

- **案例教学** 基础篇精讲简单案例，项目篇详解综合案例
- **CDIO 框架** 从项目构思与设计出发，展现游戏开发的全流程
- **应用场景** 既可用于课堂教学，又可进行自主学习与进阶应用



Unity3D 游戏 设计与开发

曹晓明 编著

清华大学出版社

本书获深圳大学教材出版基金资助

Unity 3D 游戏设计与开发

曹晓明 编著

清华大学出版社

北京

内 容 简 介

本书由基础篇和项目篇两个部分组成，分为 13 章。其中，基础篇包括第 1~9 章，内容为环境准备、Unity 的开发环境、熟悉游戏物体和组件、创建 3D 游戏场景、物理系统、2D 动画、人形动画、UGUI 系统、粒子系统，针对 Unity 5.6 的新特性，介绍了地形系统、物理系统、粒子系统、UGUI 系统等模块；项目篇包括第 10~13 章，内容为兔子快跑、开心农场、寻找消失的大洲、保卫碉楼，通过由浅入深的工程案例介绍游戏项目开发的全过程。读者可通过兔子快跑游戏项目，掌握开发 2D 游戏的基本方法和技能；通过开心农场游戏项目，了解结合地形系统开发种植游戏的过程；通过寻找消失的大洲游戏项目，熟悉开发 AR 游戏的基本概念；通过保卫碉楼游戏项目，了解使用最新的动画系统制作 3D 游戏的方法。

本书尽量使用简单的代码实现功能，语言通俗易懂，配图丰富而具体，可作为高等院校或职业院校游戏设计相关专业（如教育技术、数字媒体技术、数字媒体艺术、计算机软件等）的教材，也适合 Unity 初学者、游戏开发爱好者和游戏美术人员使用。

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。

版权所有，侵权必究。侵权举报电话：010-62782989 13701121933

图书在版编目 (CIP) 数据

Unity3D 游戏设计与开发 / 曹晓明 编著. —北京：清华大学出版社，2019

ISBN 978-7-302-52261-4

I . ① U… II . ①曹… III . ①游戏程序—程序设计—教材 IV . ① TP311.5

中国版本图书馆 CIP 数据核字 (2019) 第 085647 号

责任编辑：王 定

封面设计：周晓亮

版式设计：思创景点

责任校对：牛艳敏

责任印制：丛怀宇

出版发行：清华大学出版社

网 址：<http://www.tup.com.cn>, <http://www.wqbook.com>

地 址：北京清华大学学研大厦 A 座 邮 编：100084

社 总 机：010-62770175 邮 购：010-62786544

投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈：010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者：三河市铭诚印务有限公司

经 销：全国新华书店

开 本：185mm×260mm 印 张：20.75 字 数：505 千字

版 次：2019 年 6 月第 1 版 印 次：2019 年 6 月第 1 次印刷

定 价：128.00 元

产品编号：078625-01



前 言

当前，游戏设计与开发课程主要存在三大问题：一是缺少同课程教学目标与内容相匹配的教材，二是缺少同 MOOC 课程等混合教学组织形式相匹配的教材，三是面向高等教育领域非游戏专业的教材比较少。本书面向计算机科学与技术、计算机软件、数字媒体技术、教育技术等专业的学生，以及其他游戏开发爱好者，是一套项目化、任务式的立体化教材。本书一方面服务于相关院校开设游戏设计相关的课程，支持教师开展面授或混合式的学习，另一方面也为游戏开发者提供丰富案例，支持自定步调的自主学习。

作为 3D 游戏开发引擎，Unity 由于跨平台能力强、开发流程简便快捷受到开发者们的喜爱。从 2005 年诞生至今，Unity 不断更新版本，从 Unity 1.0 到 Unity 5.6，又从 Unity 2017.x 到 Unity 2018.x，功能越来越强大，已成为开发者的首选工具。在游戏行业火爆的今天，学习 Unity 日益流行，也是初学者迅速进入游戏开发大门的首荐方式。以 Unity 作为游戏的开发引擎，具有上手快、开发场景丰富、发布跨终端等其他引擎不可比拟的优势，因此本书选定 Unity 作为游戏开发的引擎，介绍游戏设计与开发的过程。

编写目的

本书主要面向对动漫、游戏设计有兴趣的读者或交互媒体方向、计算机软件开发相关专业的读者，具有较广的读者群体。编写本书的主要目的是普及游戏设计与开发的知识与流程，通过案例教学引导读者参与游戏设计的全过程，培养读者的计算思维与创新能力，帮助读者具备独立设计与开发游戏的基本素养。

编写思想

CDIO 是本书编写的主要指导思想。CDIO 是由美国麻省理工学院发起的，代表的是构思 (Conceive)、设计 (Design)、实现 (Implement)、运作 (Operate)，是最近几年工程教育改革的最新成果；以 CDIO 为框架将使得本书在实践型的教学与创作中更有指向性。

主要内容

本书的主要内容包括 Unity 软件的基本操作和实践案例两部分，涵盖手机游戏开发、2D 多场景游戏开发、3D 多场景游戏开发等单元。全书由基础篇和项目篇两个部分组成，分为 13 章。其中，基础篇包括第 1~9 章，内容为环境准备、Unity 的开发环境、熟悉游戏物体和组件、

创建 3D 游戏场景、物理系统、2D 动画、人形动画、UGUI 系统、粒子系统等 9 个相对独立的 Unity 技术章节；项目篇包括第 10~13 章，内容为兔子快跑、开心农场、寻找消失的大洲、保卫碉楼等 4 个综合型的游戏开发案例。

目标读者

本书可作为高等院校或职业院校游戏设计相关专业（如教育技术、数字媒体技术、数字媒体艺术、计算机软件等专业）的教材。适合的教学方法主要有案例教学、任务驱动教学等，引导读者了解游戏开发逻辑，逐步掌握操作要领，激发自我动手能力；并通过生活中游戏主题的选择，激发读者的创新意识与能力，引导学有余力的读者将课程设计选题同未来创新创业结合起来开展实践。

本书的知识点均使用浅显的描述，结合丰富且具体的配图，实现真正的“无门槛”学习。随着在线学习的兴起，本书采用任务式、项目化编制方法，也适合读者依托 MOOC 课程进行混合学习，编者将在后续的工作中推出本书配套的 MOOC 课程，以更好地为广大读者服务。

资源说明

本书为 Unity 的技术使用教材，着重介绍 Unity 引擎的使用、代码的编写及工程的设计，其提供的图片、模型等资源仅作为演示案例的需要，不作为可销售部分。本书提供部分素材资源的目的是便于读者学习，其所有内容（包括但不限于素材、资源和工程代码）不可作为任何目的与形式的商业或正式出版用途。

本书案例编写的主要素材和资源为原创。但由于案例较为丰富，引用了部分第三方的素材和资源，仅用于学习和演示、测试项目编码，在此特别致谢。若读者以其他目的使用这些素材和资源，请联系原素材和资源的版权方。部分素材资源来源如下：

(1) 第 4 章中房子模型资源来源：

CG 模型王 <http://www.cgmhxw.com/thread-7021-1-1.html>

(2) 部分模型资源从 Asset Store 中获取，列表如下：

“Unity-Chan!” Model

Fantasy Skybox FREE

Gorilla Character

Golden Tiger

Butterfly with Animations

Imperial Penguin

Rhino Cartoon

Elephant Cartoon

Fantasy Horde-Barbarians

(3) 第 11 章部分图片资源来源：

我图网 https://weili.ooopic.com/weili_15901694.html

在使用本书的过程中，若发现素材资源等出处有疑问之处，请联系作者或出版社，我们将妥善处理。

资源下载

本书课件和素材资源分享地址如下：



课件



素材资源

致谢

在本书的编写过程中，郑琳怡辅助编者完成了大量文稿的整理和文字的编写工作，对本书的完成起到了非常重要的作用；林卓泉、简海鹏、蔡诗毓为本书提供了教学案例，林丽明、姚辉锋帮助编者进行了校稿，感谢以上几位为本书做出的贡献。

信息反馈

由于编者精力、水平及编写时间所限，书中难免会有疏漏的地方，也恳请广大读者批评指正。若有勘误等相关书籍问题，请联系出版社，邮箱为 wkservice@163.com。

编 者

2019 年 5 月





目 录

基础篇

第1章 环境准备	2
1.1 认识 Unity	2
1.2 Windows 平台下的环境准备	3
1.3 MacOS 平台下的环境准备	6
1.4 资源链接	7
第2章 Unity 的开发环境	8
2.1 打开项目工程	9
2.2 常用界面	10
2.2.1 常用工作窗口	10
2.2.2 工具栏	18
2.2.3 菜单栏	24
2.3 基本操作	27
2.4 思考练习	33
第3章 熟悉游戏物体和组件	34
3.1 游戏物体	36
3.1.1 创建游戏物体	37
3.1.2 获取游戏物体	39
3.1.3 改变游戏物体的 Transform	40
3.1.4 克隆游戏物体	44
3.1.5 销毁游戏物体	45
3.2 组件	46
3.2.1 添加组件	46
3.2.2 获取组件	47
3.2.3 修改组件	48
3.2.4 移除组件	49
3.2.5 禁用组件	49
3.3 案例实战——Alphabet Game	49
3.3.1 游戏介绍与分析	49
3.3.2 前期准备	50
3.3.3 制作字母卡片 A	50
3.3.4 制作预制体 A	51
3.3.5 生成预制体 A	52
3.3.6 制作所有字母的预 制体	53
3.3.7 生成所有字母的预 制体	54
3.3.8 添加背景	56
3.3.9 完善游戏细节	58
3.4 思考练习	59
第4章 创建 3D 游戏场景	60
4.1 打开项目工程	61
4.2 地形编辑器	62
4.2.1 地形参数设置	63
4.2.2 绘制高度	63
4.2.3 升降高度	64

4.2.4 平滑高度	65
4.2.5 绘制贴图	66
4.2.6 放置树木	67
4.2.7 绘制细节	68
4.3 水效果	68
4.4 天空盒子	69
4.5 雾效	72
4.6 使用外部的 3D 模型	73
4.7 制作简单的第一人称漫游	76
4.8 思考练习	78
4.9 资源链接	78

第 5 章 物理系统 79

5.1 基础知识	80
5.2 刚体 (Rigidbody) 组件	80
5.3 碰撞体 (Collider) 组件	84
5.4 角色控制器 (Character Controller) 组件	90
5.5 射线 (Ray)	93
5.6 思考练习	96
5.7 资源链接	97

第 6 章 2D 动画 98

6.1 给游戏物体添加简单的动画效果	99
6.2 制作 Sprite Animation	103
6.3 思考练习	111
6.4 资源链接	111

第 7 章 人形动画 112

7.1 了解人形动画模型的构成	113
7.2 模型导入设置	117
7.3 制作动画状态机	119
7.4 第三人称漫游	122
7.5 动画重定向	125
7.6 思考练习	126
7.7 资源链接	126

第 8 章 UGUI 系统 127

8.1 了解基础知识	128
8.2 Canvas(画布)	130
8.2.1 Canvas(画布) 组件	130
8.2.2 Canvas Scaler(画布缩放器) 组件	135
8.2.3 Graphic Raycaster(图像射线) 组件	137
8.3 Image(图片)	137
8.4 Raw Image(原始图片)	140
8.5 Text(文本)	141
8.6 Button(按钮)	142
8.7 InputField(输入框)	146
8.8 Toggle(切换开关)	147
8.9 Slider(滑块)	149
8.10 Scroll View(滚动视图)	150
8.10.1 Scrollbar(滚动条)	150
8.10.2 Scroll Rect(滚动矩形)	151
8.10.3 Mask(遮罩)	152
8.11 Dropdown(下拉列表)	153
8.12 RectTransform(矩形变换)	154
8.12.1 Pivot(轴心点)	155
8.12.2 Anchor(锚点)	155
8.13 思考练习	156
8.14 资源链接	157

第 9 章 粒子系统 158

9.1 Particle System 组件	159
9.2 烟雾特效	165
9.3 繁星特效	169
9.4 礼花特效	174
9.5 思考练习	179
9.6 资源链接	179

项目篇

第 10 章 平台游戏——兔子

快跑 182

10.1 游戏构思与设计	183
10.1.1 游戏流程分析	183
10.1.2 游戏脚本	183
10.1.3 知识点分析	183
10.1.4 游戏流程设计	184
10.1.5 游戏元素及场景 设计	184
10.2 游戏开发过程	186
10.2.1 资源准备	186
10.2.2 搭建场景	186
10.2.3 制作兔子的动画	190
10.2.4 兔子的状态程序 设计	197
10.2.5 兔子的总体控制程序 设计	199
10.2.6 摄像机跟随	203
10.2.7 制作乌龟动画	204
10.2.8 乌龟的状态程序 设计	204
10.2.9 UI 制作：开始，胜利， 失败	205
10.2.10 游戏总控制程序 设计	210
10.2.11 项目输出与测试	213

