

绪论

1. 虚拟现实的过去、现在和将来
2. 虚拟现实的应用领域
3. 虚拟现实与人工智能

PART 1 入门篇- 新手的unity3d漫游指南

第1章 初识神兵- 为什么选择Unity3d

主要内容：常用3D引擎对比分析，说明使用Unity3d进行VR/AR开发的优缺点

1.1 常用3D引擎介绍

- 1.1.1 游戏引擎的出现
- 1.1.2 游戏引擎架构基础
- 1.1.3 主流3D引擎对比分析

1.2 Unity的发展史

- 1.2.1 Unity引擎的发展史
- 1.2.2 Unity引擎的代表作品
- 1.2.3 Unity引擎的Roadmap

1.3 Unity软件安装、授权与服务

- 1.3.1 在Windows下的安装
- 1.3.2 在Mac 下的安装
- 1.3.3 Unity授权类型
- 1.3.4 Unity服务简介

1.4 本章小结

第2章 蹒跚学步- 开始使用Unity

2.1 Unity 编辑器界面使用简介

- 2.1.1 界面布局
- 2.1.2 工具栏
- 2.1.3 菜单栏
- 2.1.4 Hierarchy
- 2.1.5 Inspector
- 2.1.6 Project

2.1.7 Scene

2.1.8 Console

2.2 Unity Asset Store 开发资源介绍

2.2.1 Asset Store简介

2.2.2 Asset Store的基本使用

2.3 游戏对象、组件和Prefabs

2.3.1 游戏对象简介

2.3.2 组件简介

2.3.3 Prefabs简介

2.4 本章小结

第3章 无往不利- 在Unity中使用C#进行编程

3.1 3D数学基础知识

3.1.1 3D坐标系

3.1.2 向量

3.1.3 矩阵

3.1.4 齐次坐标

3.1.5 四元数

3.2 C# 基本语法

3.2.1 基本数据类型

3.2.2 变量和函数

3.2.3 控制流程

3.2.4 类和方法

3.3 在Unity中使用C#

3.3.1 创建脚本

3.3.2 使用MonoDevelop编辑器及Visual Studio

3.3.3 声明和使用变量

3.3.4 如何访问游戏对象和组件

3.3.5 Unity中的重要事件函数

3.3.6 Coroutine

3.3.7 Unity中的重要类

3.4 Unity开发学习的常用资源获取

3.5 本章小结

PART 2 进阶篇- Unity3d开发实战

第4章 创建一个新世界-Modelling

4.1 3D建模技术简介

4.1.1 主流3D建模软件简介

4.1.2 主流3D扫描技术简介

4.1.3 动作捕捉和表情捕捉技术

4.2 创建3D模型的基本流程

4.2.1 基础3D模型的创建-场景

4.2.2 基础3D模型的创建-人物

4.2.3 3D模型的重拓扑

4.2.4 展UV

4.2.5 创建纹理贴图

4.2.6 基于PBS理论制作材质

4.3 ShootingQuest游戏实战-创建项目并准备所需的美术资源

4.3.1 ShootingQuest的游戏背景

4.3.2 ShootingQuest游戏的策划与设计

4.3.3 创建ShootingQuest项目

4.3.4 给ShootingQuest项目添加Git版本控制

4.3.5 准备ShootingQuest项目所需的美术资源

4.3.6 导入ShootingQuest所需的美术资源

4.4 本章小结

第5章 有了光就有了一切- Enlighten

5.1 Unity光照系统（Enlighten）介绍

5.1.1 Light组件简介

5.1.2 常见的光源类型

5.1.3 Light inspector简介

5.2 全局光照

5.2.1 全局光照简介

5.2.2 烘焙

5.2.3 Lightmaps的使用

5.2.4 Light Probes和Reflection Probes的使用

5.3 ShootingQuest游戏实战-创建游戏场景并添加光照

5.3.1 导入游戏场景

5.3.2 添加点光源

5.3.2 添加Lightprobe和Reflection Probes

5.3.3 烘焙场景光照

5.3.4 如何有效的搜索和查询技术解决方案

5.4 本章小结

第6章 让游戏画面栩栩如生-粒子系统和shader、Image Effect的使用

6.1 Shuriken粒子系统

6.1.1 粒子系统的使用概述

6.1.2 Unity中的粒子系统简介

6.2 ShootingQuest游戏实战-完善游戏场景的视觉效果

6.2.1 添加火焰粒子系统效果

6.2.2 给游戏添加屏幕后期特效

6.3 Shader的使用

6.3.1 Shader概述

6.3.2 Unity的标准Shader介绍

6.3.3 如何创建自定义Shader

6.4 Image Effect的使用

- 6.4.1 Image Effect简介
- 6.4.2 Unity中内置的Image Effect介绍

6.5 本章小结

第7章 让角色动起来- Unity动画系统

- 7.1 Unity动画系统概述
 - 7.1.1 Legacy Animation System
 - 7.1.2 Mecanim动画系统

7.2 ShootingQuest游戏实战- 让游戏中的角色动起来

- 7.2.1 导入NPC角色资源
- 7.2.2 创建角色的Prefab
 - 7.2.2 添加Animator组件
 - 7.2.3 设置基础状态机
 - 7.2.4 添加状态机之间的切换
 - 7.2.5 编写控制角色动画的脚本

