

新媒体与设计类系列教材
“互联网+”新形态一体化教材

Unity游戏开发

主审 陈彦许
主编 张红艳 郭海新 聂哲



上海交通大学出版社
SHANGHAI JIAO TONG UNIVERSITY PRESS

新媒体与设计类系列教材
“互联网+”新形态一体化教材

Unity游戏开发

主审 陈彦许
主编 张红艳 郭海新 聂哲



上海交通大学出版社
SHANGHAI JIAO TONG UNIVERSITY PRESS

内容提要

本书内容包括 Unity 引擎基础、Unity 资源、Unity 地形系统、Unity 脚本系统、Unity UI 系统、Unity 动画系统 6 个模块，通过结合相应案例对 Unity 软件进行了详细的讲解，旨在让读者通过学习和实践，对 Unity 软件的使用有一个全面的认识，并对通过 Unity 软件进行游戏开发有一个初步的了解，为后续的学习打下坚实的基础。

本书可作为高等院校相关专业的教材，也可作为游戏制作、虚拟现实等相关行业人员的参考书。

图书在版编目 (CIP) 数据

Unity 游戏开发 / 张红艳, 郭海新, 聂哲主编. —

上海: 上海交通大学出版社, 2023.10

ISBN 978-7-313-29581-1

I. ① U… II. ①张… ②郭… ③聂… III. ①游戏程序—程序设计 IV. ① TP311.5

中国国家版本馆 CIP 数据核字 (2023) 第 191965 号

Unity 游戏开发

Unity YOUXI KAIFA

主 编: 张红艳 郭海新 聂 哲

地 址: 上海市番禺路 951 号

出版发行: 上海交通大学出版社

电 话: 021-6407 1208

邮政编码: 200030

印 制: 北京荣玉印刷有限公司

经 销: 全国新华书店

开 本: 889mm × 1194mm 1/16

印 张: 15

字 数: 378 千字

版 次: 2023 年 10 月第 1 版

印 次: 2023 年 10 月第 1 次印刷

书 号: ISBN 978-7-313-29581-1

定 价: 79.00 元

版权所有 侵权必究

告读者: 如发现本书有印装质量问题请与印刷厂质量科联系

联系电话: 010-6020 6144

国家职业教育智慧教育平台学习指南

本书配套国家职业教育智慧教育平台在线精品课程“Unity 游戏开发”。读者可通过以下方式在线自主学习。

1. 登录国家职业教育智慧平台

输入网址 <https://vocational.smartedu.cn/>，进入智慧教育平台，在搜索栏输入“Unity 游戏开发”，单击搜索。



2. 加入课程

单击搜索后出现“Unity 游戏开发”课程界面，单击“现在去学习”，即可进入“加入课程”界面。单击“加入课程”即可开始学习。



前言

数字游戏作为人类社会文化的重要组成部分，它的蓬勃发展在很大程度上推动着社会发展以及人类文明的进步和发展。数字游戏的发展为我们提供了一个全新的文化、艺术、技术、经济、社交等领域的多元化发展空间。

在这场数字游戏的新革命中，Unity 以其强大的综合开发性能和易用性，成为游戏开发行业最受欢迎的游戏引擎之一。它可以用于制作 PC (personal computer) 游戏、主机游戏，制作的 PC 游戏和主机游戏也可以轻松移植到移动端等多个平台。

随着 Unity 游戏引擎的普及和应用，越来越多的学习者涌入这个领域，体验数字游戏的乐趣和创作的魅力。正是基于这个背景，我们编写了这本《Unity 游戏开发》。

本书共分为 6 个模块，模块 1 主要介绍游戏引擎的相关概念及 Unity 的基本操作方法；模块 2 介绍了 Unity 游戏开发过程中必备的资源文件的导入和导出方法以及资源商店的使用方法；模块 3 详细讲述了如何使用 Unity 制作游戏地形；模块 4 介绍了 Unity 脚本语言 C# 的基础知识；模块 5 基于 Unity 官方提供的 UI 系统详细说明了如何开发游戏登录界面、设置界面等，增加用户在游戏中的交互体验；模块 6 详细讲解了 Unity 动画的创建及使用方法，以及如何利用 Timeline 和 Cinemachine 插件制作 CG 动画等。

本书在编写过程中注重课程思政的融入，落实立德树人根本任务，深入贯彻党的二十大精神。在模块二中引入了中国凉亭资源，宣传中国传统文化，增强民族文化自信；我们牢记“绿水青山就是金山银山”，注重人与自然的和谐共生，在模块 2 中引入了自然风光场景的制作。

本书还注重理论与实践的结合，每个模块都设置了一个综合实践项目，方便读者掌握相关的知识。本书高度融合 1+X 3D 引擎技术应用职业技能等级证书（中级）考试的内容，并在模块 6 的综合实践环节引入了 1+X 证书的实训项目。

此外，本书还有随书素材和教学课件等配套资源，有需要者可致电 13810412048 或发邮件至 2393867076@qq.com 领取。

本书由张红艳、郭海新、聂哲担任主编，孙小华、尹腾腾、梁悦担任副主编。参与编写的还有温爱华、陈宇、刘立圆、王东方和王峥。特别感谢陈彦许教授在本书撰写过程中给予的大力支持和殷切指导。感谢 1+X 3D 引擎技术应用职业技能等级证书评价组织的无私帮助。但是由于 Unity 技术发展迅速，同时受编者自身水平所限，本书存在疏漏和不足之处，敬请广大读者提出宝贵的意见和建议。

目 录

模块 1 Unity 引擎基础 001

任务 1.1 认识游戏引擎	001
1.1.1 什么是游戏引擎	001
1.1.2 游戏引擎在游戏开发中的地位	002
1.1.3 游戏引擎的组成部分	002
任务 1.2 认识 Unity 引擎	003
1.2.1 Unity 引擎的特性	003
1.2.2 Unity 引擎的优势	003
1.2.3 Unity 引擎的应用领域	004
任务 1.3 搭建 Unity 集成开发环境	005
1.3.1 安装 Unity Hub	005
1.3.2 安装 Unity Editor	007
1.3.3 添加管理许可证	009
1.3.4 创建第一个 Unity 项目	011
任务 1.4 熟悉 Unity 界面布局及基础操作	016
1.4.1 Project 视图	016
1.4.2 Hierarchy 视图	017
1.4.3 Scene 视图	018
1.4.4 Game 视图	023
1.4.5 Inspector 视图	025
1.4.6 菜单栏	025
1.4.7 工具栏	027
任务 1.5 综合实践——创建时钟	029
1.5.1 创建表盘	029
1.5.2 创建时间刻度	031
1.5.3 创建表针	032
任务 1.6 Unity 项目发布	033
1.6.1 设置文件图标和鼠标	033
1.6.2 设置分辨率	035

1.6.3	设置闪屏	036
1.6.4	打包文件	038
任务 1.7	拓展训练	040

模块 2 Unity 资源 041

任务 2.1	了解 Unity 资源	041
任务 2.2	导入资源	042
任务 2.3	导入 / 导出资源包	043
2.3.1	导入资源包	043
2.3.2	导出资源包	043
任务 2.4	使用资源商店	044
任务 2.5	综合实践——制作 Nature 场景	047
2.5.1	下载 Nature 场景	047
2.5.2	添加人物模型	048
2.5.3	添加长凳	049
任务 2.6	拓展训练	053

模块 3 Unity 地形系统 054

任务 3.1	创建地形	054
3.1.1	创建相邻地形	055
3.1.2	了解地形参数	056
3.1.3	绘制地形	057
任务 3.2	创建树木	061
3.2.1	添加树木模型	061
3.2.2	绘制树木	062
任务 3.3	添加花草	063
3.3.1	添加细节	063
3.3.2	绘制花草	064
任务 3.4	添加环境效果	065
3.4.1	添加水和雾	065

3.4.2	添加天空盒	066
任务 3.5	添加角色控制器	067
3.5.1	添加第一人称角色控制器	068
3.5.2	添加第三人称角色控制器	068
任务 3.6	综合实践——漫游毛泽东故居	069
3.6.1	导入资源	069
3.6.2	创建和绘制地形	070
3.6.3	绘制山丘	070
3.6.4	导入房子	070
3.6.5	绘制细节	071
3.6.6	创建树木	072
3.6.7	绘制绿草	072
3.6.8	绘制荷花池	073
3.6.9	设置天空盒	074
3.6.10	设置风力	074
3.6.11	导入第一人称角色控制器	075
任务 3.7	拓展训练	075
模块 4 Unity 脚本系统		
任务 4.1 初识脚本		077
4.1.1	创建脚本	077
4.1.2	选择脚本后端	077
4.1.3	编辑脚本	077
4.1.4	运行脚本	078
任务 4.2 定义与使用变量		079
4.2.1	初识变量	079
4.2.2	熟悉变量的命名规则	079
4.2.3	认识数据类型	079
4.2.4	定义变量	081
4.2.5	使用变量	081
任务 4.3 使用条件语句		083
4.3.1	条件判断语句	083
4.3.2	条件分支语句	084
4.3.3	使用条件语句	085

任务 4.4	使用循环语句	090
4.4.1	循环语句	090
4.4.2	使用循环语句	092
任务 4.5	定义与调用函数	093
4.5.1	熟悉常用的函数	093
4.5.2	定义函数	094
4.5.3	调用函数	094
4.5.4	使用函数	094
任务 4.6	定义与使用结构体	096
4.6.1	声明结构体	096
4.6.2	声明结构体变量	096
4.6.3	使用结构体变量	096
4.6.4	为结构体变量赋值	097
4.6.5	序列化结构体	097
4.6.6	使用结构体	097
任务 4.7	定义、使用类与对象	099
4.7.1	认识类和对象	099
4.7.2	定义类	100
4.7.3	定义及使用对象	102
4.7.4	使用类与对象	102
任务 4.8	综合实践——简易第一人称射击游戏	103
4.8.1	制作子弹	104
4.8.2	自动销毁子弹	104
4.8.3	发射子弹	105
任务 4.9	拓展训练	105