第 11 章 种植游戏——开心

农场 215

11.1 游戏构思与设计	217
11.1.1 游戏流程分析	217
11.1.2 游戏脚本	217
11.1.3 知识点分析	217
11.1.4 游戏流程设计	217

11.1.5 游戏元素及场景

设计 218

11.2 游戏开发过程	218
11.2.1 资源准备	218
11.2.2 搭建自然场景	220
11.2.3 添加场景中的其他 物品	226
11.2.4 游戏界面设计	237
11.2.5 游戏逻辑设计	244
11.2.6 本地化存储	259
11.2.7 项目输出与测试	266

第 12 章 AR 游戏——寻找消失的

大洲 268

12.1 游戏构思与设计	269
12.1.1 游戏流程分析	269
12.1.2 游戏脚本	270
12.1.3 知识点分析	270
12.1.4 游戏流程设计	270
12.1.5 游戏元素及场景 设计	271
12.2 游戏开发过程	272
12.2.1 资源准备	272
12.2.2 搭建 Start 场景	272
12.2.3 搭建 Menu 场景	274
12.2.4 设置识别图和对应 显示的物体	277
12.2.5 显示大洲简介	278
12.2.6 在拼图里显示不同大洲 的动植物图片	282
12.2.7 项目输出与测试	285

第 13 章 塔防游戏——保卫

碉楼 287

13.1 游戏构思与设计	289
13.1.1 游戏流程分析	289



13.1.2 游戏脚本	289	13.2.4 敌人巡线系统	296
13.1.3 知识点分析	289	13.2.5 游戏交互	300
13.1.4 游戏流程设计	289	13.2.6 完善游戏 (天空盒子、 音效、小地图)	310
13.1.5 游戏元素及场景 设计	290	13.2.7 项目输出与测试	314
13.2 游戏开发过程	291	附录 EasyAR 的使用 316	
13.2.1 资源准备	291	参考文献 320	
13.2.2 搭建场景	291		
13.2.3 搭建 Player	293		

• 基础篇 •

第1章 环境准备

第2章 Unity 的开发环境

第3章 熟悉游戏物体和组件

第4章 创建 3D 游戏场景

第5章 物理系统

第6章 2D 动画

第7章 人形动画

第8章 UGUI 系统

第9章 粒子系统

第 1 章



环境准备

作为一款流行的游戏开发引擎，Unity 以便捷高效、效果精美的特点受到众多开发者喜爱。同时，其安装开发环境也非常简单。在本章中，我们将在了解 Unity 的基本背景后，着手安装 Unity 开发环境。

1.1 认识 Unity

Unity 是由 Unity Technologies 公司开发的专业跨平台游戏开发及虚拟现实引擎，工作流程精简直观，功能强大，能够轻松完成各种游戏创意和三维互动开发。通过 3D 模型、图像、视频和声音等相关资源的导入，借助 Unity 内置相关的场景构建模块，用户可以轻松地创建复杂的虚拟世界。

Unity 编辑器可以在 Windows、MacOS X 平台运行，支持发布的平台有 21 种，其中包括 iOS、Android、Windows、Wii 等，用户只需一次开发，即可部署到相应的平台，极大地缩短了开发周期并节省人力。

在虚拟现实行业和游戏行业中，Unity 日益受到开发者的青睐，Unity 引擎开发的移动游戏、网页游戏和端游层出不穷。大家耳熟能详的有《奥日和黑暗森林》(如图 1-1 所示)、《纪念碑谷》(如图 1-2 所示)、《炉石传说》(如图 1-3 所示)等。

快速开发，快速实现游戏构想，Unity 是不二之选。



图 1-1 《奥日和黑暗森林》

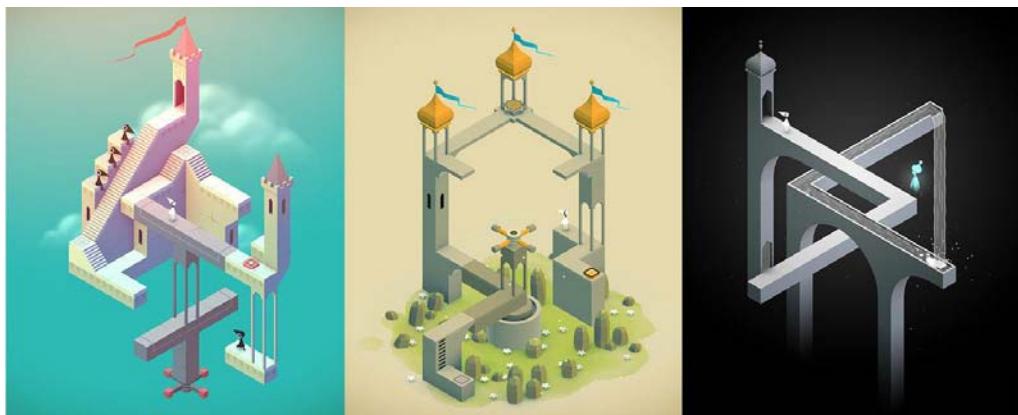


图 1-2 《纪念碑谷》



图 1-3 《炉石传说》

1.2 Windows 平台下的环境准备

登录 Unity 官方下载地址 <https://store.unity.com/cn>，进入 Unity 下载页面，如图 1-4 所示。初学者下载使用个人版即可。

拖动到网页末尾，如图 1-5 所示，单击“Unity 旧版本”链接，可以查看所有 Unity 版本。

选择所需要的版本，下载 Unity 安装程序。本书使用 Unity 5.6.3 版本，如图 1-6 所示。Unity 引擎是向下兼容的，即高版本编辑器可以打开低版本工程，但是要注意的是，低版本编辑器不可以打开高版本开发的工程。

安装程序下载完成后，双击即可打开 Unity Download Assistant 安装对话框，如图 1-7 所示。单击 Next 按钮进入 License Agreement(许可协议) 对话框，如图 1-8 所示。选中 I accept the terms of the License Agreement 复选框，接受许可协议。



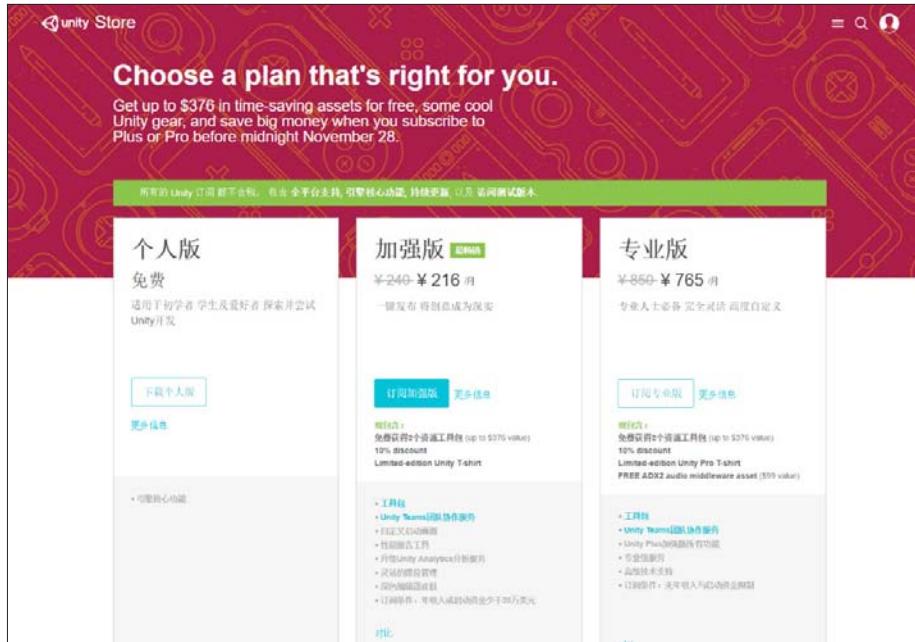


图 1-4 Unity 下载页面



图 1-5 Unity 下载页面的页脚

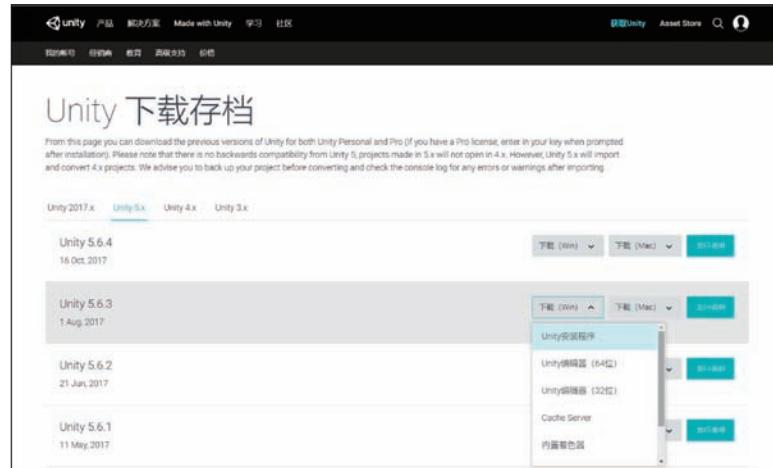


图 1-6 选择 Unity 下载版本

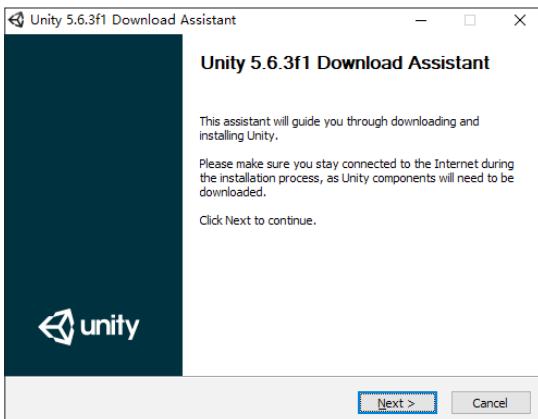


图 1-7 Unity Download Assistant 安装对话框

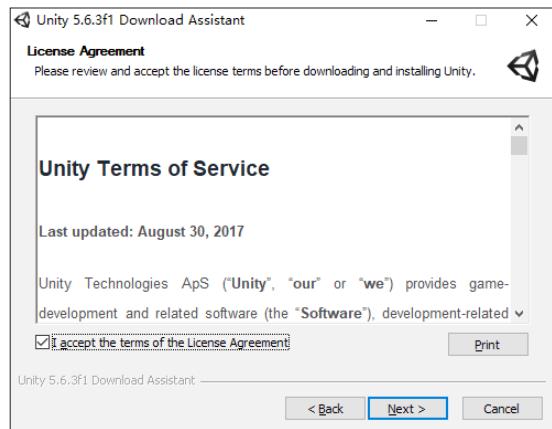


图 1-8 接受许可协议

单击 Next 按钮进入版本位数选择对话框，如图 1-9 所示，根据计算机的位数选择相应的版本，这里选择 64 bit 单选按钮。

单击 Next 按钮进入组件选择对话框，如图 1-10 所示。在此对话框中可以有选择地安装 Unity 开发组件。其中，Unity 主程序是必选的，用户可根据开发需求选择安装其他组件。

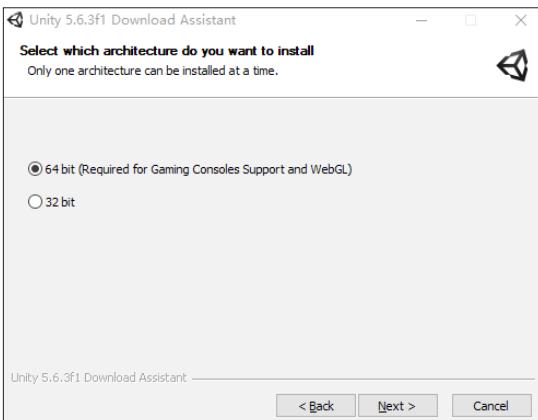


图 1-9 选择版本位数

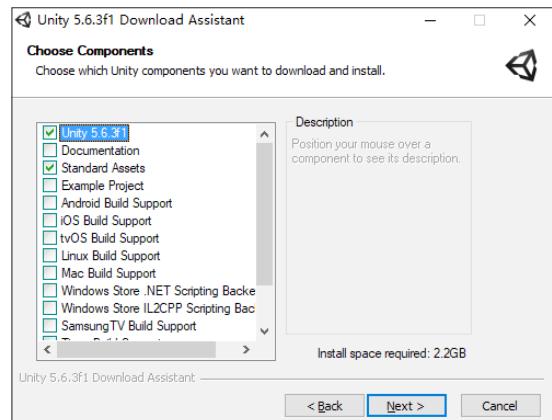


图 1-10 选择组件

常用的有 Standard Assets(Unity自带的资源)、Example Project(示例项目)，如果需要发布安卓应用，需要选中 Android Build Support 复选框。如果不知道是否需要安装哪些组件，以后也可以根据需要再打开该安装程序来选择安装。