7.3 实战-更复杂的人物动画机制

- 7.3.1 如何使用Blend Trees
- 7.3.2 如何实现人形角色动画的配置和重定向

7.4 本章小结

第8章 有了目标才有方向- Pathfinding

8.1 Unity中的寻路系统

- 8.1.1 寻路系统内部工作原理
- 8.1.2 NavMesh 烘焙设置
- 8.1.3 NavMesh Obstacle
- 8.1.4 Off-Mesh Link

8.2 ShootingQuest游戏实战-在游戏中使用寻路系统

- 8.2.1 添加地板对象的寻路烘焙
- 8.2.2 在NPC角色对象上添加并设置NavMeshAgent组件

8.2.3 在脚本中添加寻路状态的动画切换

8.2.4 添加寻路距离

8.3 本章小结

第9章 真实世界的物理法则- 物理系统

9.1 Unity中的物理系统

9.1.1 物理系统概述

9.1.2 Rigidbody组件

9.1.3 Character Controller组件

9.1.4 碰撞体组件

9.1.5 布料组件

9.1.6 关节组件

9.1.7 其它组件

9.2 ShootingQuest游戏实战-给游戏添加物理系统

9.2.1 ShootingQuest游戏中的物理系统设计

9.2.2 在场景中添加物理碰撞系统

9.3 本章小结

第10章 且听风吟-音乐和音效

10.1 Unity中的Audio系统

10.1.1 Audio系统概述

10.1.2 Audio Source

10.1.3 Audio Reverb Zone

10.1.4 Audio Mixer

10.2 ShootingQuest游戏实战-给游戏添加背景音乐和音效

10.2.1 将音乐和音效资源导入到项目中

10.2.2 给游戏添加背景音乐

10.2.3 给游戏添加音效

10.3 本章小结

第11章 游戏中的UI界面系统

11.1 Unity 中的UI系统简介

11.1.1 NGUI插件简介

11.1.2 UGUI系统简介

11.1.3 Canvas组件简介

11.1.4 视觉组件简介

11.1.5 交互组件简介

11.1.6 自动布局

11.2 ShootingQuest游戏实战-给游戏添加UI

11.2.1 ShootingQuest游戏中的UI设计

11.2.2 给ShootingQuest游戏添加UI

11.2 本章小结

第12章 Unity中的数据存取与游戏性能优化

12.1 Unity中数据存取的常用方法

12.1.1 使用PlayerPrefs

12.1.2 使用Json

12.1.3 使用数据库

12.2 Unity中性能优化的原则和常用方法

12.2.1 Unity中的性能优化简介

12.2.2 Unity中的遮挡剔除

12.2.3 使用Profiler和Frame Debugger

12.2.4 对游戏资源进行审核

12.2.5 灵活使用AssetBundle加载资源

12.3 ShootingQuest游戏实战-添加数据存取机制并优化游戏

12.3.1 游戏数据存取机制的设计

12.3.2 实现游戏数据存取机制

12.3.3 实现游戏的核心逻辑机制

12.3.4 对游戏性能进行优化

12.3.5 完善ShootingQuest项目

12.3.6 ShootingQuest项目的配置和发布

12.4 本章小结

第13章 一个人的世界很孤单- Unity网络编程

13.1 UNET简介

13.1.1 UNET常见概念简介

13.1.2 UNET主要类简介

13.1.3 High Level API

13.1.4 Transport Layer API

13.1.5 如何设置Unity Multiplayer

13.2 虚拟聊天室-使用Unity创建一个简单的多人在线系统

13.2.1 Photon+ 和DFVoice插件简介

13.2.2 游戏的目标

13.2.3 创建Unity项目和基础场景

13.2.4 设置场景的光照和视觉效果

13.2.5 添加游戏角色和摄像机

13.2.6 导入Photon+插件和DFVoice插件

13.2.7 添加多人在线功能

13.2.8 添加语音对话功能

13.2.9 添加UI界面

13.2.10 添加背景音乐和音效

13.2.11 对游戏性能进行优化

13.2.12 对网络性能进行优化

13.2.13 项目配置和发布

13.3 本章小结

PART 3 高级篇- VR/AR开发实战-William（除第14章）

第14章 VR/AR开发技术入门

14.1 主流的AR/VR设备平台

14.1.1 Oculus Rift

14.1.2 Samsung Gear VR

14.1.3 HTC Vive

14.1.4 Sony PSVR

14.1.5 Google Daydream VR

14.1.6 VR一体机

14.1.7 微软HoloLens

14.1.8 Magic Leap

14.1.9 智能手机

14.2 主流的自然交互技术及SDK

14.2.1 Leap Motion

14.2.2 Intel Realsense

14.2.3 Google Project Tango

14.2.4 Siri

14.2.5 Google Assistant

14.2.6 Alexa

14.2.7 Cortana

14.2.8 科大讯飞语音识别技术

14.2.9 其它感知交互技术介绍

14.3 AR/VR应用开发的基本流程

14.3.1 选择合适的硬件设备平台