模块 5 Unity UI 系统 106

任务 5.1	熟悉 UGUI 的基本控件	106
5.1.1	Canvas	106
5.1.2	画布中绘制元素的顺序	107
5.1.3	创建画布	107
5.1.4	画布的渲染模式	108
5.1.5	Canvas Scaler 组件	110

任务 5.2 认识 UGUI 的可视控件	111
5.2.1 UI 控件的布局	111
5.2.2 Text 控件	113
5.2.3 Image 控件	116
5.2.4 RawImage 控件	118
5.2.5 Mask 控件	119
任务 5.3 认识 UGUI 的交互控件	119
5.3.1 Button 控件	120
5.3.2 Toggle 控件	124
5.3.3 Toggle Group 控件	127
5.3.4 Slider 控件	128
5.3.5 Scrollbar 控件	131
5.3.6 Dropdown 控件	134
5.3.7 Input Field 控件	138
任务 5.4 综合实践——游戏界面制作	142
5.4.1 设计登录界面	142
5.4.2 设计设置界面	150
5.4.3 实现 UI 交互设计	157
任务 5.5 拓展训练	164
模块 6 Unity 动画系统	165
任务 6.1 认识 Mecanim 动画系统	165
6.1.1 准备动画模型	166
6.1.2 设置动画	167
6.1.3 Animator 组件	170
6.1.4 反向动力学动画	178
6.1.5 动画层	184
6.1.6 混合树	188
6.1.7 使用动画重定向	192
任务 6.2 认识时间轴	193
6.2.1 Timeline 的组成部分	194
6.2.2 创建 Timeline 资源和 Timeline 实例	194
6.2.3 创建 Timeline 轨道	197

任务 6.3 认识 Cinemachine 虚拟相机	204
6.3.1 安装与删除 Cinemachine.....	205
6.3.2 创建 Virtual Camera.....	207
6.3.3 创建 FreeLook Camera	213
6.3.4 创建 Dolly Camera With Track	216
任务 6.4 综合实践——制作 CG 动画	220
6.4.1 创建远景相机	220
6.4.2 创建近景相机并实现镜头切换	221
6.4.3 创建轨道相机	222
6.4.4 创建超远景相机	224
6.4.5 添加淡入淡出特效和音乐	225
任务 6.5 拓展训练	226

参考文献	227
-------------------	------------

模块 1

Unity 引擎基础

◉【学习目标】

知识目标

- (1) 了解游戏引擎的概念及组成部分。
- (2) 了解 Unity 引擎的特性及应用领域。
- (3) 掌握 Unity 引擎开发环境的配置方式。
- (4) 掌握 Unity 引擎的基本操作。

能力目标

- (1) 能够配置 Unity 开发环境。
- (2) 能够使用 Unity 引擎进行简单场景设计。

素养目标

- (1) 提高分析问题、解决问题的能力。
- (2) 培养严谨的工作态度。

◉【模块导读】

随着个人计算机的不断普及和技术的不断革新，计算机游戏逐渐成为人类重要的娱乐方式，计算机游戏行业得到迅猛的发展。(以下所称“游戏”均指计算机游戏。)而游戏引擎作为游戏开发的重要工具，也得到了社会的广泛关注。目前市面上的游戏引擎有很多，除了常见的 Unreal 引擎和 Unity 引擎，还有 Frostbite、Source、Cocos、Godot、IW 等引擎。本模块将以 Unity 引擎为例介绍游戏引擎的基本概念以及 3D 引擎的使用方法。

任务 1.1 认识游戏引擎

1.1.1 什么是游戏引擎

二十世纪九十年代以前，游戏引擎的概念还没有出现，大部分电子游戏都是根据不同厂商生产的特定机型进行定制开发的，计算机也大多用于商业用途而非娱乐。当时每开发一款游戏都需要开发者从头编写代码，开发期间存在大量重复性的工作，耗时耗力。开发者们逐渐意识到这一问题，并在开发过程中发现有一些代码可以在不同游戏中重复使用。随着对这些通用代码的积累，游戏引擎的雏形逐渐形成。

究竟什么才是游戏引擎呢？游戏引擎是一些已编写好的可编辑电脑游戏系统或者一些交互式实时图像应用程序的核心组件。这些系统或核心组件为游戏设计者提供编写游戏所需的各种工具，其目的在于让游戏设计者能更加容易和快速地做出游戏程序而不用由零开始。简单来说，游戏引擎就是一个框架，为用户提供了各种各样的功能。开发者可以通过使用游戏引擎提供的功能模块提升开发效率，从而将更多的精力放在游戏的创意和内容的设计上。

1.1.2 游戏引擎在游戏开发中的地位

有人将游戏引擎比作汽车的“发动机”，也有人将其比作人的“大脑”，游戏引擎究竟在游戏开发的过程中扮演怎样的角色，处于什么样的地位呢？我们可以将游戏的开发过程抽象成一幅金字塔图，如图 1-1-1 所示，位于最底层的是图形处理器（Graphics Processing Unit, GPU），它是游戏开发的物理基础，所有的游戏效果都需要使用 GPU 进行渲染。第二层是图形应用程序接口（Application Program Interface, API），目前主流的图形 API 有 OpenGL 和 DirectX。第三层是游戏引擎（Game Engine），它建立在图形 API 基础之上，并通过图形 API 指挥 GPU 完成指定操作。第四层是软件开发工具包（Software Development Kit, SDK），游戏程序员和美工可以利用 SDK 提供的功能为自己的游戏建立模型、动画以及画面效果，最终形成第五层的游戏产品（Game）。

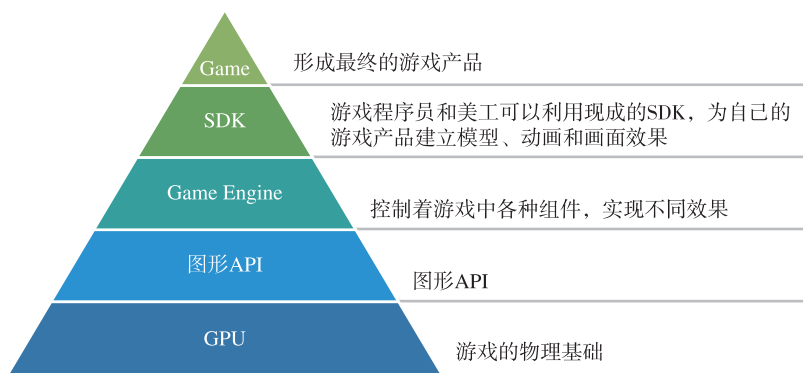


图 1-1-1 游戏引擎在游戏开发中的地位

1.1.3 游戏引擎的组成部分

游戏引擎主要包括渲染系统、物理引擎、音效系统、脚本引擎、动画系统、人工智能、网络引擎和场景管理系统等。

1. 渲染系统

渲染系统是游戏引擎的重要组成部分，它将模型、动画、光影、特效等所有效果实时计算出来并展示在屏幕上，它的强大与否直接决定着最终的输出质量高低。

2. 物理引擎

物理引擎是游戏引擎的另一重要功能，它可以帮助游戏开发者在开发项目中模拟物理系统，以确保游戏内的物体能与环境、角色和其他物体进行正确的运动交互。

3. 音效系统

音效系统主要用来完成对声音的加载及控制等。

4. 脚本引擎

脚本引擎是为游戏开发者提供的，游戏开发者可以通过脚本引擎编写代码控制整个游戏的逻辑及运行。

5. 动画系统

游戏所采用的动画系统可以分为两种：一种是骨骼动画系统，另一种是模型动画系统。前者用内置的骨骼带动物体产生运动，比较常见；后者则是在模型的基础上直接进行变形。游戏引擎把这两种动画系统预先植入游戏，方便动画师为角色设计丰富的动作造型。

6. 人工智能

人工智能即 AI 技术，主要用于游戏内容生成、剧情生成、人物生成、场景生成、智能 NPC（非玩家角色）和世界导航等。

7. 网络引擎

网络引擎主要用来完成服务器和客户端之间的通信。

8. 场景管理系统

场景管理系统主要用来完成 3D 场景的组织 and 绘制。

任务 1.2 认识 Unity 引擎

目前市面上流行的游戏引擎有很多，其中 Unreal 和 Unity 两款引擎表现尤为出色，在国内的一些中小型游戏公司中，又以 Unity 的使用更为广泛。

Unity 是由 Unity Technologies 公司研发的一款实时 3D 开发平台，它提供了一整套完善的软件解决方案，用户可以通过它轻松地实现各种游戏创意和三维互动开发，创作出精彩的 2D 和 3D 游戏内容。它支持手机、电脑、游戏主机、增强现实和虚拟现实设备等平台。

1.2.1 Unity 引擎的特性

Unity 作为一款成熟的游戏引擎，主要具有以下特性：

- (1) 完整的游戏功能，如图形、声音、物理等功能。
- (2) 完善的编辑器，如场景编辑器、模型编辑器、动画编辑器等。
- (3) 完善的第三方插件，如 NGUI、EasyTouch、Final IK 等。
- (4) 易用的脚本语言，Unity 采用主流的开发语言 C#，开发者不需要了解 C# 底层就可以开发应用程序。
- (5) 良好的生态系统，Unity 提供了一个活跃的资源商店，里面有丰富的资源可供用户下载使用。

1.2.2 Unity 引擎的优势

对于开发者来说，选择 Unity 具有以下优势：

1. 开发效率高，学习成本低

Unity 作为一款成熟的游戏引擎，它的功能比较完备，开发者学习起来容易上手。因为关于 Unity 的学习资料比较完善，所以开发者的学习成本也比较低。

2. 非常强大且便捷的跨平台性和适配性

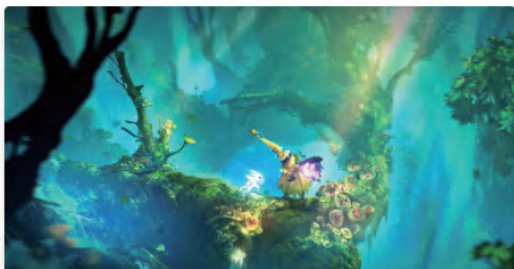
当前 Unity 支持 20 多个主流平台的游戏开发。开发者只需要开发一次，就可以发布到不同的平台，节省了游戏移植的成本。