单击 Next 按钮，进入选择安装路径对话框，如图 1-11 所示，可以使用默认的安装路径，也可以单击 Browse 按钮选择其他安装路径。

单击 Next 按钮，开始下载 Unity 相关安装包。等待一段时间，安装完成后出现安装完成窗口，单击 Finish 按钮即可完成安装。

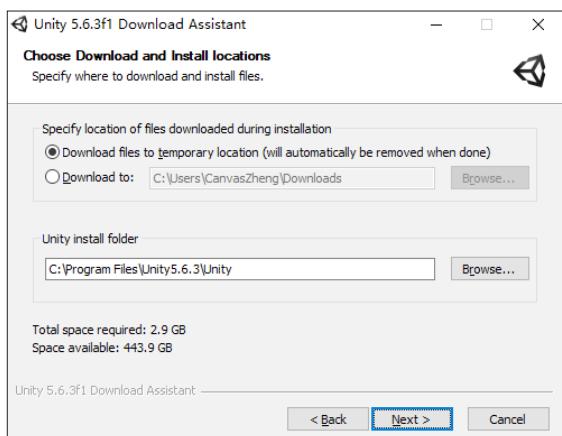


图 1-11 选择安装路径

1.3 MacOS 平台下的环境准备

和 Windows 环境一样，在 Mac OS 平台下安装 Unity，首先进入版本选择页面。Unity 会自动识别计算机操作系统，直接选择想要的版本即可下载 .dmg 文件。

双击下载的 dmg 安装包，打开“Unity 下载助手”界面，如图 1-12 所示。双击 Unity Download Assistant 按钮，打开 Download And Install Unity 对话框，可看到 Unity 安装介绍，如图 1-13 所示。

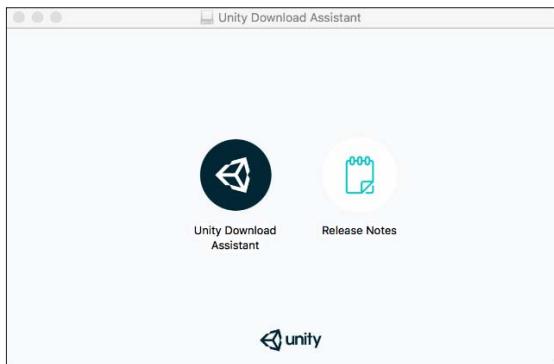


图 1-12 Unity 下载助手界面

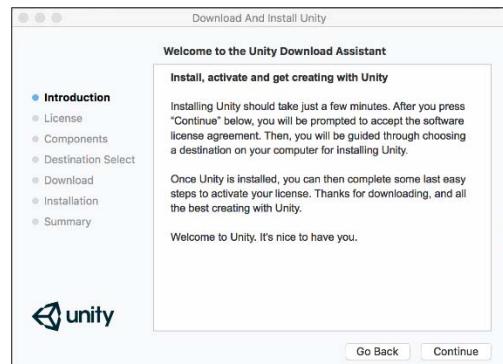


图 1-13 Download And Install Unity 对话框

单击 Continue 按钮，弹出对话框提示用户确认许可协议，如图 1-14 所示。

单击 Agree 按钮，进入开发组件选择对话框，选择所需要的 Unity 开发组件，如图 1-15 所示。

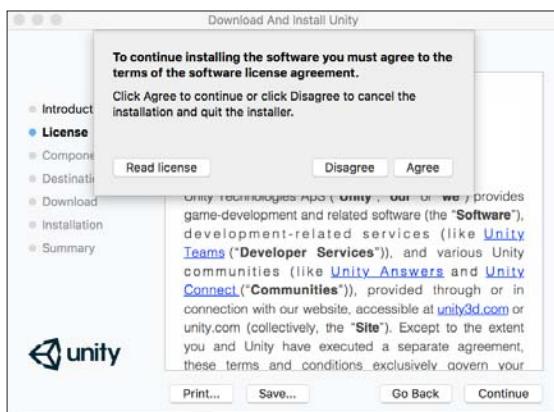


图 1-14 接受许可协议

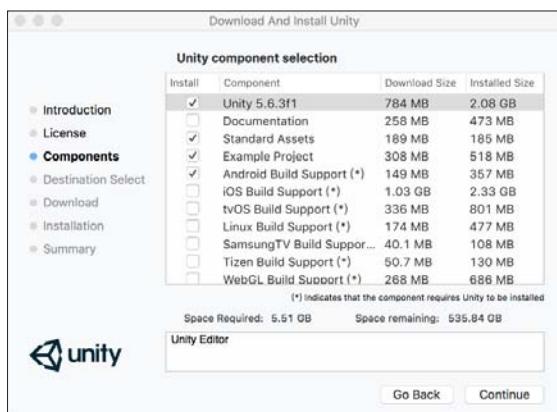


图 1-15 选择开发组件

单击 Continue 按钮，进入选择安装路径对话框，选择一个硬盘空间安装 Unity，如图 1-16 所示。

单击 Continue 按钮，在弹出的对话框中输入用户名和密码，如图 1-17 所示。然后等待下载完成，一直单击 Continue 按钮即可完成安装。

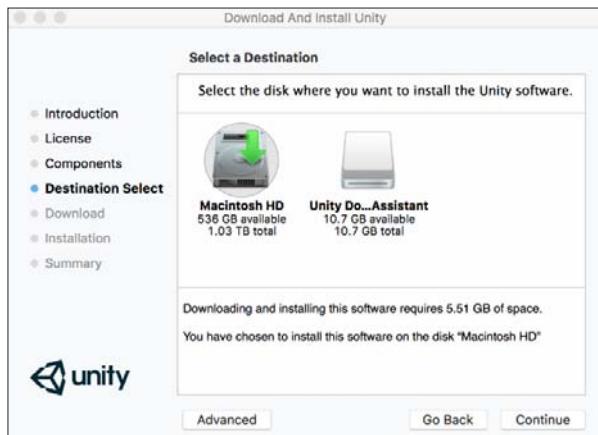


图 1-16 选择安装路径



图 1-17 输入用户名和密码

提示：

用户可根据项目，选择安装需要的组件即可。其中，Unity 运行程序是必需的；本书操作过程中需要使用标准资源，对初学者非常有用；示例工程也建议大家安装来学习。若开发安卓应用或苹果应用，则需要安装相应的开发组件。

不必一次性全部安装，在以后的学习过程中，可根据需要再次打开选择安装相关开发组件。具体组件说明如图 1-18 所示。

组件说明	Component	Download Size	Installed Size
Unity运行程序（必须）	Unity 5.6.3f1	784 MB	2.08 GB
使用文档	Documentation	258 MB	473 MB
标准资源（方便初学者快速开发）	Standard Assets	189 MB	185 MB
示例工程（观摩学习）	Example Project	308 MB	518 MB
开发安卓应用必备	Android Build Support (*)	149 MB	357 MB
开发苹果iOS应用必备	iOS Build Support (*)	1.03 GB	2.33 GB
开发AppleTV应用必备	tvOS Build Support (*)	336 MB	801 MB
开发Linux应用必备	Linux Build Support (*)	174 MB	477 MB
开发SamsungTV应用必备	SamsungTV Build Suppor...	40.1 MB	108 MB
开发Tizen应用必备	Tizen Build Support (*)	50.7 MB	130 MB
开发WebGL应用必备	WebGL Build Support (*)	268 MB	686 MB
开发Windows应用必备	Windows Build Support (*)	200 MB	741 MB
开发Facebook-Games应用必备	Facebook Gameroom Buil...	39.4 MB	89.4 MB

图 1-18 组件说明

1.4 资源链接

Unity 下载相关网站如下。

- Unity 官方网站：<https://unity3d.com/cn>
- Unity 下载网址：<https://unity3d.com/cn/get-unity/download>



第 2 章

Unity 的开发环境

上一章搭建了 Unity 的开发环境,本章使用 Unity 的范例工程来熟悉一下 Unity 的开发环境。

【学习目标】

1. 熟悉 Unity 编辑器,了解 Unity 的工作窗口、工具栏和菜单栏。
2. 掌握 Unity 开发的基本操作,学会使用 Unity 开发一个演示案例。

【知识点说明】

本章的知识点结构如图 2-1 所示。

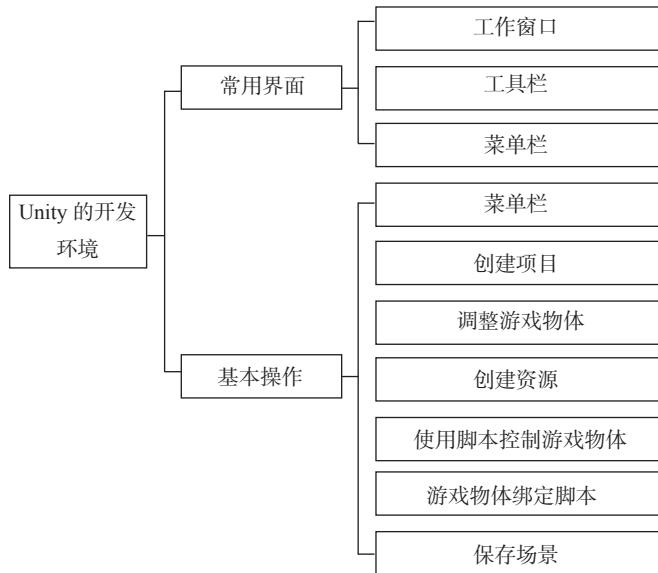


图 2-1 本章知识点结构

【任务说明】

本章任务及对应的知识点如表 2-1 所示。

表 2-1 任务及对应的知识点

任务	知识点
了解 Unity 的工作界面	常用的工作窗口、菜单栏、工具栏
掌握 Unity 的基本操作	创建项目、创建游戏物体、创建资源等

2.1 打开项目工程

双击打开 Unity，选择名为 Standard Assets Example Project 的工程文件，如图 2-2 所示。

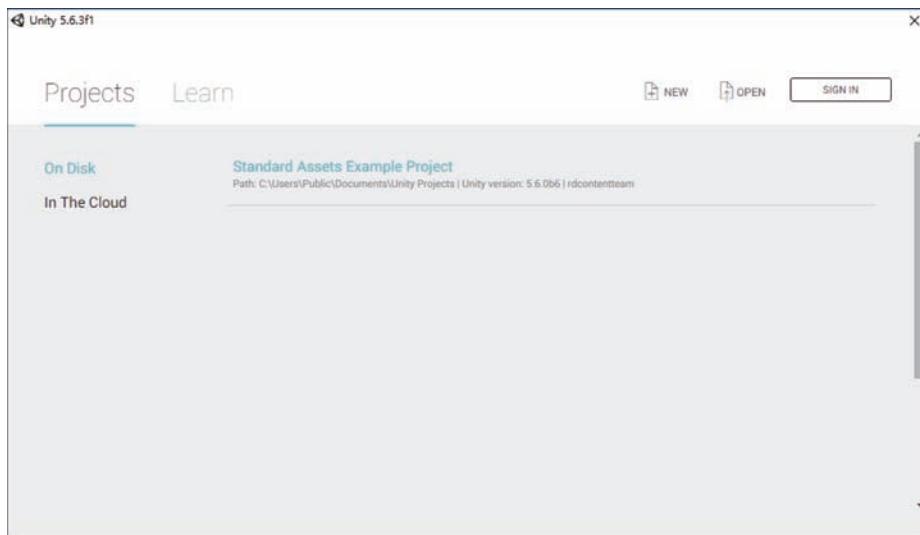


图 2-2 Unity 选择项目工程

打开该工程文件之前，会弹出一个警告对话框。因为这个工程是用 5.6.0 版本的编辑器开发的，而我们的编辑器是 5.6.3 版本。**高版本编辑器打开低版本的工程文件，都会弹出警告对话框**，如图 2-3 所示。单击 Continue 按钮即可继续打开。

打开后的工程界面如图 2-4 所示，这是默认的工作界面布局 (Default Layout)。

界面最上方是**菜单栏**，能够对编辑器和工程项目进行操作。第二行是**工具栏**，能辅助用户搭建场景。中间是编辑游戏场景的区域，可以自由创建 3D 世界，叫作**Scene** 窗口。

和 Scene 窗口并排的选项卡还有**Game** 和**Asset Store** 窗口。单击 Game 选项卡，会显示 Scene 中的某个位置，是 Scene 中的 Main Camera(主摄像机) 拍到的呈现给玩家的画面。Asset Store 是 Unity 的资源商店，注册并登录自己的 Unity 账号，能够从中下载免费或付费的各类丰富资源。