14.3.2 选择合适的开发引擎

14.3.3 选择恰当的交互方式

14.3.4 选择合适的第三方SDK或插件

14.3.5 设计产品时充分考虑AR/VR设备的特点

14.3.6 选择合适的产品发布平台

14.4 本章小结

第15章 实战开发HTC Vive平台上的FPS游戏

15.1 HTC Vive平台开发概述

15.1.1 HTC Vive设备及平台简介

15.1.2 Lighthouse技术原理

15.1.3 HTC Vive手柄交互详解

15.1.4 SteamVR插件简介

15.1.5 VRTK插件简介

15.2 VR版FPS游戏的产品策划

15.3 配置HTC Vive的开发测试环境

15.3.1 安装HTC Vive硬件

15.3.2 安装Vive和Steam软件

15.3.3 设置并打开SteamVR

15.3.4 运行SteamVR的测试场景

15.4 在Unity中导入所需的美术资源

15.5 导入SteamVR和VRTK插件

15.4.1 从AssetStore上下载并导入SteamVR插件

15.4.2 从Github上下载并导入VRTK插件

15.6 游戏的UI设计与开发

15.5.1 VR游戏中UI设计的注意事项

15.5.2 VR版FPS游戏的UI设计思路

15.5.3 添加3D版的开始菜单

15.5.4 添加手柄和菜单的交互事件

15.7 添加游戏的核心机制

15.8 添加背景音乐和音效

15.7.1 添加背景音乐

15.7.2 添加交互时的音效

15.9 将游戏发布到VivePort平台

15.8.1 VivePort 平台介绍

15.8.2 注册VivePort的开发者

15.8.3 上传并提交游戏

15.8.4 等待审核发布

15.10 本章小结

第16章 实战-开发Google Daydream VR平台上的《VR虚拟聊天室》

16.1 Google Daydream VR平台开发概述

16.1.1 Google Daydream VR设备及平台简介

16.1.2 Google Daydream VR中的交互

16.2 《VR虚拟聊天室》的产品策划

16.3 配置Google Daydream VR的开发测试环境

16.3.1 准备好相关的开发硬件

16.3.2 系统和软件的安装

16.3.3 运行 Daydream VR的测试场景

16.4 在Unity中导入所需的美术资源

16.5 添加科大讯飞的语音识别SDK

16.6 游戏的UI设计与开发

16.7 添加游戏互动逻辑机制

16.6.1 添加Photon+插件

16.6.2 添加对话功能

16.8 添加背景音乐和音效

16.7.1 添加背景音乐

16.7.2 添加游戏中的音效

16.9 将产品发布到Google Daydream VR平台

16.8.1 Daydream VR平台介绍

16.8.2 注册为Daydream VR的开发者

16.8.3 上传并提交游戏

16.8.4 等待审核发布

16.10 本章小结

第17章 实战使用Unity和高通Vuforia SDK开发《口袋动物园》AR游戏

17.1 高通Vuforia SDK简介

17.2 《口袋动物园》的产品策划

17.3 配置开发测试环境

17.3.1 注册为Vuforia Developer

17.3.2 申请AppID

17.3.3 在Unity中导入Vuforia SDK

17.3.4 上传图片并下载Package

17.3.5 进行设置

17.3.6 运行Vuforia的测试场景

17.4 在Unity中导入所需的美术资源

17.5 游戏的UI设计与开发

17.4.1 AR版《口袋动物园》的UI设计思路

17.4.2 设计《口袋动物园》的UI

17.6 添加游戏互动机制

17.7 添加背景音乐和音效

17.6.1 添加背景音乐

17.6.2 添加音效

17.8 将游戏发布到Android应用商城

17.9 将游戏发布到苹果AppStore

17.10 本章小结

第18章 实战开发HoloLens平台的《精灵宝可梦》游戏

18.1 HoloLens平台开发概述

18.1.1 HoloLens设备及平台简介

18.1.2 HoloLens中的语音交互

18.1.3 HoloLens中的手势动作

18.1.4 HoloLens中的SpatialMapping

18.2 HoloLens版的《精灵宝可梦》产品策划

18.3 配置HoloLens的开发测试环境

18.3.1 准备好HoloLens硬件

18.3.2 安装Visual Studio和其它软件

18.3.3 设置并运行HoloLens的测试场景

18.4 在Unity中导入所需的美术资源

18.5 游戏的UI设计与开发

18.4.1 HoloLens应用中UI设计的注意事项

18.4.2 AR版《精灵宝可梦》的UI设计思路

18.4.3 添加场景中的菜单

18.6 添加游戏的核心机制

18.5.1 让世界充满可爱的口袋妖怪

18.5.2 用手抓住你的最爱

18.7 添加背景音乐和音效

18.6.1 添加背景音乐

18.6.2 添加交互时的音效

18.8 将产品发布到Windows Store平台

18.7.1 微软Windows Store平台介绍

18.7.2 注册为微软的开发者

18.7.3 上传并提交游戏

18.7.4 等待审核发布

18.9 本章小结

附录

Unity3d中的快捷方式

本书中涉及的计算机术语