想一想

2022 年 12 月，腾讯公司出品的《王者荣耀》在全球 App 商店和 Google Play 中吸金近 2 亿美元，蝉联全球手游畅销榜冠军。假如时光穿梭回《王者荣耀》的开发阶段，而你成了这款游戏开发的决策者，那么你会选用 Unity 和 Unreal 哪款游戏引擎呢？

1.2.3 Unity 引擎的应用领域

目前，Unity 的应用遍布游戏、汽车、运输与制造、工程建设与数字城市等多个领域，如图 1-2-1 所示。在游戏领域，腾讯公司出品的《王者荣耀》、暴雪公司出品的《炉石传说》、盛趣游戏出品的《龙之谷 2》等众多风靡全球的游戏都是使用 Unity 开发的；在汽车、运输与制造领域，Unity 与多家汽车公司在自动驾驶训练、设计体验、培训指导人机交互（Human Machine Interface, HMI）、市场销售等方面展开了合作；在工程建设与数字城市领域，瑞典建筑集团斯堪雅公司使用基于 Unity 的 VR 进行员工安全培训，Mortenson 建筑公司利用 Unity 的交互式 VR 和 360° 全景视频功能改善了医院设计。Unity 也与温哥华机场、海尔集团等公司建立前期合作，构建数字孪生平台用于调度、管理。



游戏

您制作、运营移动端游戏并获利、PC 和游戏主机游戏、AR 和 VR 游戏以及即时游戏所需的一切。

[了解详情](#)



汽车、运输与制造

运用实时 3D 技术在快速变化的行业局势中夺得竞争优势。

[了解详情](#)



电影与动画

通过实时制作加快动画制作流程并提高创作自由度

[了解详情](#)



建筑、工程与施工

为大规模的现实生活应用情形创建沉浸式 3D 模型体验。

[了解详情](#)

图 1-2-1 Unity 应用领域

任务 1.3 搭建 Unity 集成开发环境

1.3.1 安装 Unity Hub

首先登录 Unity 的中国官网 <https://unity.cn/>，单击网页右上角的“下载 Unity”，打开下载界面，如图 1-3-1 所示。

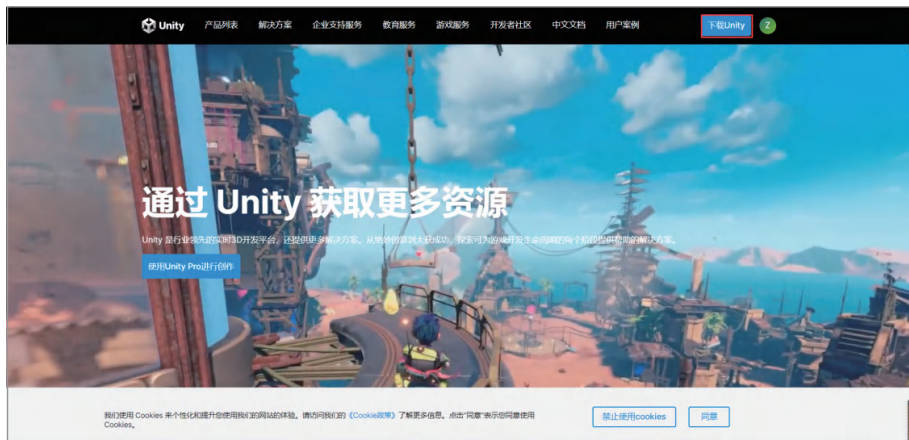


图 1-3-1 Unity 官网

滑动鼠标滚轮，将页面定位到下载 Unity 部分，单击“下载 Unity Hub”，如图 1-3-2 所示，在弹出的提示对话框中单击“Windows 下载”，下载 Unity Hub 的 Windows 版，如图 1-3-3 所示。



图 1-3-2 Unity Hub 下载



图 1-3-3 Unity Hub 版本选择

单击“Windows 下载”后，网页将会跳转到 Unity 登录界面，如果是第一次使用 Unity，并且之前未注册过 Unity ID，则需要先单击“创建 Unity ID”，完成 Unity 账号的注册，如图 1-3-4 所示。

Unity 账号注册完成后单击“帐户登录”，通过手机号或邮箱完成登录后，重新单击“下载 Unity Hub”，即可开始下载 Unity Hub。

下载完成后，双击运行 Unity Hub 安装程序，在安装界面上单击“我同意 (I)”，切换到安装路径选择界面，如图 1-3-5 所示。



图 1-3-4 创建 Unity ID



图 1-3-5 同意许可证协议

先单击“浏览 (B) ...”，将安装路径切换到想要放置的位置，再单击“确定”即可，如图 1-3-6 所示。路径切换完毕后，单击“安装 (I)”即可开始安装，如图 1-3-7 所示。



图 1-3-6 切换安装路径



图 1-3-7 安装 Unity Hub

安装完成后，单击“完成 (F)”即可运行 Unity Hub，如图 1-3-8 所示。



图 1-3-8 Unity Hub 安装完成

! 小提示

含有中文的 Unity 安装路径可能会导致 Unity 安装失败或出现乱码等情况，所以建议安装路径选择全英文。

1.3.2 安装 Unity Editor

目前 Unity 官方推出的编辑器版本众多，本书将以 Unity 2021.3.6 版本为例介绍安装步骤。由于 Unity 的编辑器具有向下兼容性，读者也可以在 2021.3.6 以后推出的编辑器中打开本书的项目。

启动 Unity Hub，单击“跳过安装”按钮，进入 Unity Hub 工作界面，如图 1-3-9 所示。

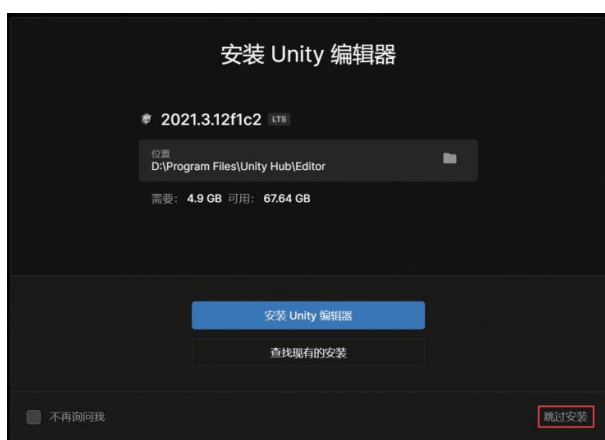


图 1-3-9 Unity Hub 启动界面

进入 Unity Hub 工作界面后，单击界面左侧的“安装”按钮，切换到安装界面，然后单击右上角的“安装编辑器”按钮，打开安装列表，如图 1-3-10 所示。

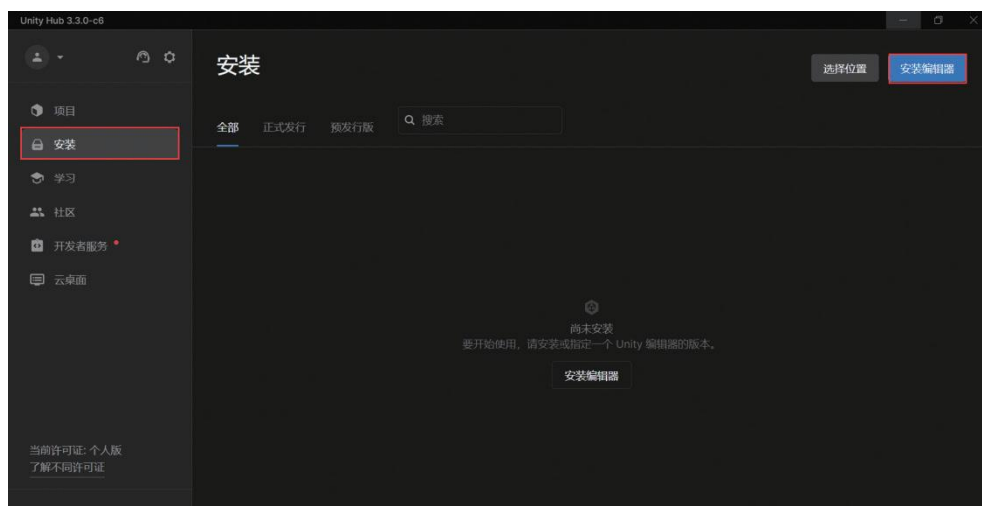


图 1-3-10 Unity Hub 安装界面

安装列表打开后，单击“存档”选项卡下的“下载存档”按钮，即可打开 Unity 下载界面，如图 1-3-11 所示。



图 1-3-11 下载存档

Unity 下载界面打开后，单击所有版本中的“Unity 2021.x”选项卡，选择 2021.3.6 版本，单击“从 Hub 下载”按钮，跳转到 Unity Hub 的安装界面，如图 1-3-12 所示。



图 1-3-12 下载 Unity 2021.3.6 版本

Unity Hub 安装界面打开后，单击“安装”按钮开始下载并安装 Unity 编辑器。如果需要使用中文版，则需要勾选“语言包 (预览)”下的“简体中文”语言包，如图 1-3-13 所示。