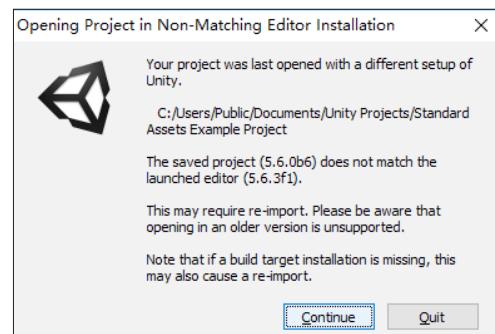


图 2-3 版本不兼容警告



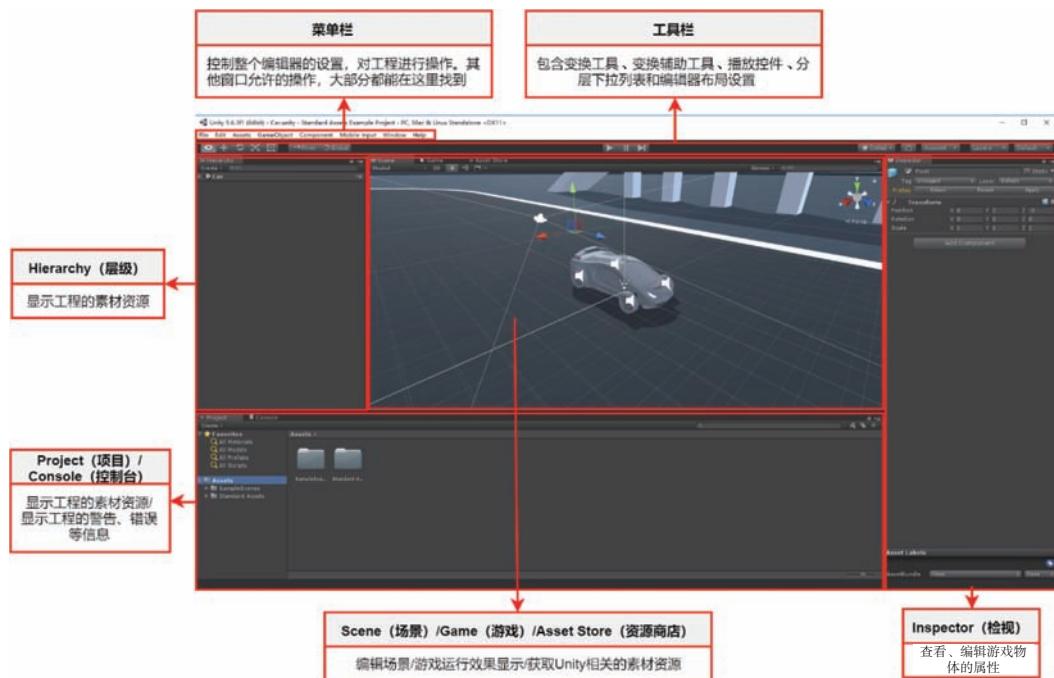


图 2-4 Unity 编辑器工作界面

左边是 **Hierarchy(层级)** 窗口，是 **Scene** 窗口中的游戏物体的列表表现形式。**Scene** 窗口中存在的游戏物体，都与 **Hierarchy** 窗口中的某个名称一一对应。在 **Hierarchy** 窗口中双击某个游戏物体的名称，能迅速在 **Scene** 窗口中找到该物体。

右边是 **Inspector(检视)** 窗口。在 **Hierarchy** 窗口或 **Scene** 窗口中选择某个游戏物体，其详细信息属性会罗列在 **Inspector** 窗口中。

下方是 **Project(项目)** 窗口，是游戏资源素材存放的地方。当我们从外界获取了资源，先存放到这里，在场景中需要时，再将其拖到场景中去。

与 **Project** 窗口选项卡并排的是 **Console(控制台)** 窗口，用于显示游戏开发过程中的错误或警告信息，方便开发者找到错误，辅助开发者了解游戏运行情况。

接下来详细介绍 Unity 的工作界面。

2.2 常用界面

2.2.1 常用工作窗口

1. Project 窗口

Project(项目) 窗口是存放工程资源的地方，这些资源可以是来自 Unity 之外创建的文件，如 3D 模型、图片、音频等，还可以是 Unity 创建的一些文件，如脚本、Animator、Prefab 等。

Project 窗口由 Create 菜单、Search by Type(按类型搜索) 菜单、Search by Label(按标签搜索) 菜单、搜索栏和资源显示区域等部分组成，如图 2-5 所示。

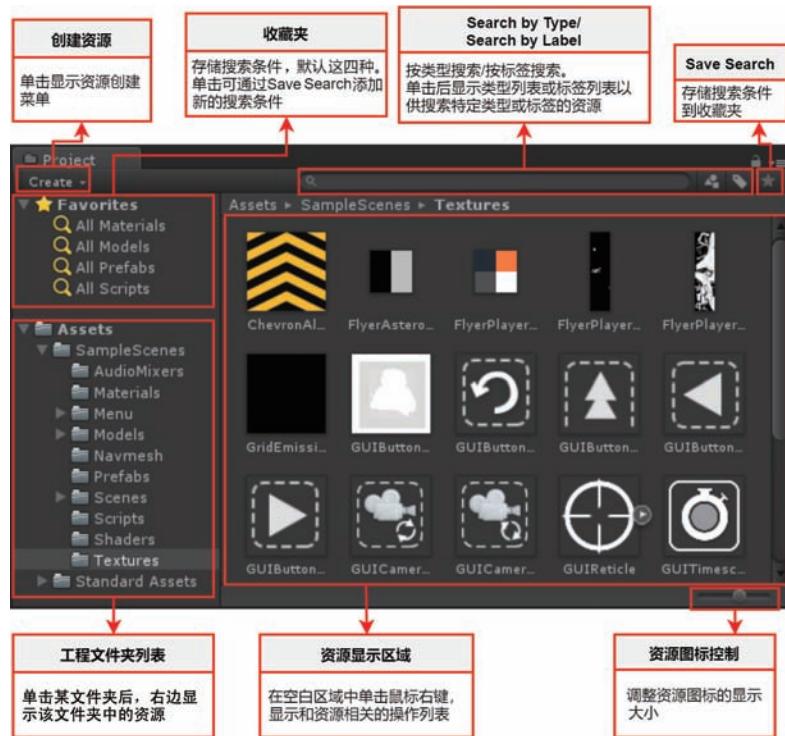


图 2-5 Project 窗口

(1) 导入资源。每个 Unity 工程都会包含一个 Assets 文件夹，所有创建的资源或导入的资源都会存放在里面。单击左边的项目文件夹，右边会显示该文件夹下的资源。可以直接将资源拖入 Project 窗口中；或在该窗口中右击鼠标，在弹出的快捷菜单中选择 Import New Asset(导入新资源) 命令依次导入。

(2) 查找资源。当资源过多无法通过逐个浏览文件夹寻找资源时，可以在搜索栏输入资源名称快速找到资源。如果想要寻找某个类型或某个标签的资源，可以通过过滤查找。如图 2-6 所示，输入 t:Animation，资源显示框中即可显示所有 Animation 类型的资源。“t:”表示类型过滤，“l:”表示标签过滤。

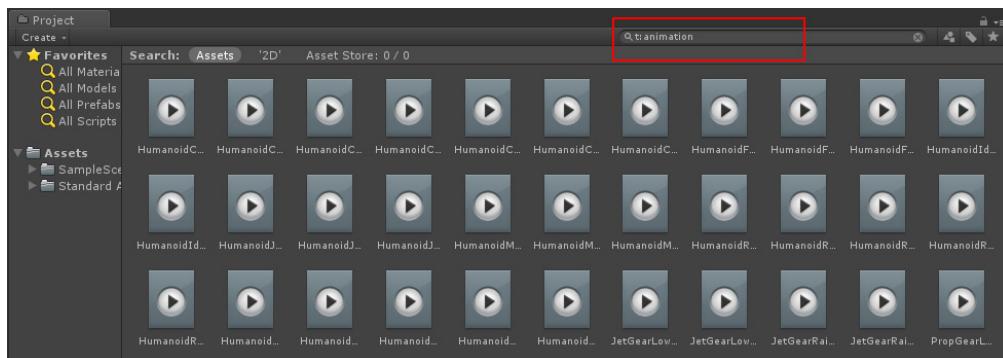


图 2-6 在搜索栏输入资源类型后的搜索结果

也可以直接通过单击搜索栏右边的 Search by Type(按类型搜索) 按钮搜索，如图 2-7 所示。

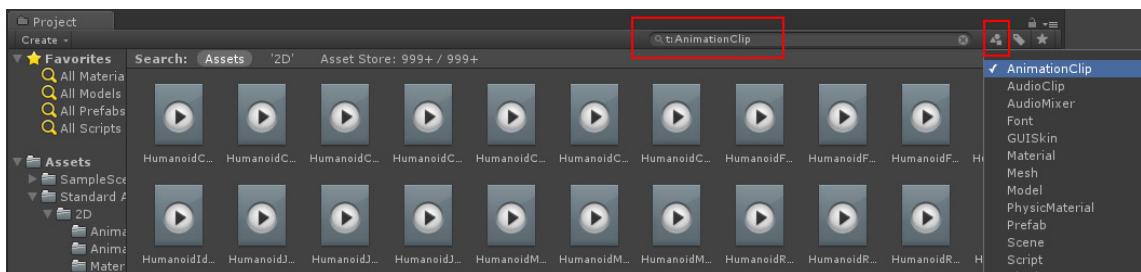


图 2-7 单击 Search by Type(按类型搜索) 按钮的搜索资源示意图

2. Hierarchy 窗口

Hierarchy(层级) 窗口只显示当前场景中的所有游戏物体，如图 2-8 所示。开发者通过对游戏物体命名区分不同的物体，双击游戏物体的名称即可在 Scene 窗口中聚焦到该游戏物体。

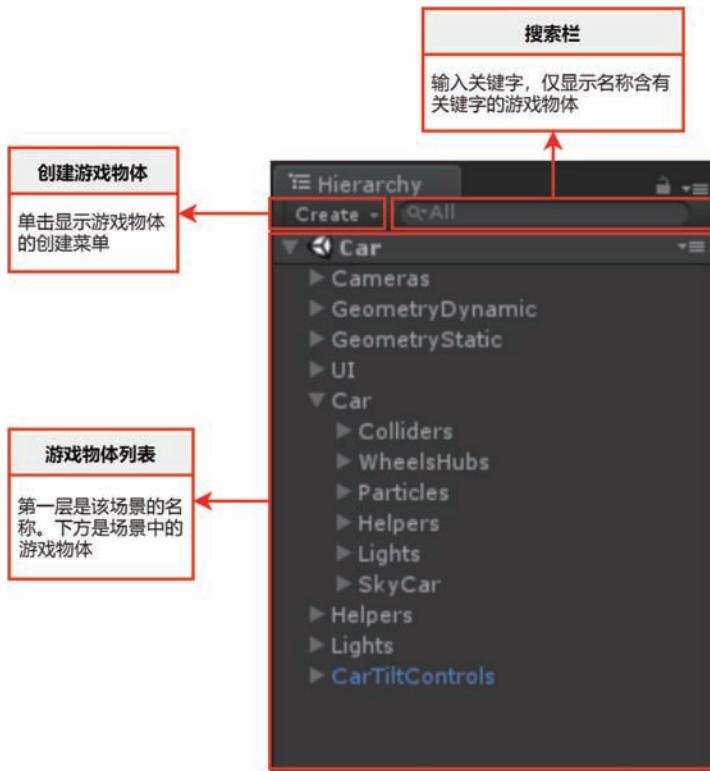


图 2-8 Hierarchy 窗口

在 Hierarchy 窗口中，通过拖动一个物体到另一个物体上，可以定义游戏物体的父子关系。一个游戏物体可以有多个子物体，但只能有一个父物体。对父物体的操作会影响其所有子物体。

3. Inspector 窗口

Inspector(检视) 窗口用于显示游戏场景中当前所选择对象的详细信息，包括对象的名称、标签，以及对象的各种组件等信息，如图 2-9 所示。



图 2-9 Inspector 窗口

每个游戏物体都会附加许多组件，添加、修改组件能使游戏物体表现不同的特性。所有组件右上方都有两个按钮：一个是帮助按钮，一个是设置按钮。图 2-10 所示是 Transform 组件的两个按钮。

单击设置按钮，弹出下拉菜单，可以看到对组件信息操作的选项，如图 2-11 所示。

设置按钮的下拉菜单主要功能如下。

- Reset: 重置组件的属性值，将所有属性还原成默认值。
- Move to Front: 将该游戏对象移到最上层，改变的是游戏对象的层级。
- Copy Component: 复制组件的值。复制的值只能粘贴到同类组件上。
- Paste Component As New: 将复制了的组件值粘贴到另一个没有该组件或可以有很多个同类组件的游戏物体中，作为一个新的组件。
- Paste Component Values: 将复制了的组件值粘贴到另一个游戏物体的同类组件中。

