本介绍

分类：
1. 主动执行节点
2. 被动调用节点（纯结点、纯函数）

左侧：输入　右侧：输出

线的操作
1. Ctrl 移动
2. Alt 删除

蓝图数据类型

基础类型：

小数

　浮点 float

整数

integer

integer64

字符串

引用类型：

Actor 类型

Component 类型

复合类型

数组

蓝图结点

Actor 拥有的节点

1. GetCurrentLevelName
2. OpenLevel
3. GetPlayerPawn
4. GetPlayerCameraManager
5. StartCameraFade
6. DestroyActor
7. SetActorLocation
8. Lerp（Vector）
9. SpawnActorFromClass
10. ApplyDamage
11. GetAllActorsOfClass
12. MakeLiteralFloat（创建一个浮点类型的值）

Pawn 拥有的功能

1. GetActorLocation
2. GetActorScale3D

3. SetActorScale3D

场景组件拥有的功能

1. GetWorldLocation

2. GetWorldRotation

粒子系统组件

1. Active

分支节点

1. Branch

判断节点

1. ==

函数分类：

1. 纯函数
    a. 固定的输入固定的输出
    b. 不对外界有影响
2. 普通函数

蓝图节点分类：

1、事件节点（Event）

2、宏节点

3、函数节点

4、纯函数节点（不改变游戏状态）

蓝图事件

1. Event BeginPlay
2. Event ActorBeginOverlap
3. Event Tick
4. Event Hit
5. Event AnyDamage（ApplyDamage）

蓝图整洁

1. 折叠到节点（创建一个子图）
2. 折叠到函数
3. 折叠到宏

开发中会遇到的问题

问题 1 - 打包问题，UE5 发布时出现“Windows 的 SDK 未正确安装”错误解决方法。

1. 安装 WindowsSDK
2. 重启
3. （可选）强制更新设备
4. 重启

文章：https://blog.csdn.net/qq_16599281/article/details/127543840