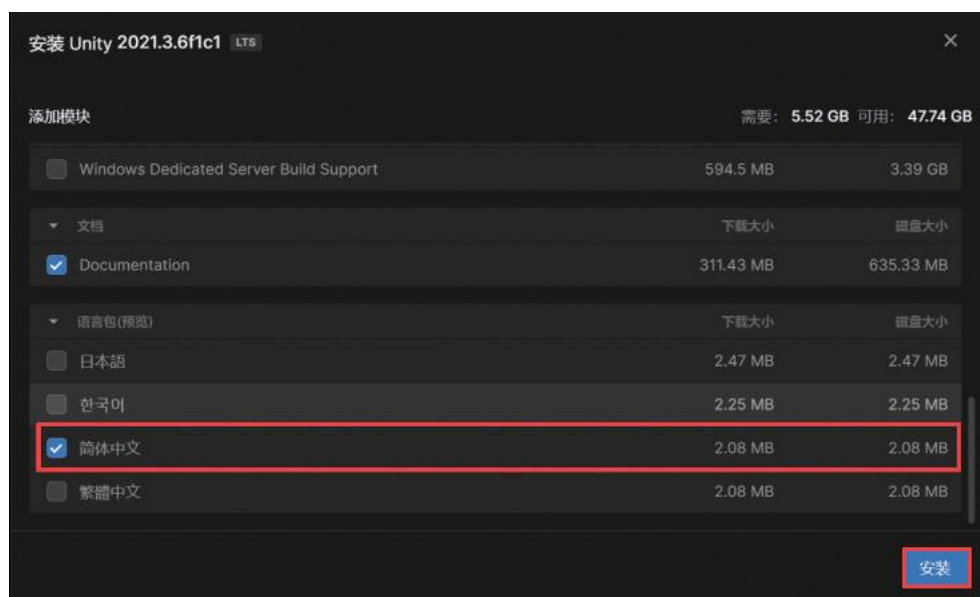


图 1-3-13 选择“简体中文”语言包

关闭安装 Unity 编辑器窗口，Unity Hub 窗口的左下方会出现下载进度，单击“下载”按钮或下载进度的任意位置即可打开下载界面，如图 1-3-14 所示。

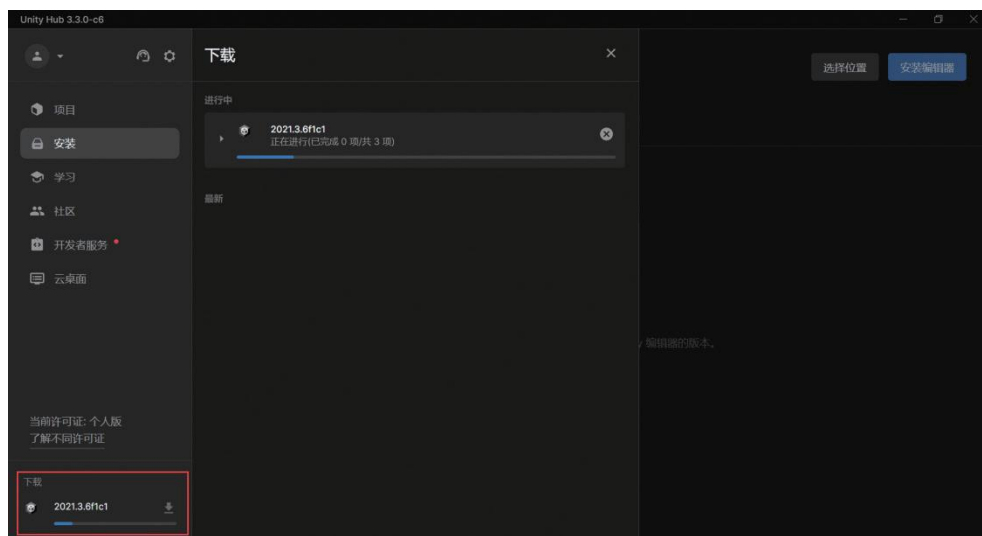


图 1-3-14 Unity 下载进度

Unity 2021.3.6 下载完成后会自动安装。安装完成后，Unity Hub 的安装完成界面中将会显示刚刚安装好的 Unity 编辑器，如图 1-3-15 所示。

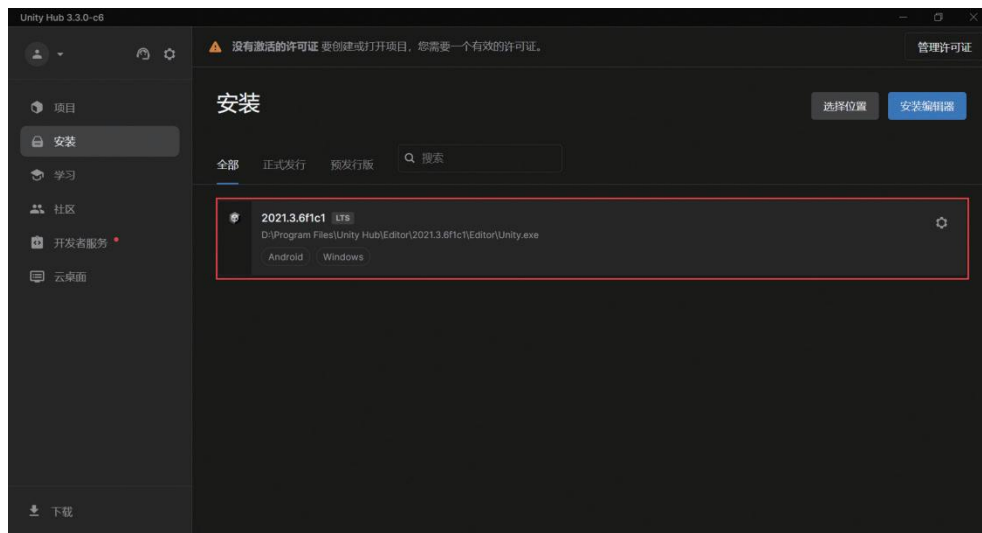


图 1-3-15 安装完成界面

1.3.3 添加管理许可证

Unity 编辑器在安装完成后并不能直接使用，在 Unity Hub 的上方会提示用户没有激活的许可证，需要先添加管理许可证再进行使用。单击 Unity Hub 右上角的“管理许可证”，打开“偏好设置”界面，单击“许可证”选项卡下的“登录”按钮，完成用户登录，如图 1-3-16 所示。

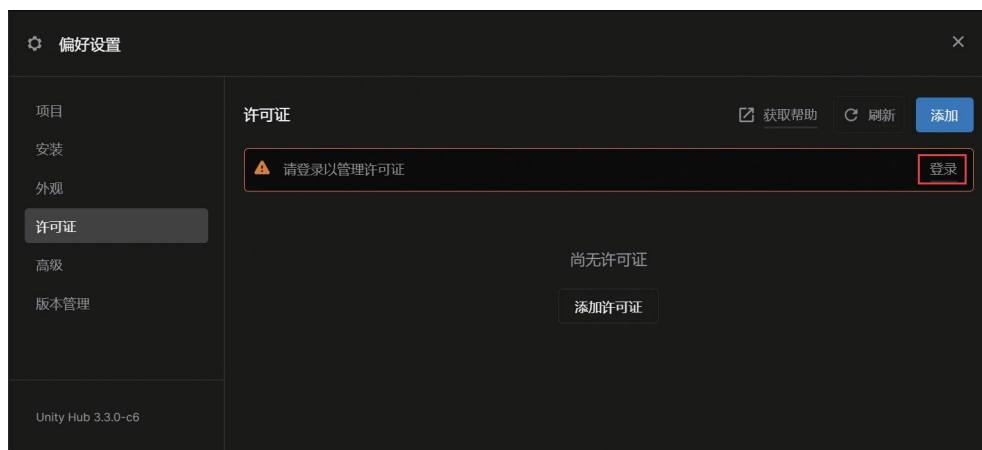


图 1-3-16 登录账号

用户登录完成后，单击“添加许可证”按钮，打开“添加新许可证”界面，如图 1-3-17 所示。

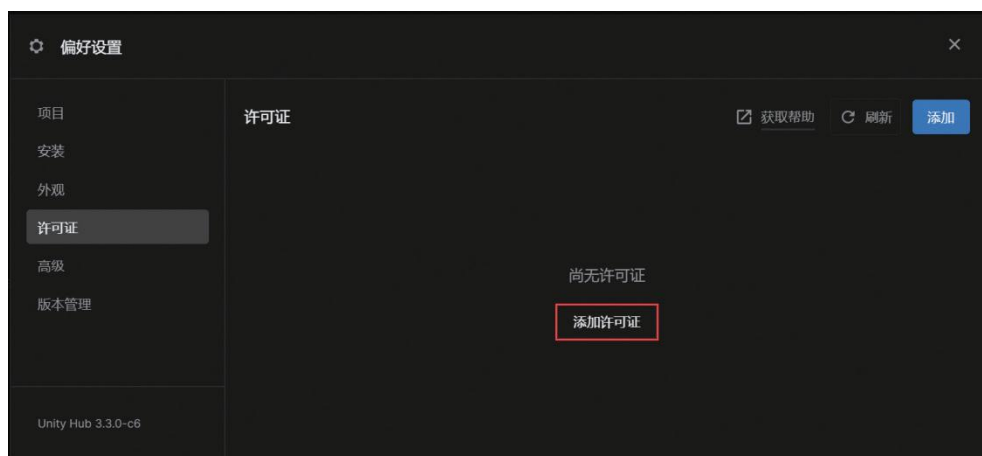


图 1-3-17 添加许可证

“添加新许可证”界面打开后，单击“获取免费的个人版许可证”按钮，如图 1-3-18 所示，在弹出的界面中单击“同意并获得个人版授权”按钮，即可完成许可证的添加。

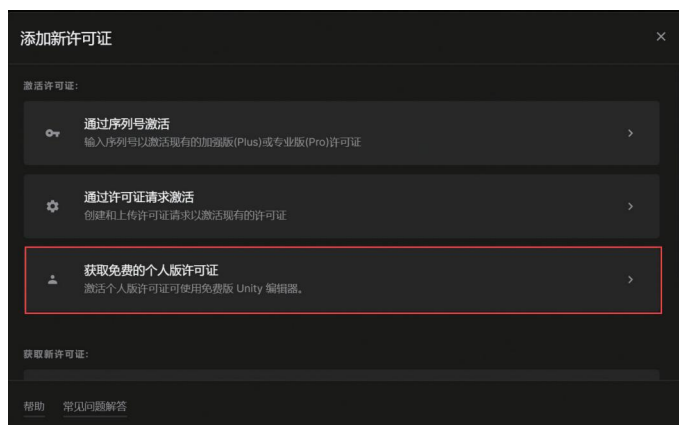


图 1-3-18 获取免费的个人版许可证

添加完成后，Unity 的许可证将会显示在许可证界面的列表中。此时单击“退还许可证”，将会删除之前添加的许可证，如图 1-3-19 所示。单击“添加”，即可继续添加其他许可证。