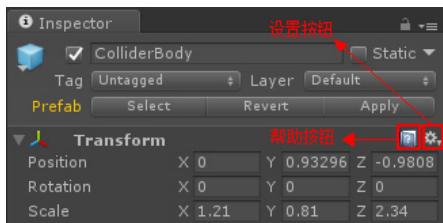


图 2-10 Transform 组件的帮助按钮和设置按钮

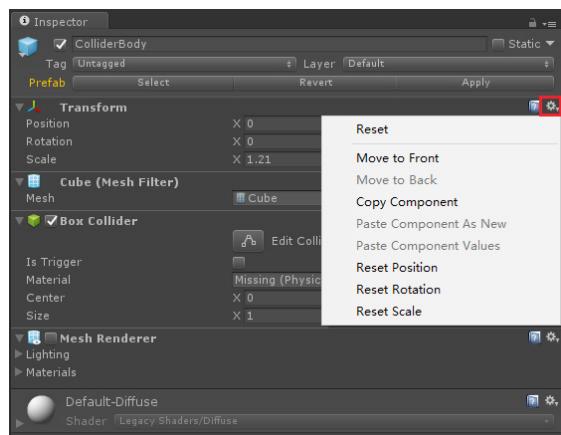


图 2-11 设置按钮的下拉菜单

4. Game 窗口

Game(游戏) 窗口是游戏的预览窗口，不能对场景进行编辑，用于呈现完整的游戏效果。

当单击播放按钮后，该窗口可以进行游戏的实时预览，方便调试开发，其主要功能如图 2-12 所示。

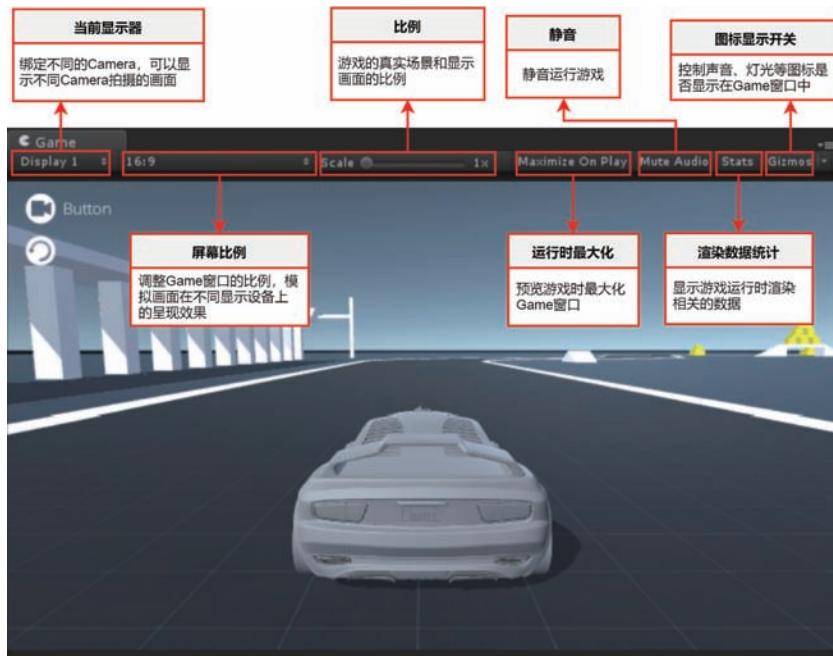


图 2-12 Game 窗口

5. Scene 窗口

Scene(场景) 窗口是最常用的窗口，场景中所用到的模型、光照、摄像机等游戏物体都会显示在此窗口，在此窗口可以对游戏对象进行编辑，如图 2-13 所示。可以使用以下快捷键进行快速操作。

- 旋转：按住 Alt 键，并拖动鼠标，以当前轴心点来旋转场景。该操作在 2D 模式下不可用。

- 移动：按住鼠标中键或按快捷键 Q，拖动游戏对象可进行移动。
- 缩放：滚动鼠标滚轮，或按住 Alt 键，并拖动鼠标右键，可以放大或缩小视角。
- 居中显示所选物体：按 F 键会在场景视图中居中显示该物体。要将移动的游戏物体锁定在视图中央，则按 Shift+F 键。
- 飞行模式：以第一人称视角漫游场景，可以按住鼠标右键并拖动可移动视图，同时按 W/A/S/D 键可向左 / 右 / 前 / 后移动，按 Q/E 键可向上 / 下移动，按 Shift 键加速移动。



图 2-13 Scene 窗口

下面对场景进行一些操作来熟悉场景窗口。

(1) 2D/3D 模式切换。单击 **2D**，将 Scene 窗口从 3D 模式切换到 2D 模式，如图 2-14 所示。在 2D 模式下，右上角的轴向控制器消失，因为 2D 模式下场景使用正交相机，物体与相机的距离不影响物体的显示，不需要轴向控制器。而 3D 模式下使用透视相机，物体根据近大远小的规则渲染。

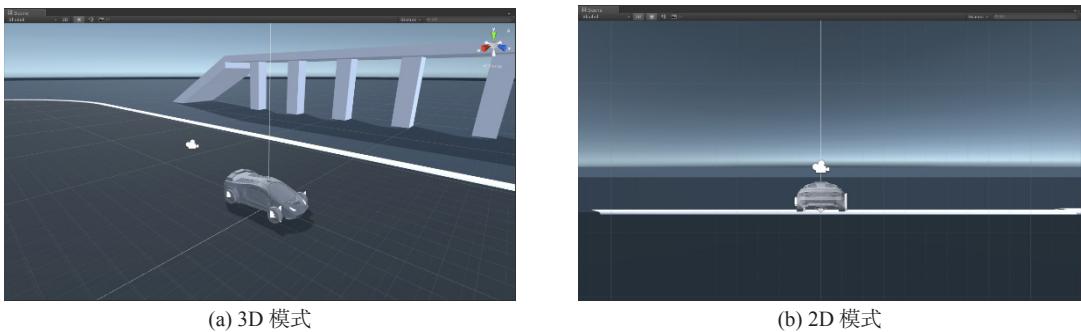


图 2-14 Scene 窗口从 3D 模式切换到 2D 模式

(2) 只显示包含某个名称的游戏对象。在搜索栏输入对象名称，这里输入 car，找到的对象显示本身的颜色，其他对象显示灰色，如图 2-15 所示。同时 Hierarchy 窗口也只显示带有 car 的对象，在实际开发中方便选择所有某个名称的游戏对象。

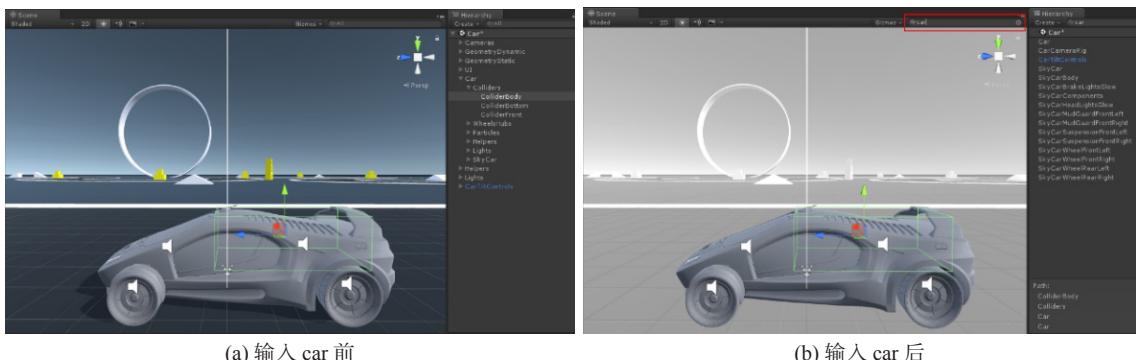


图 2-15 显示名称包含 car 的游戏对象

(3) 缩放视角。当按住 Alt 键，并单击鼠标右键时，鼠标箭头变成放大镜Q，此时可以滑动鼠标滚轮键，或者按住 Alt 键和鼠标右键并移动鼠标来缩放视角，如图 2-16 所示。

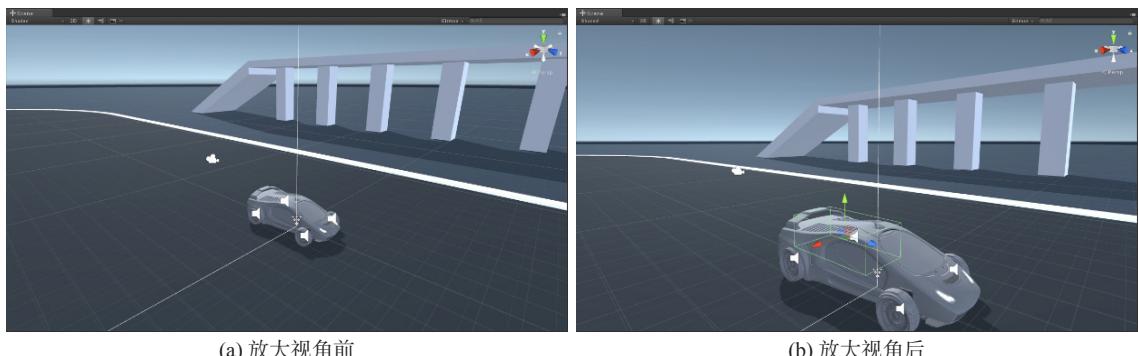


图 2-16 放大视角前后的 Scene 窗口

(4) 旋转视角。按下 Alt 键时，鼠标箭头变成眼睛O，此时按住 Alt 键，并拖动鼠标左键，即可旋转视角，从另一个角度观察游戏场景，如图 2-17 所示。

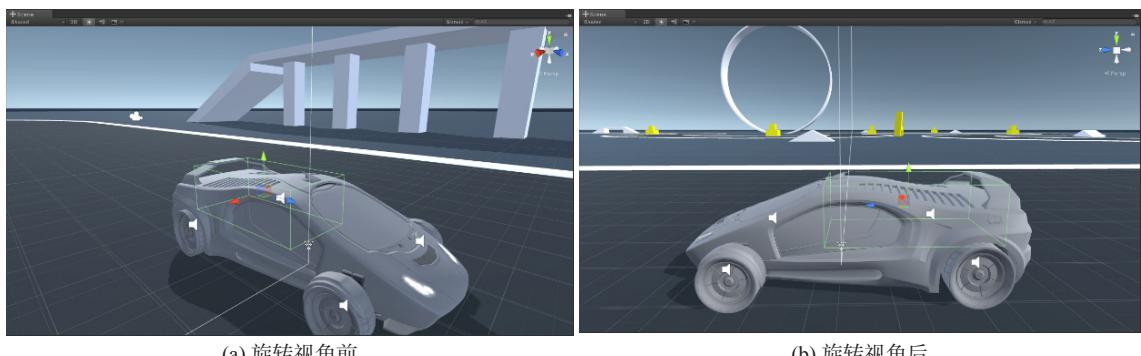


图 2-17 旋转视角前后的 Scene 窗口

(5) 移动视角。按住鼠标中键，或按下 Q 键（英文输入法下），鼠标箭头会变成手掌W，此时移动鼠标即可移动视角，如图 2-18 所示。

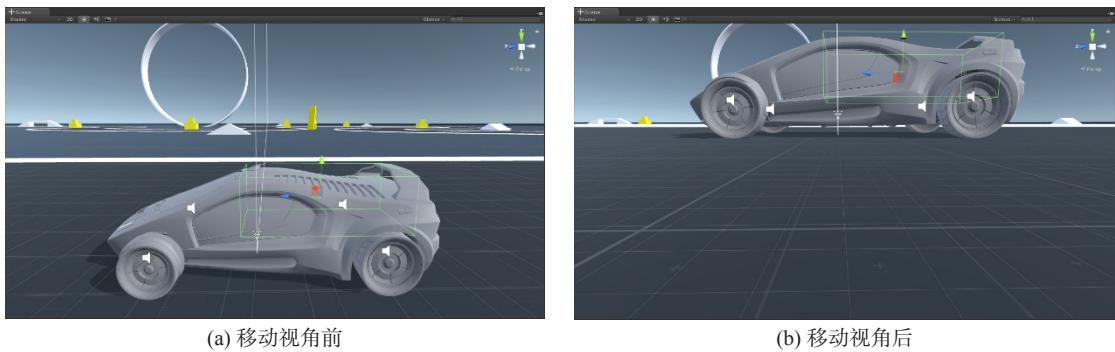


图 2-18 移动视角前后的 Scene 窗口

(6) 居中显示所选物体。选择远处的黄色方块，按下 F 键，可将所选择的游戏物体居中显示，如图 2-19 所示。

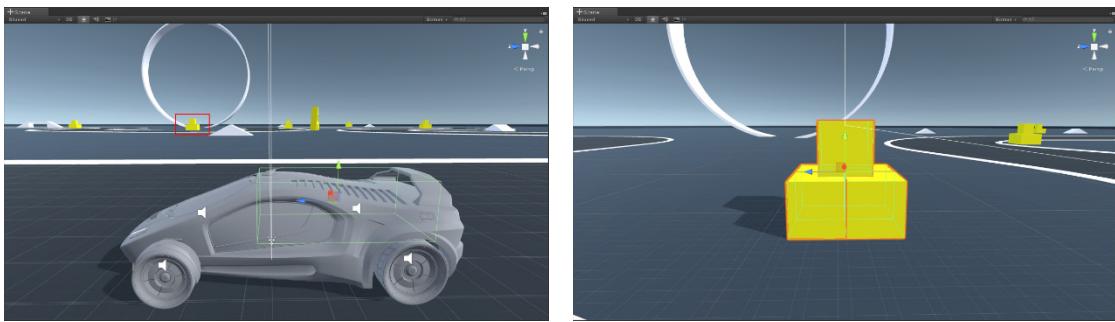


图 2-19 居中显示黄色方块前后的 Scene 窗口

(7) 飞行浏览模式 (Flythrough)。按住鼠标右键，鼠标箭头会变成眼睛和矩形 。此时按下 W/A/S/D/Q/E 键可在飞行模式下切换飞行方向，让用户以第一人称视角漫游场景。同时按 Shift 键可加速移动，如图 2-20 所示。