关闭“偏好设置”界面后，还可以通过单击 Unity Hub 左上角的图标再次打开，继续进行许可证的管理，如图 1-3-20 所示。

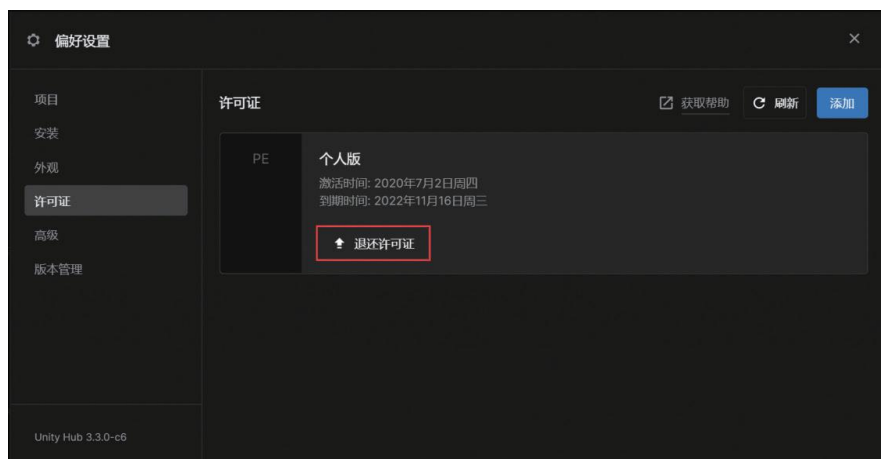


图 1-3-19 退还许可证



图 1-3-20 打开“偏好设置”界面的按钮

1.3.4 创建第一个 Unity 项目

1. 创建项目

打开 Unity Hub，单击左侧的“项目”选项卡，切换到项目界面，单击右上角的“新项目”按钮，如图 1-3-21 所示，在弹出的“新项目”窗口中创建一个“3D”新项目，将其重命名为“MyFirstProject”，并将“位置”切换到想要存放的路径，取消“启用版本管理并同意政策条款”复选框的勾选，单击“创建项目”按钮完成项目创建，并在 Unity 编辑器中打开创建的项目，如图 1-3-22 所示。

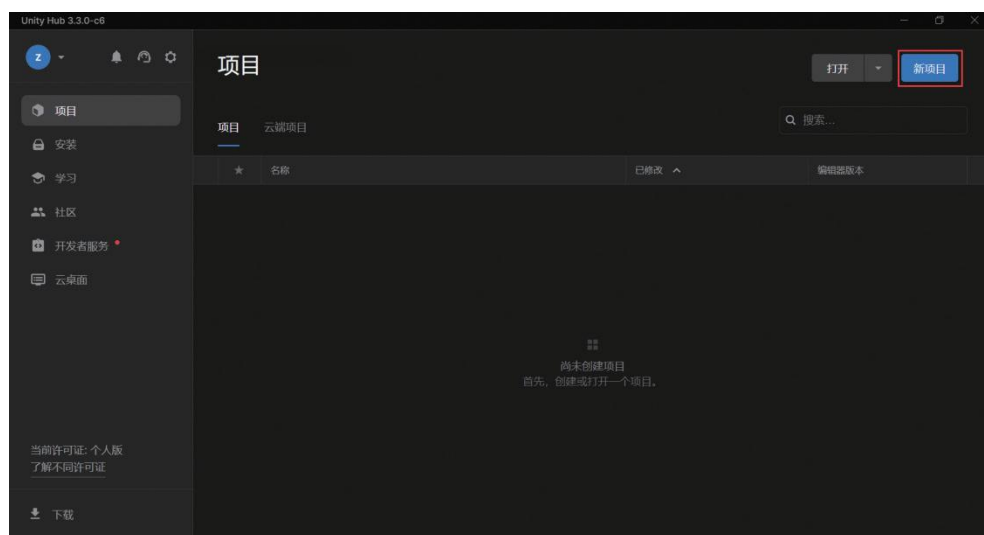


图 1-3-21 创建项目

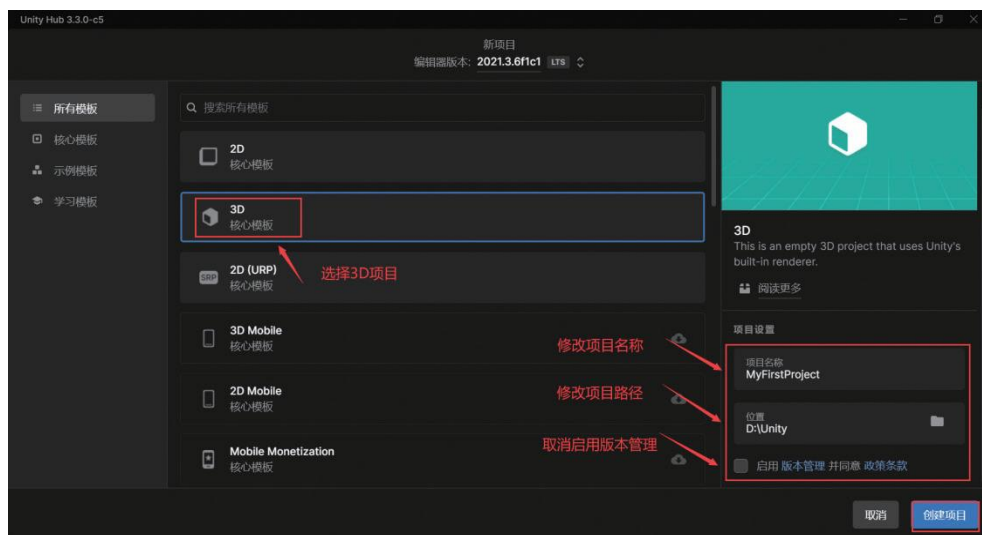



图 1-3-22 项目设置

小提示

与 Unity 的安装路径一样，为了保证项目正常运行，项目路径中尽量不要有中文。

2. 创建 3D 物体并运行项目

接下来为“MyFirstProject”项目的“SampleScene”添加一个 3D 物体，首先在“Hierarchy”视图中单击鼠标右键，在弹出的菜单中执行“3D Object”→“Cube”命令，创建一个立方体，如图 1-3-23 所示。然后单击窗口上方的图标，即可运行项目并查看项目的运行结果，如图 1-3-24 所示。

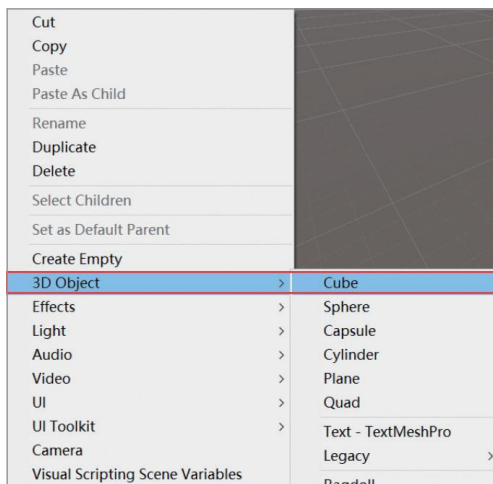


图 1-3-23 创建立方体



图 1-3-24 运行项目

3. 保存场景和项目

当 Unity 窗口的标题栏上显示 * 号时，如图 1-3-25 所示，这代表场景中有修改的内容尚未保存，此时执行菜单栏“File”→“Save”命令（或使用“Ctrl+S”组合键）即可完成保存，如图 1-3-26 所示。

如果需要将场景另存为其他名称或者另存到其他位置，则可以通过执行菜单栏“File”→“Save As”

命令（或使用“Ctrl+Shift+S”组合键），此时 Unity 会将当前场景另存为一个“.unity”文件，如图 1-3-27 所示。

如果需要保存整个项目，则可以通过执行菜单栏“File”→“Save Project”命令完成操作，如图 1-3-28 所示。

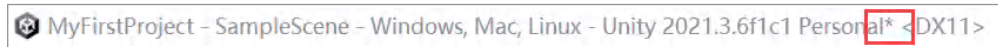


图 1-3-25 场景中有待保存内容

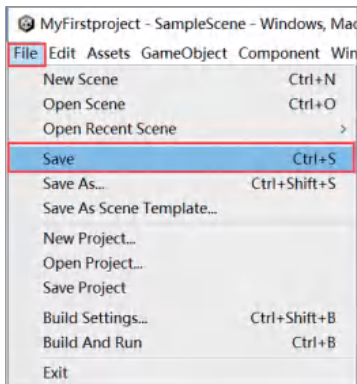


图 1-3-26 保存场景

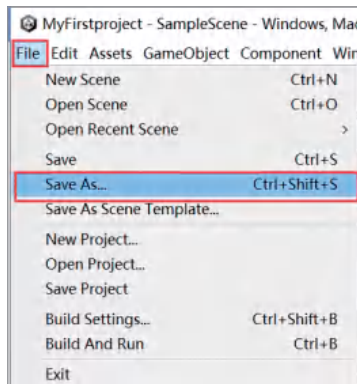


图 1-3-27 另存场景

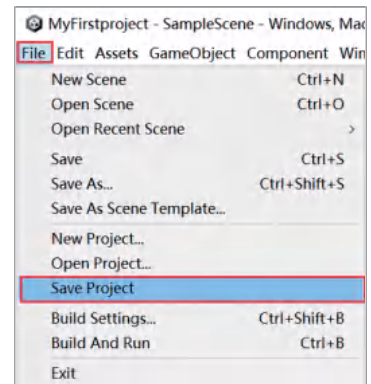


图 1-3-28 另存项目

4. 重新打开项目

项目关闭后，如果需要重新打开已经创建好的项目，则需要再次打开 Unity Hub，此时最近创建的项目会出现在项目列表中，如图 1-3-29 所示，单击即可打开。如果需要打开项目的编辑器版本跟安装版本不一致，则可以通过单击列表中的编辑器版本进行更换。

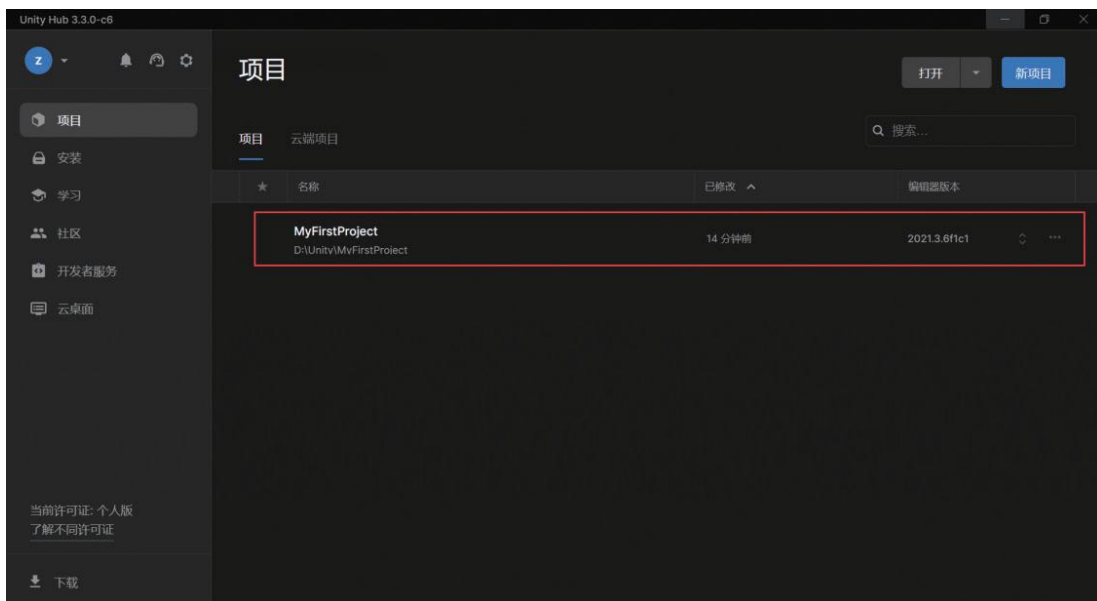


图 1-3-29 项目列表

小提示

除了使用 Unity Hub 打开项目外，还可以通过 Unity 编辑器直接打开项目。同时，Unity 编辑器具有向下兼容性，即高版本可以打开低版本创建的项目，但是低版本不能打开高版本创建的项目。

如果想把某个项目从项目列表中移除，则需要单击项目右侧的“...”按钮，执行下拉菜单中的“从列表移除项目”命令，如图 1-3-30 所示。

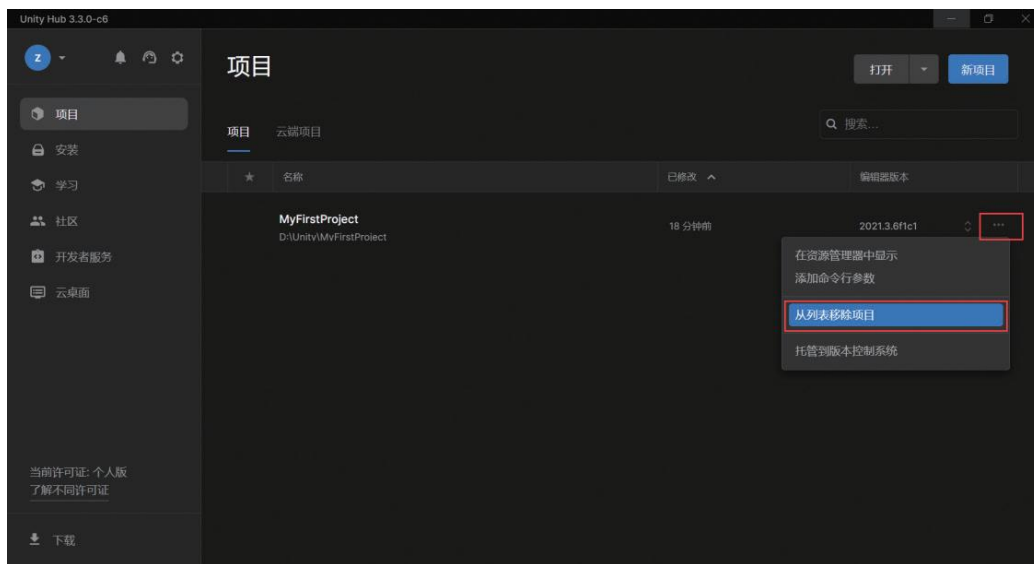


图 1-3-30 从列表中移除项目

如果需要打开的项目不在项目列表中，可以通过单击 Unity Hub 窗口上方的“打开”按钮，或者单击“打开”按钮右侧的下拉菜单图标，选择“从磁盘添加项目”，如图 1-3-31 所示，在弹出的项目选择对话框中将路径切换到待打开项目所在位置，单击“添加项目”即可将项目添加到项目列表并打开项目，如图 1-3-32 所示。

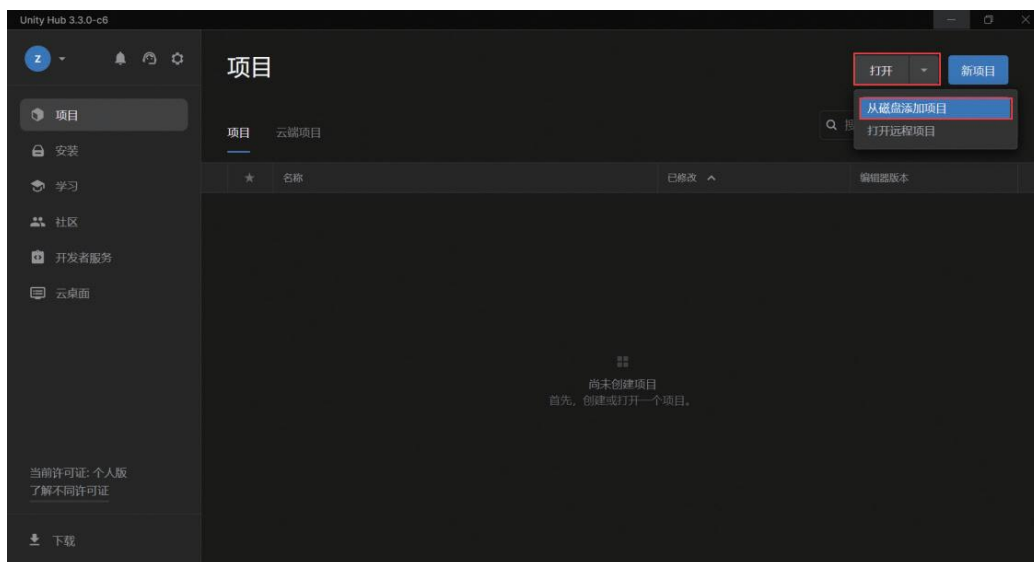


图 1-3-31 打开本地项目

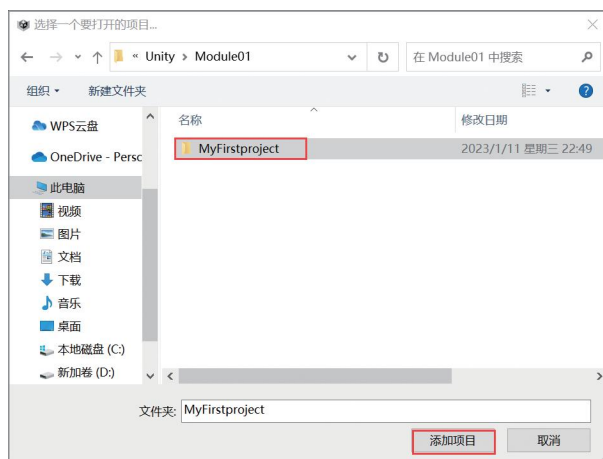


图 1-3-32 切换到项目所在位置

5. 切换中文模式

Unity 编辑器的默认语言是英文，用户可以通过执行菜单栏“Edit”→“Preferences...”命令打开 Preferences 设置窗口，如图 1-3-33 所示。单击窗口左侧的“Languages”选项卡，切换到 Languages 设置界面，单击“Editor Language”右侧的下拉菜单，选择“简体中文”即可切换成中文模式，如图 1-3-34 所示。

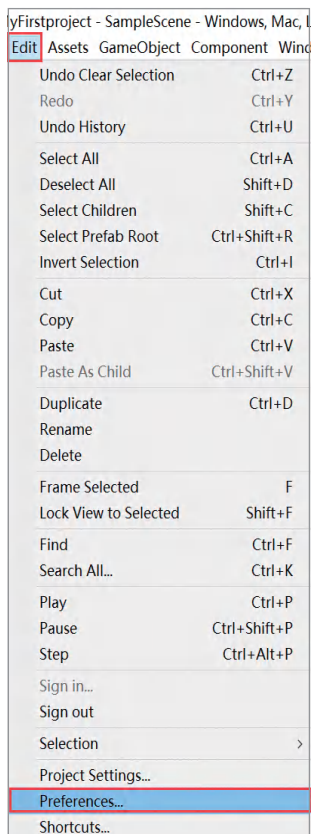


图 1-3-33 打开 Preferences 设置窗口

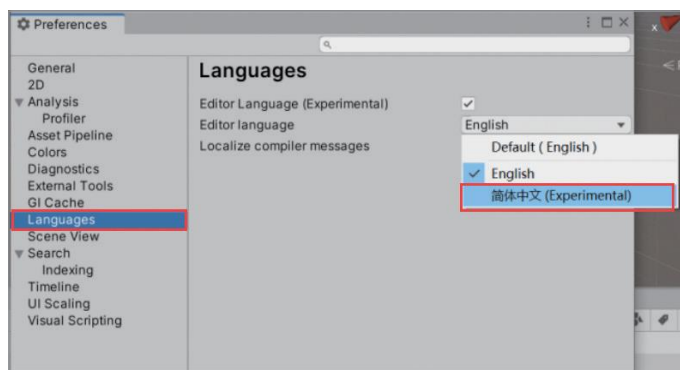


图 1-3-34 切换编辑器语言

小提示

虽然 Unity 提供了不同的语言版本，但是本书讲解采用的是英文模式，为了方便理解，建议读者采用英文模式。

任务 1.4 熟悉 Unity 界面布局及基础操作

Unity 作为一款成熟的商业引擎，具有完善的编辑器，用户可以通过可视化的编辑器界面轻松完成项目的制作。

Unity 的基本界面主要包括 Project 视图、Hierarchy 视图、Scene 视图、Game 视图、Inspector 视图、菜单栏、工具栏等，如图 1-4-1 所示。