(8) 迅速切换顶 / 底 / 前 / 后视图。单击轴向控制器上的箭头可切换场景的视角：Top(顶视图)、Bottom(底视图)、Front(前视图)、Back(后视图)，如图 2-21 所示。

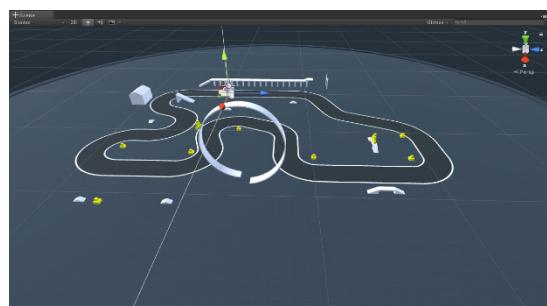


图 2-20 飞行浏览模式

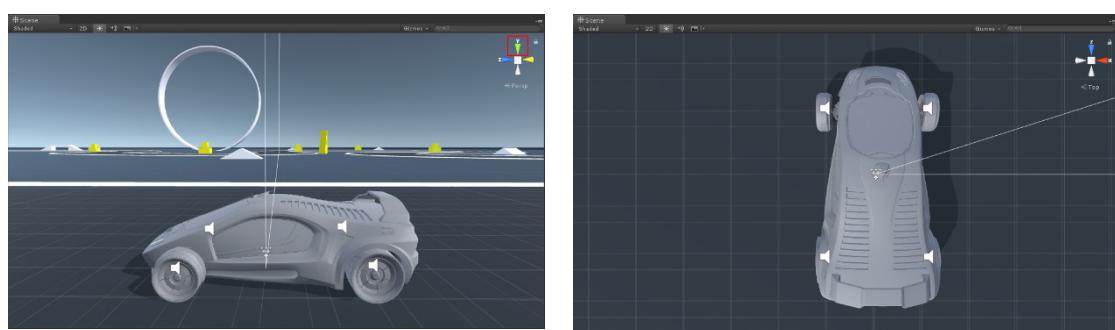


图 2-21 单击 Top 箭头切换到顶视图

单击轴向控制器中间的方块或者下方的文字，可以切换投影模式：Isometric Mode(等角投影模式)、Perspective Mode(透视模式)，如图 2-22 所示。

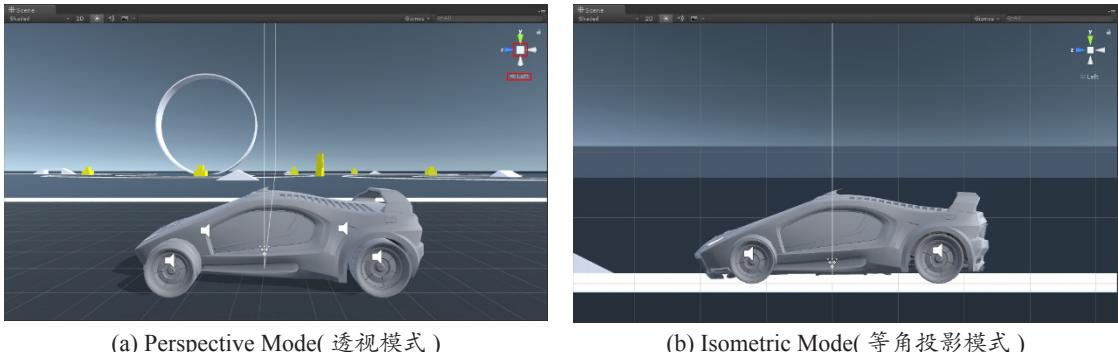


图 2-22 切换投影模式

6. Console 窗口

Console(控制台) 窗口是 Unity 中重要的调试工具，当用户测试项目或导出项目时，在 Console 窗口状态栏都会有相关的信息提示。通过双击错误信息，可以调出代码编辑器并定位到有问题的脚本代码位置，如图 2-23 所示。

要打开 Console 窗口，可以选择菜单栏中的 Window → Console，或者使用快捷键 Ctrl+Shift+C。



图 2-23 Console 窗口

2.2.2 工具栏

Unity 工具栏位于菜单栏的下方，主要由 5 个控制区域组成，它提供了常用功能的快捷访问方式。

工具栏主要包括以下部分：Transform Tools (变换工具)、Transform Gizmo Tools(变换辅助工具)、Play(播放控件)、Layers(分层下拉菜单) 和 Layout(布局下拉菜单)，如图 2-24 所示。



图 2-24 工具栏

1. Transform Tools

Transform Tools(变换工具) 有四种, 如图 2-25 所示。

- Translate(移动) 工具: 快捷键为 W, 改变游戏物体的位置。
- Rotate(旋转) 工具: 快捷键为 E, 改变游戏物体的旋转角度。
- Scale(缩放) 工具: 改变游戏物体的大小。
- Rect Transform(矩形变换) 工具: 是移动、缩放和旋转工具的组合。

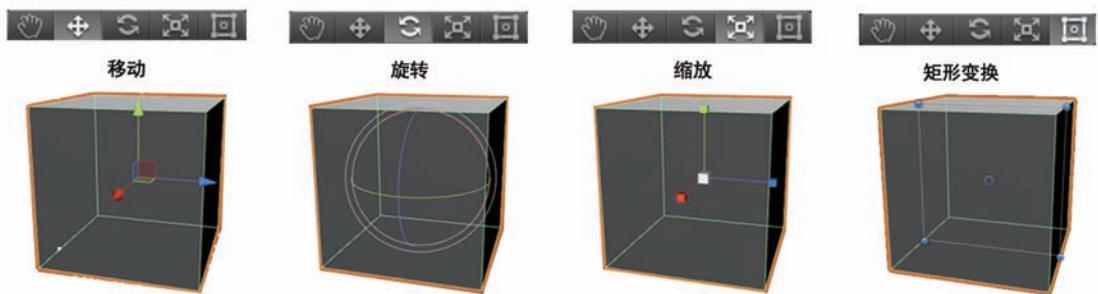


图 2-25 变换工具

这里通过以下操作熟悉变换工具的使用。

(1) 改变物体的位置 。选择某物体, 在 Scene 窗口中看到所选的物体上出现三维坐标轴, 拖动某个箭头即可改变该方向的位置, 对应改变的是 Inspect 窗口中的 Transform 组件的 Position 的值, 如图 2-26 所示。

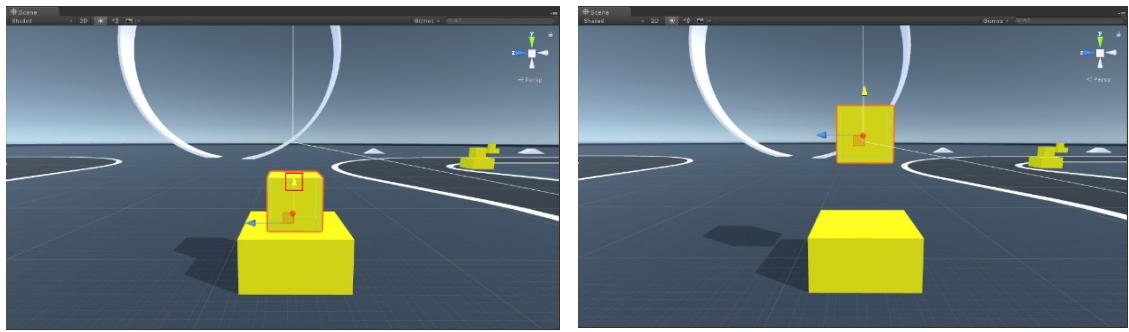


图 2-26 拖动三维坐标轴改变物体位置

(2) 改变物体的旋转角度 。选择某物体, 在 Scene 窗口中看到所选物体上出现不同颜色的圆圈。选择其中一个圆圈, 按住鼠标左键拖动即可改变某个方向的旋转角, 对应改变的是 Inspect 窗口中的 Transform 组件的 Rotation 的值, 如图 2-27 所示。

(3) 改变物体的大小 。选择某物体, 在 Scene 窗口中看到所选物体上出现三条线连接四色方块, 如图 2-28 所示。

其中，蓝色方块表示沿 Z 轴缩放，红色方块表示沿 X 轴缩放，黄色方块表示沿 Y 轴缩放。选择其中一个方块，按住鼠标左键拖动即可将物体沿某个轴缩放。中间的灰色方块可使物体在三个轴上等比缩放。对应改变的是 Inspector 窗口中的 Transform 组件的 Scale 的值。

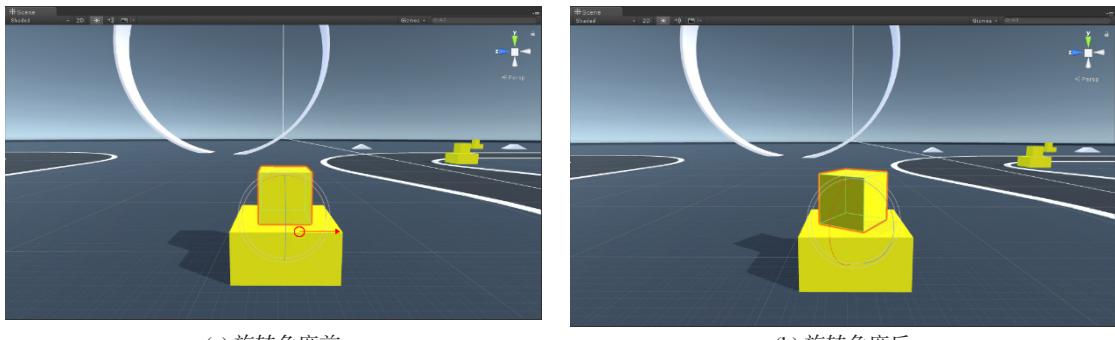


图 2-27 拖动旋转角度工具改变物体的放置角度

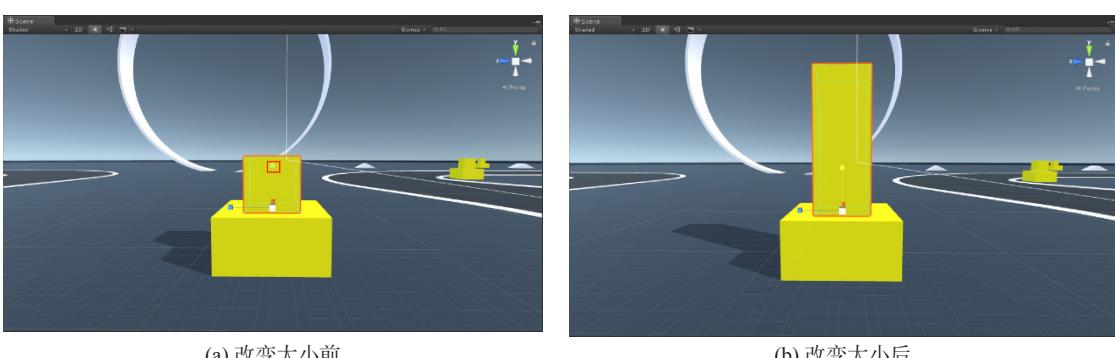


图 2-28 拖动缩放大小工具改变物体的大小

(4) 改变物体的位置、旋转角和大小

选择某物体，该物体在当前视角下的横截面出现矩形 Gizmos。在矩形 Gizmos 中单击并拖动可以移动游戏物体，单击并拖动矩形 Gizmos 的任何角或边可以缩放游戏物体。将光标放在矩形的某个角落之外，光标变成旋转图标，单击并拖动可以旋转游戏物体，拖动边缘可以沿一个轴缩放该物体，如图 2-29 所示。另外，可以通过按住 Shift 键等比缩放物体。