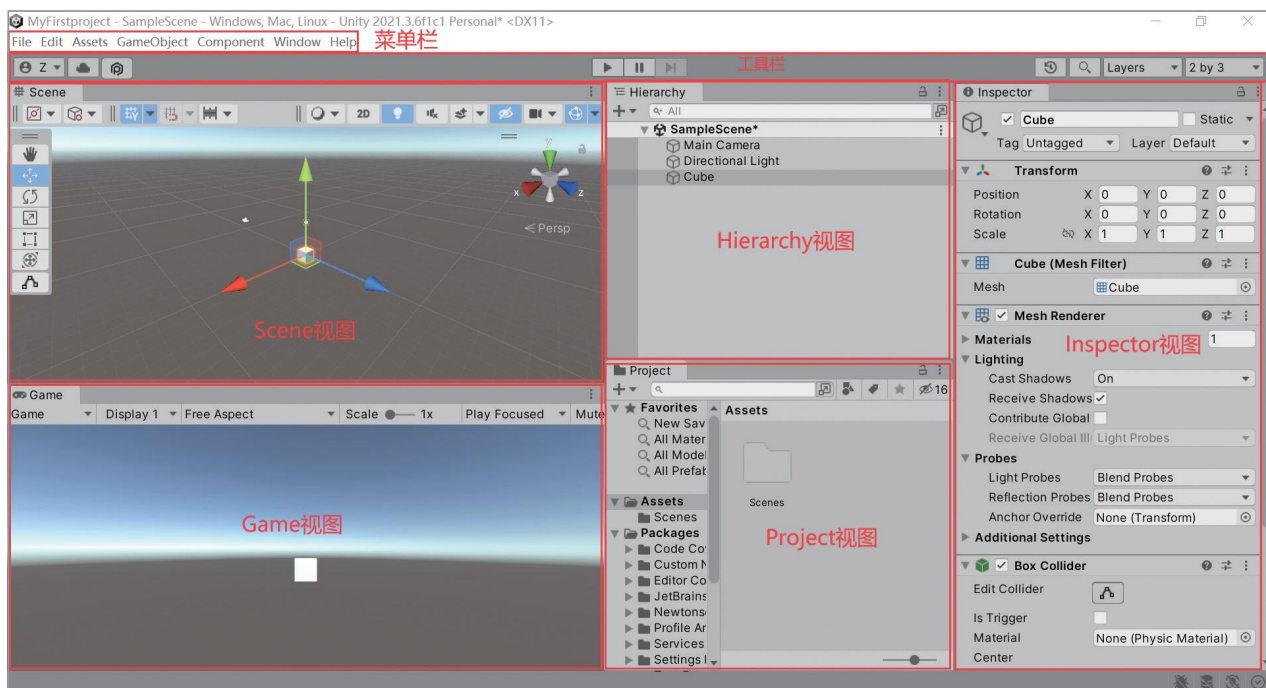


图 1-4-1 Unity 2021.3.6 界面布局

1.4.1 Project 视图

Project 视图显示所有与项目相关的文件，用户可以通过 Project 视图导航和查找资源与其他项目文件。默认情况下启动新项目时，Project 视图将被打开。如果找不到 Project 视图或者它被关闭，可以通过执行菜单栏“Window”→“General”→“Project”命令打开（或使用“Ctrl+5”组合键）它。

Project 视图默认分两栏，如图 1-4-2 所示，左侧面板采用层级列表形式显示项目的文件夹结构，单击下拉菜单图标▾可以展开或折叠文件夹，按住“Alt”键的同时单击下拉菜单图标▾可以以递归的方式展开或折叠所有嵌套文件夹；右侧面板采用图标的形式显示选中文件夹的所有内容，这些图标指示了资源的类型。拖动右侧面板下方的滑动条可以调整图标的显示大小，当滑动条移动到最左侧时，这些图标将切换成层级列表视图。滑动条左侧的空白位置显示当前选定项的路径。

左侧面板中的“Favorites”用于保存经常访问的资源，可以将经常访问的资源从项目结构列表直接拖动到“Favorites”中，也可在“Favorites”中保存搜索查询结果。

左侧面板中的“Assets”文件夹用于管理项目中所有创建和导入的资源文件，包括场景、材质、预物体、脚本等。选中“Assets”文件夹并单击鼠标右键，此时会弹出一个菜单，该菜单与菜单栏中的“Assets”功能相同，用户可以在这里创建自己需要的资源，也可以将外部资源直接拖动到该文件夹中。

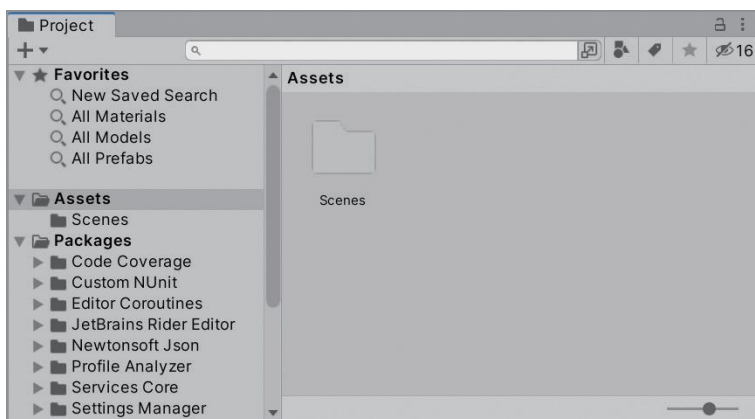


图 1-4-2 Project 视图





单击右侧面板上方的搜索栏，可以快速搜索项目资源。单击搜索框右侧的“Open in Search”图标（或使用“Ctrl+K”组合键）可以打开搜索窗口。单击“Search by Type”图标可以设定搜索类型。单击“Search by Label”图标可以设定搜索标签。单击“Save search”图标可以将搜索结果添加到左侧面板的“Favorites”中，如图 1-4-3 所示。



图 1-4-3 搜索工具

1.4.2 Hierarchy 视图

Hierarchy 视图用来显示场景中的所有游戏对象，如模型、摄像机或预制体等，如图 1-4-4 所示。用户可以通过 Hierarchy 视图对场景中的游戏对象进行排序和分组。在 Scene 视图中添加或删除游戏对象时，Hierarchy 视图中也会相应添加或删除。

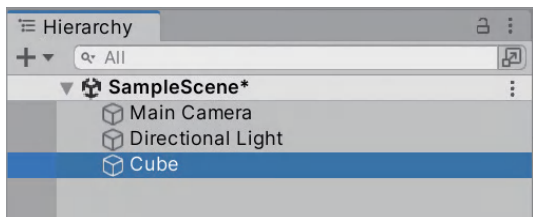


图 1-4-4 Hierarchy 视图

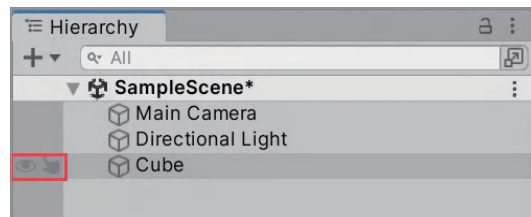






图 1-4-5 场景可视性控件和拾取控件

Unity 通过为游戏对象建立父子关系对游戏对象进行分组，将具有相似功能的对象组织在一起，方便后续的操作与管理，如当移动父游戏对象时，所有子游戏对象也会随之移动。

在 Hierarchy 视图的空白位置单击鼠标右键，可以创建需要的游戏对象。选中某个游戏对象，单击鼠标右键可以为选中对象创建子对象。当鼠标划过 Hierarchy 视图中的游戏对象时，在游戏对象的左侧会出现场景可视性控件图标和拾取控件图标，如图 1-4-5 所示，单击图标可以显示或隐藏该游戏对象，单击图标可以允许或禁止在 Scene 视图中拾取该游戏对象。

单击选中 Hierarchy 视图中的某个游戏对象，按住鼠标左键可以拖动该游戏对象，选择其在 Hierarchy 视图中的位置。当需要将某个游戏对象变成另一个游戏对象的子对象时，则需要将作为子对象的游戏对象拖动到另一个游戏对象上。

1.4.3 Scene 视图

Scene 视图是正在创建的游戏世界或关卡的交互式视图，也是 Unity 编辑器最重要的视图之一，如图 1-4-6 所示。在 Scene 视图中，可以对游戏对象进行选择、操作和修改。

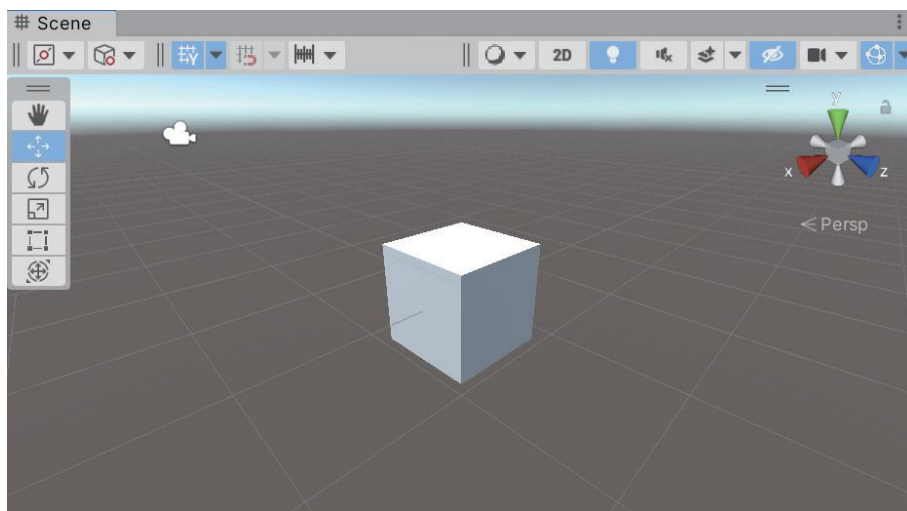


图 1-4-6 Scene 视图

1. Scene 视图导航

(1) 场景辅助工具。

场景辅助工具位于 Scene 视图的右上角，如图 1-4-7 所示，用于表示场景相机的当前方向，通过单击场景辅助工具，可以快速切换观察场景的角度。

单击场景辅助工具下方的“Persp”，可以切换视图模式。Unity 提供了 Persp（透视模式）和 Iso（正交模式）两种视图模式。在 Persp 模式下，物体距离摄像机近时显示大，距离摄像机远时显示小。在 Iso 模式下，不论物体距离摄像机远近显示都一样大，如图 1-4-8 所示。