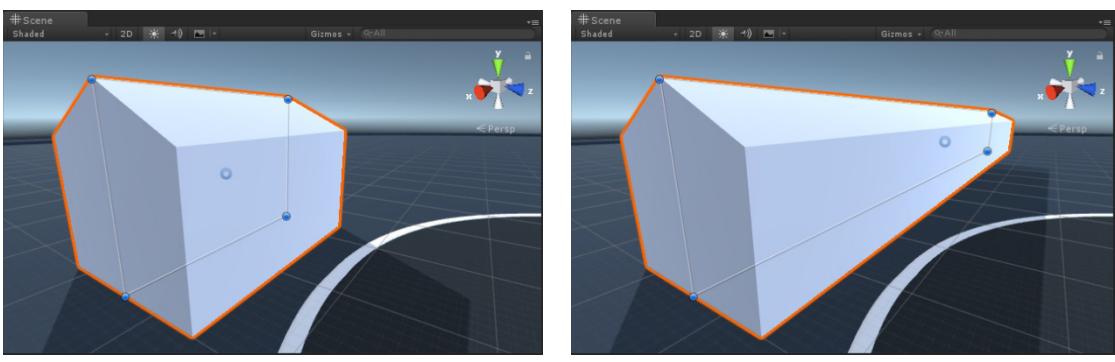
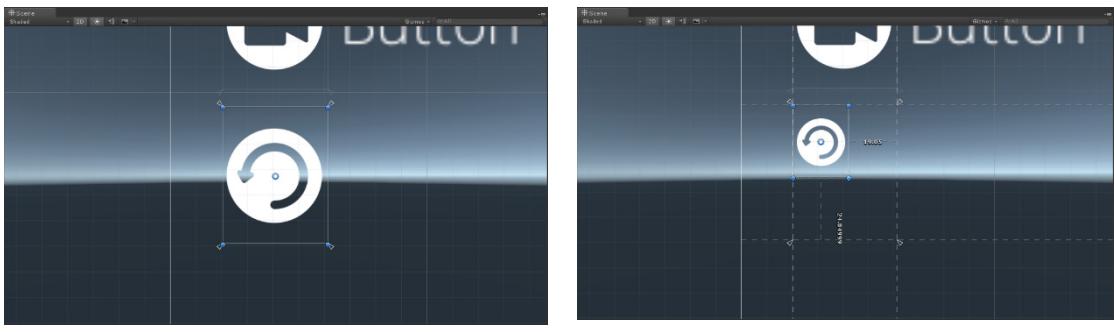


图 2-29 拖动矩形手柄改变物体的大小

矩形变换工具还通常用于定位 2D 游戏物体，如 UGUI 和 Sprite。2D 模式下无法改变 Z 轴，如图 2-30 所示。



(a) 使用矩形手柄前

(b) 使用矩形手柄 +Shift 键后

图 2-30 拖动矩形手柄改变 UGUI 的大小

2. Transform Gizmo Tools

Transform Gizmo Tools(变换辅助工具) 包括位置辅助工具和旋转辅助工具。

(1) 位置辅助工具，显示游戏物体的轴心参考点，常用于多物体的整体移动。

- Center(中心)：以所有选中物体所组成的轴心作为游戏物体的轴心参考点。
- Pivot(轴心)：以最后一个选中的游戏物体的轴心作为参考点。

我们通过以下例子来说明两者的区别。

场景中有两个物体，其中 Cube 是 Sphere 的父物体。若位置辅助工具是 Center 时，分别选中 Cube 和 Sphere 的坐标轴显示，如图 2-31 所示。选中 Cube，就选中了两个物体，坐标轴显示在两个物体中心，且只能两者一起移动。无法单独选中 Cube，但能单独选中 Sphere，且能单独移动 Sphere。



(a) 选中 Cube

(b) 选中 Sphere

图 2-31 Center：分别选中 Cube 和 Sphere 的坐标轴显示效果

若位置辅助工具是 Pivot 时，分别选中 Cube 和 Sphere 的坐标轴显示，如图 2-32 所示。选中 Cube，坐标轴显示在 Cube 中心，虽然如此，移动时还是只能两者一起移动。同时依然可以选中 Sphere，且能单独移动 Sphere。



(a) 选中 Cube

(b) 选中 Sphere

图 2-32 Pivot：分别选中 Cube 和 Sphere 的坐标轴显示效果



提示：

Center 和 Pivot 的不同主要表现在多个物体的选择上。

要注意的是，变换辅助工具不会改变游戏物体的参数，它仅仅是物体轴心点的参考，帮助我们去进行移动或旋转等变换操作。

(2) 旋转辅助工具，显示游戏物体的本地坐标系或世界坐标系，常用于辅助物体旋转。

- Global(世界坐标)：所选中的游戏物体使用世界坐标。
- Local(本地坐标)：所选中的游戏物体使用自身坐标。

我们继续使用上面的 Cube 和 Sphere 说明两者的区别。

选中 Sphere，设置参数 Rotation(0,0,0)，确保 Cube 各个方向的旋转角为 0。此时无论选择 Global 还是 Local，坐标系没有任何改变，且轴向控制器与 Sphere 的轴向保持一致。此时 Sphere 的世界坐标与本地坐标是一致的，如图 2-33 所示。

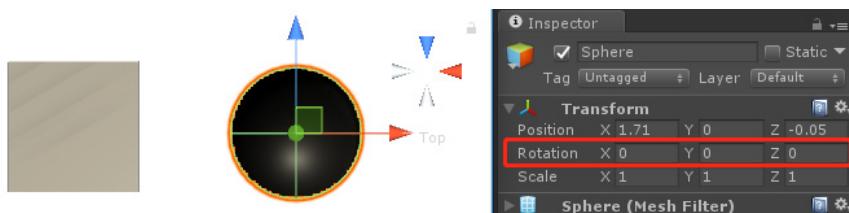
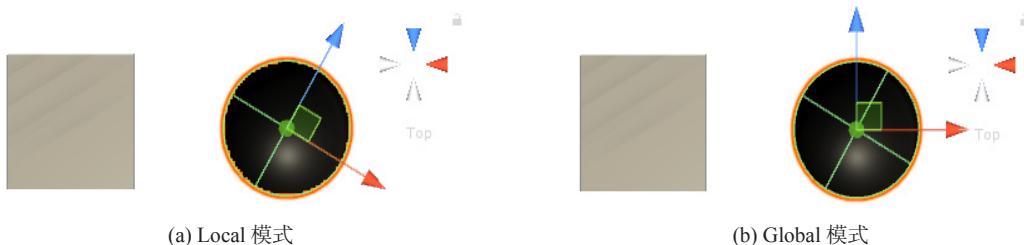


图 2-33 Rotation(0,0,0): Global 和 Local 模式下的轴向均与轴向控制器保持一致

当旋转角不为 0，如图 2-34 所示，将 Sphere 沿 Y 轴旋转了 30°，可以看到 Local 模式下，显示的是已经旋转了的本地坐标系，与表示世界坐标的轴向控制器的轴向不一样。而在 Global 模式下，显示的还是世界坐标系，如图 2-34 所示。



(a) Local 模式

(b) Global 模式

图 2-34 Rotation(0,30, 0): Local 和 Global 模式下坐标轴显示效果

3. Play

Play(播放控件) 用于控制游戏的播放，应用于 Game 窗口。其中，单击播放按钮，Game 视图被激活，可立即运行游戏，实时显示游戏运行的画面效果。暂停按钮用于分析复杂的行为，游戏过程中 (或暂停时) 可以修改参数、资源，甚至脚本。

要注意的是，播放或暂停中在 Inspector 窗口修改的数值在停止后会还原到播放前的状态 (脚本除外)。

4. Layers

Layers(分层下拉菜单)用于控制游戏物体在Scene窗口中的显示。显示状态为眼睛的物体会显示在Scene窗口中,如图2-35所示。

5. Layout

Layout(布局下拉菜单)用于切换编辑器的布局,也可以自定义并保存布局。

调整工作界面的布局为图2-36所示的样式。

方法:单击右上角的Layout下拉框,选择2 by 3选项,如图2-37所示。然后将Project窗口拖到Hierarchy窗口的下方,如图2-38所示。



图2-35 控制游戏物体在Scene窗口中的显示

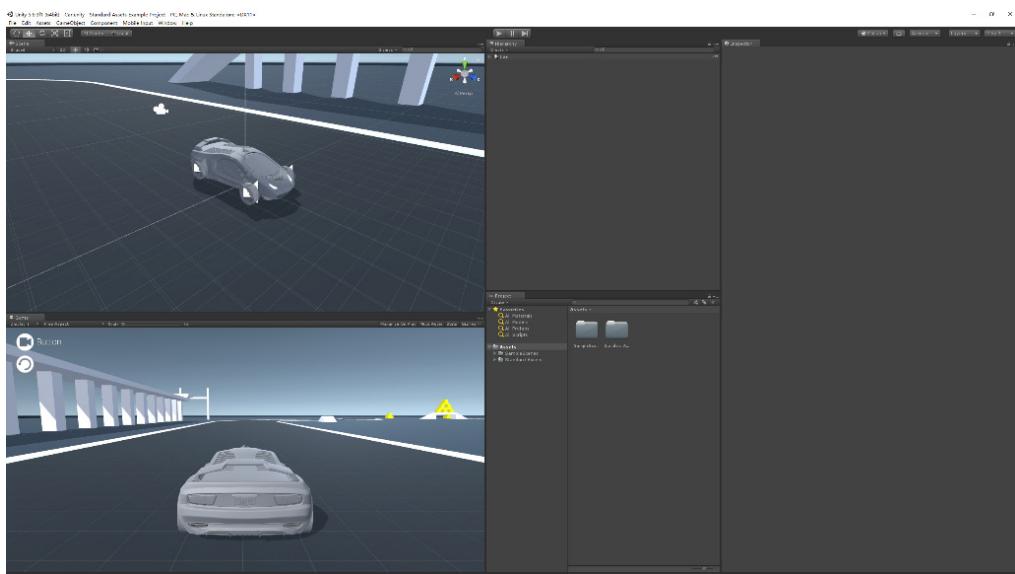


图2-36 Unity 工作界面调整后的布局

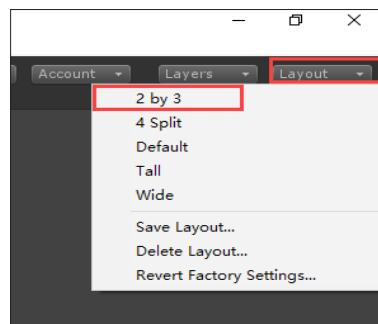


图2-37 更改布局方式

将整个窗口或大小调整至自己最舒服的状态,在右侧下拉菜单中可选择Save Layout保存布局,命名为MyLayout,此后打开布局下拉菜单可直接使用该布局,如图2-39所示。

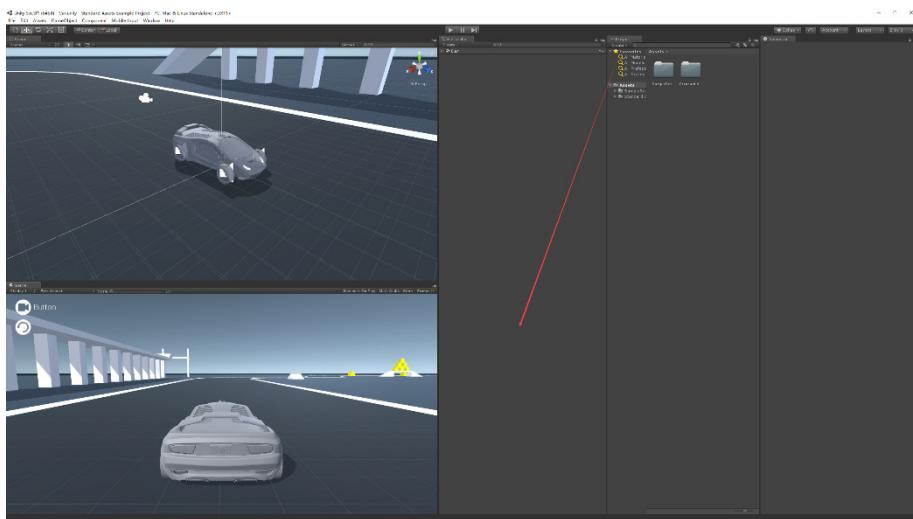


图 2-38 拖曳 Project 窗口到 Hierarchy 窗口下方

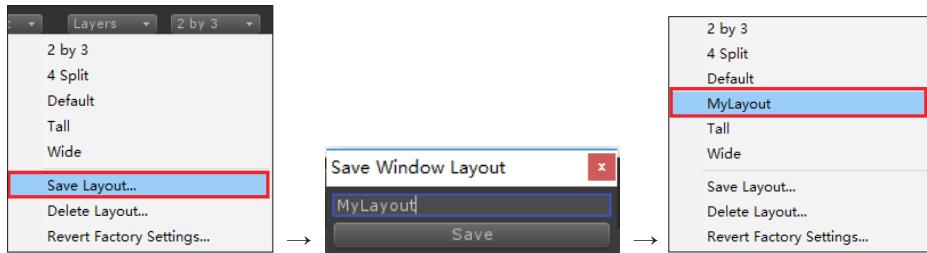


图 2-39 保存当前的界面布局并命名为 MyLayout

2.2.3 菜单栏

菜单栏集成了 Unity 的所有功能。Unity 默认情况下共有 7 个菜单项，分别是 File、Edit、Assets、GameObject、Component、Window 和 Help。

1. File(文件) 菜单

File 菜单主要包含工程与场景的创建、保存和输出等功能，如图 2-40 所示。

下面介绍如何发布 EXE 文件。

(1) 选择菜单栏中的 File→Build Settings 命令，弹出 Build Settings 窗口，如图 2-41 所示。



图 2-40 File 菜单

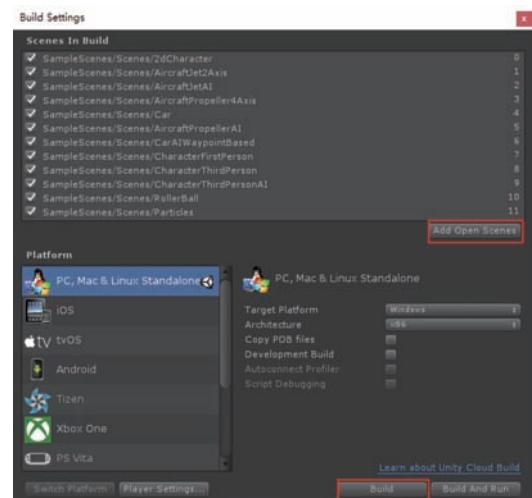


图 2-41 Build Settings 窗口

(2) 在左下角选择要发布的平台，默认选项是 PC, Mac&Linux Standalone，即输出到 PC、Mac 及 Linux 平台。右侧 Target Platform 选择 Windows，即输出到 Windows 平台。单击 Add Open Scene 按钮，将打开的场景加入发布列表中，默认已选了所有场景。

单击 Build 按钮，弹出保存文件的对话框，选择保存路径后输入文件名 Car，单击“保存”按钮即可输出 EXE 文件，如图 2-42 所示。

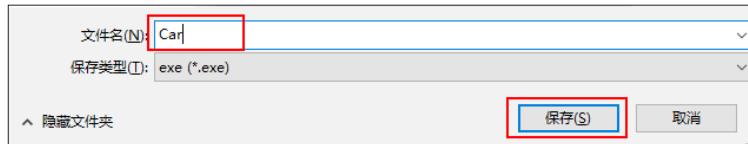


图 2-42 保存文件对话框

(3) 等待进度条结束，即可生成 Car.exe 文件，如图 2-43 所示。双击打开 Car.exe 文件，弹出 Example Project Configuration 对话框，如图 2-44 所示。将图像质量调为 Fantastic(最佳)，选中 Windowed 复选框 (窗口化运行，否则全屏)，然后单击 Play 按钮运行即可打开游戏。

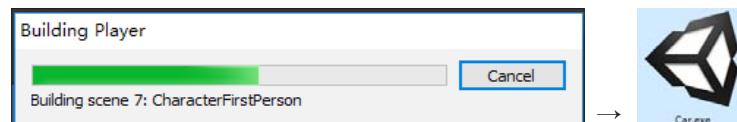


图 2-43 生成 EXE 文件

2. Edit(编辑) 菜单

Edit 菜单主要用来实现场景内部相应的编辑设置，如图 2-45 所示。

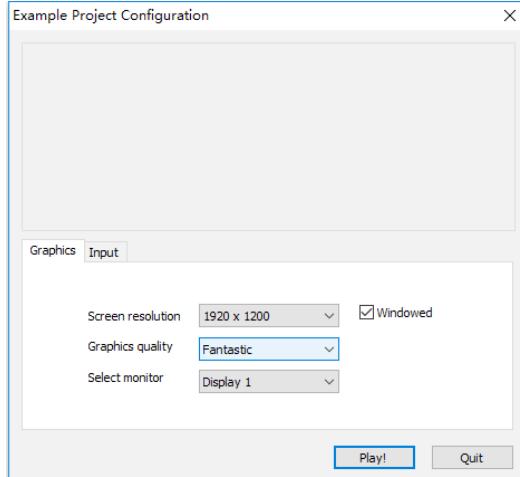


图 2-44 Example Project Configuration 对话框



图 2-45 Edit 菜单

3. Assets(资源) 菜单

Assets 菜单提供了针对游戏资源管理的相关工具，通过 Assets 菜单的相关命令，用户可以在场景内创建相应的游戏物体，导入导出所需资源包，如图 2-46 所示。

4. GameObject(游戏物体) 菜单

GameObject 菜单主要创建游戏物体，如灯光、粒子、模型和 UI 等，如图 2-47 所示。

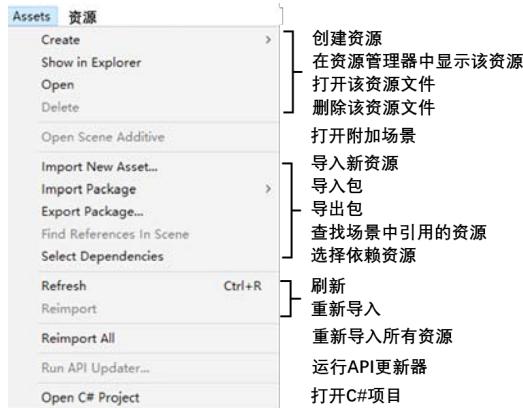


图 2-46 Assets 菜单



图 2-47 GameObject 菜单

5. Component(组件) 菜单

Component 菜单将组件添加到游戏物体上，实现游戏物体的特定属性。本质上，每个组件是一个类的实例。该菜单提供了多种常用的组件资源，如图 2-48 所示。

6. Window(窗口) 菜单

Window 菜单控制编辑器的界面布局，可用于打开各种窗口和访问 Unity 的 Assets Store 在线资源商店，如图 2-49 所示。



图 2-48 Component 菜单



图 2-49 Window 菜单

7. Help(帮助) 菜单

Help 菜单汇集了 Unity 相关资源链接，如 Unity 手册、脚本参考等，如图 2-50 所示。



图 2-50 Help 菜单

2.3 基本操作

按照“国际惯例”，学习一门语言的第一个程序就是输出一行字：“Hello World！”。在这里我们也沿袭这个惯例，用 Unity 完成一个“Hello World”程序，并将它编译成一个标准的 Windows 可执行的程序，步骤如下。

1. 新建项目工程

打开 Unity，单击 New 按钮新建 Unity 项目工程。输入 Project Name(项目名称) 和 Location(保存路径)，单击 Create Project 按钮即可创建新工程，如图 2-51 所示。

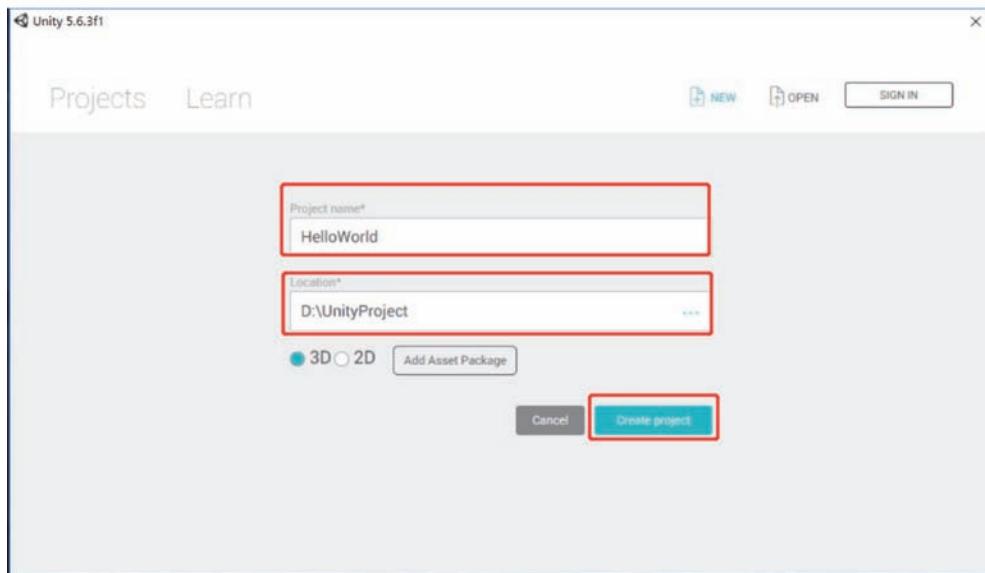


图 2-51 新建项目工程



场景中包含名为 Main Camera 的摄像机以及 Directional Light(方向光)。在 Hierarchy 窗口选择该相机，在 Scene 窗口右下角显示摄像机预览的缩略图 Camera Preview，如图 2-52 所示。

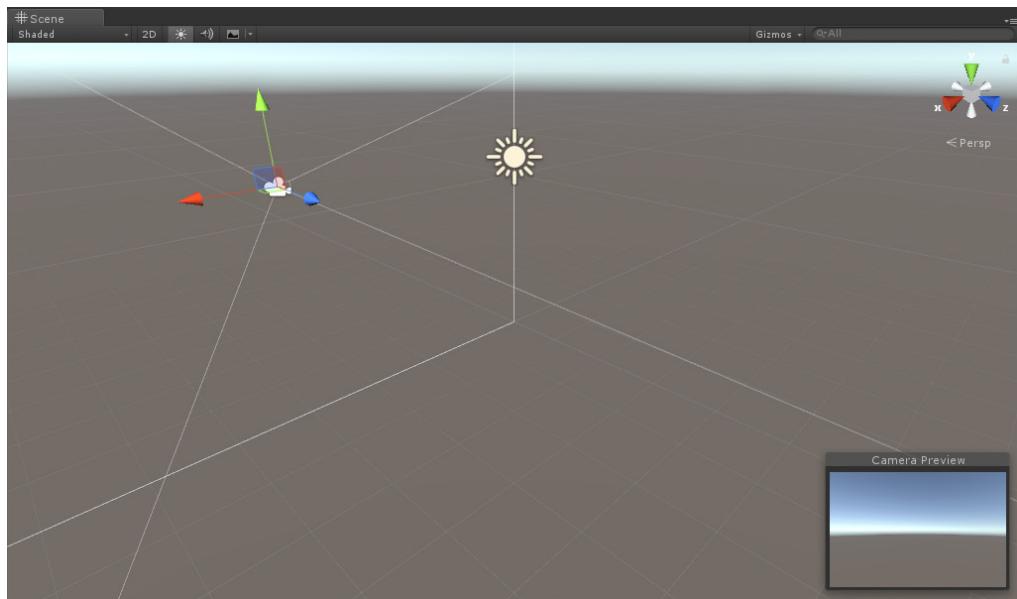


图 2-52 摄像机及其预览缩略图

2. 创建游戏物体

创建 Text 控件，用于显示 Hello World 文字，有以下三种创建方法：

(1) 在 Hierarchy 窗口单击 Create 下拉菜单，选择 UI → Text，如图 2-53 所示，即可在 Hierarchy 窗口出现 Canvas、Text 和 EventSystem 三个游戏物体，如图 2-54 所示。其中 Text 是 Canvas 的子物体。

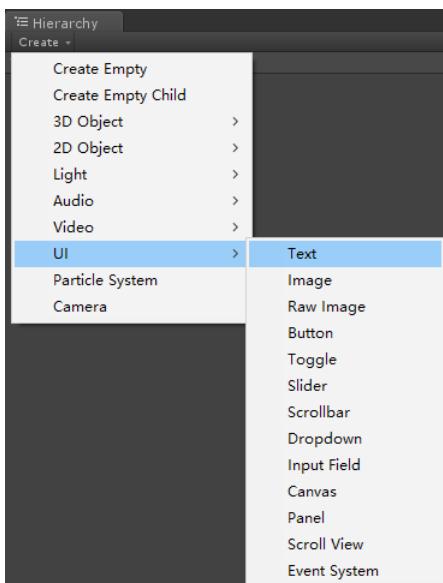


图 2-53 单击 Create 下拉菜单创建 Text

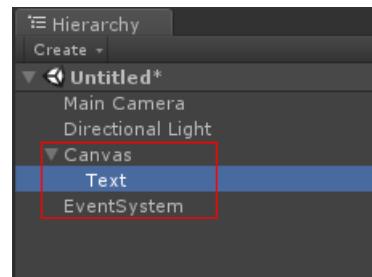


图 2-54 创建 Text 后同时创建了 Canvas 和 EventSystem