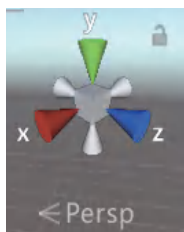


图 1-4-7 场景辅助工具

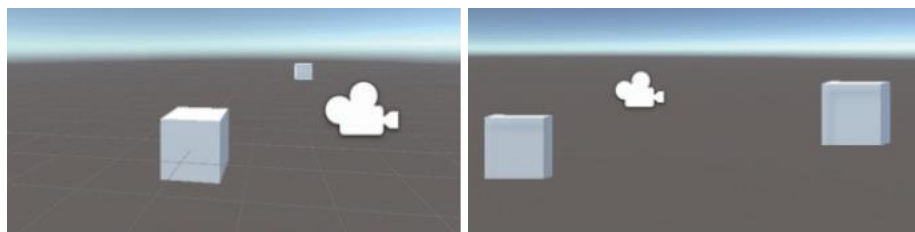


图 1-4-8 两个相同大小的物体在不同视图模式下的显示

场景辅助工具的每个箭头代表不同的几何轴。单击任意一个几何轴，可以将场景相机设置为沿着相应轴移动的正交视图，也可以通过右击场景辅助工具，从弹出的列表中选择要切换的视图，选择“Free”选项可恢复为默认视角，如图 1-4-9 所示。

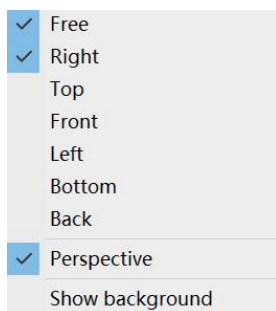


图 1-4-9 Scene 视图辅助工具视图列表

单击场景辅助工具右上角的锁定开关，可以禁止场景旋转。

! 小提示

2D 游戏开发通常采用 Iso 视图模式。

(2) 场景操作工具。

上下左右箭头：控制在场景中移动，向上和向下箭头能够在相机面向的方向向前或向后移动相机，左右箭头横向平移视图。同时按住“Shift”键可以加快移动速度。

鼠标右键+WASD 和 QE 键：单击鼠标右键可以进入飞行相机模式，此时按“W/A/S/D”键可以实现向前/向左/向后/向右移动，按下“Q/E”键可以实现向上/向下移动，同时按住“Shift”键可以加快移动速度。

F 键：选中场景中的游戏对象，按“F”键可以聚焦该对象，功能等同于在 Hierarchy 视图中双击该游戏对象的名称。

Alt+ 鼠标左键：同时按住“Alt”键和鼠标左键并进行拖动，可以旋转摄像头。

Alt+ 鼠标右键：同时按住“Alt”键和鼠标右键并进行拖动，可以放大或缩小场景。

鼠标滚轮：滚动鼠标滚轮可放大或缩小窗口。

2. 选择对象

若需要选择单个游戏对象，可以直接在场景中单击该游戏对象或者在 Hierarchy 视图中选择该游戏对象。

若需要选择多个游戏对象，可以按住鼠标左键进行框选，也可以按住“Shift”键或“Ctrl”键的同时在场景中单击游戏对象，还可以在 Hierarchy 视图中通过按“Shift”键进行连续多选，或者通过按“Ctrl”键进行不连续多选。

当游戏对象处于选中状态时，Scene 视图会突出显示该对象及其子对象。

3. Overlays

(1) 工具 (Tools)。

Unity 的 Overlays 工具栏提供了查看、移动、旋转、缩放、矩形和变换组件 6 个工具，如图 1-4-10 所示。默认显示在场景面板的左上角，用户可以通过拖动将其放在不同的位置。

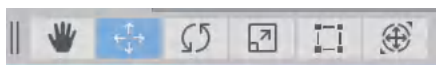


图 1-4-10 Overlays 工具栏

查看工具：查看工具的快捷键为“Q”键，当切换到查看工具后，场景中的鼠标就会变成手型，此时

按住鼠标左键并进行拖动可以移动场景，按住“Alt”键与鼠标左键并进行拖动可以旋转场景，按住“Alt”键和鼠标右键并进行拖动可以缩放场景。

移动工具：移动工具的快捷键是“W”键，当切换到移动工具后，选中场景中的游戏对象，游戏对象上就会出现移动图标，如图 1-4-11 所示。此时选择不同的坐标轴箭头并进行拖动，沿选中坐标轴方向移动游戏对象（红色表示 x 轴，绿色表示 y 轴，蓝色表示 z 轴，为了与软件截图对应，本书中坐标轴均使用正体），也可以选择移动图标中心小立方体的不同平面（红色表示 YZ 平面，绿色表示 XZ 平面，蓝色表示 XY 平面），在选中平面内移动游戏对象。

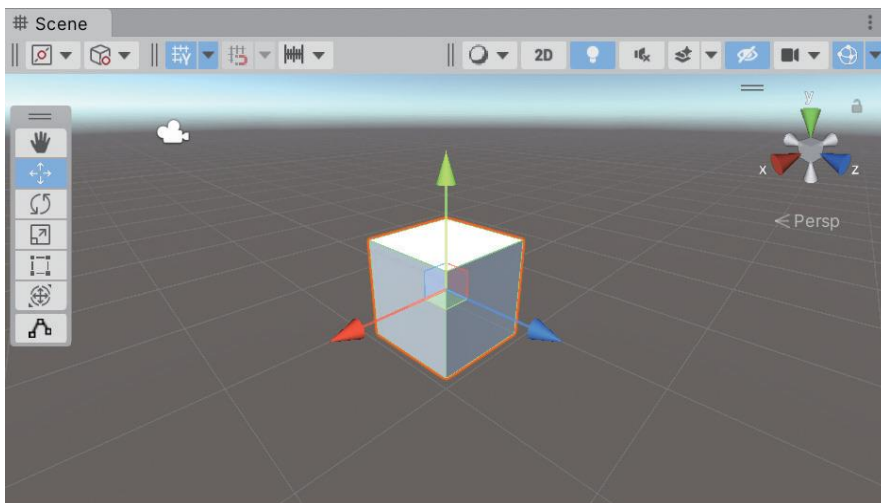


图 1-4-11 移动图标

旋转工具：旋转工具的快捷键是“E”键，当切换到旋转工具后，选中场景中的游戏对象，游戏对象上就会出现旋转图标，如图 1-4-12 所示。此时可通过单击并拖动围绕游戏对象显示的线框球体辅助图标的轴来旋转游戏对象。

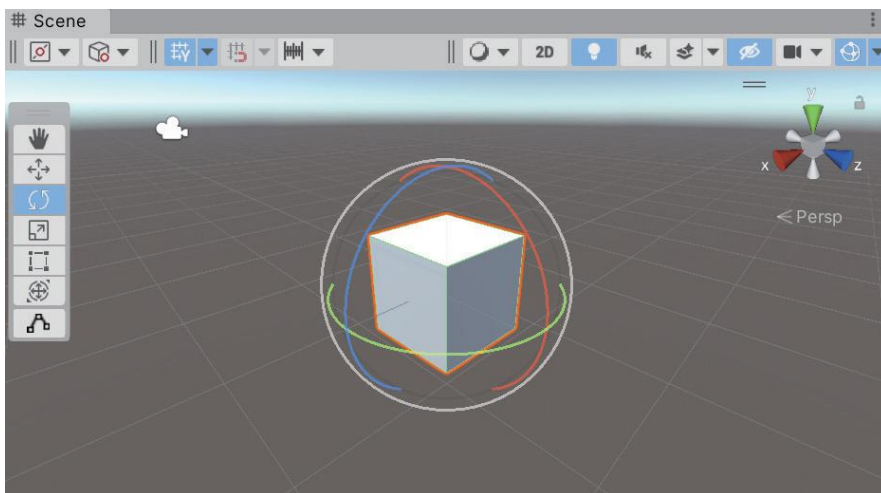


图 1-4-12 旋转图标

缩放工具：缩放工具的快捷键是“R”键，当切换到缩放工具后，选中场景中的游戏对象，游戏对象上就会出现缩放图标，如图 1-4-13 所示。此时选择缩放图标中心的立方体并进行拖动，可整体放大或缩小游戏对象。单击选择其中某一个轴并进行拖动，可沿该轴方向放大或缩小游戏对象。

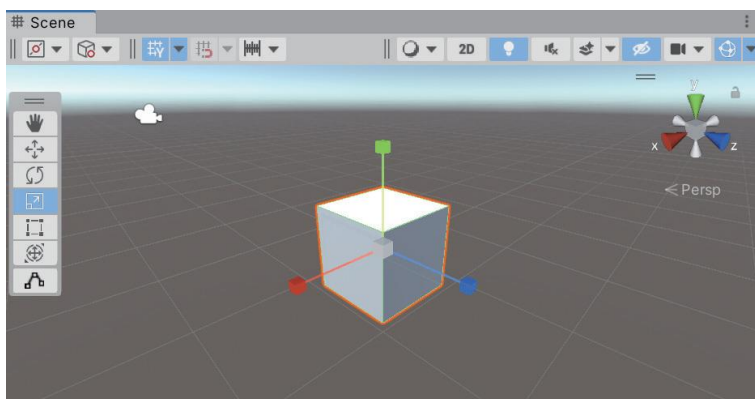



图 1-4-13 缩放图标

矩形工具 ：矩形工具的快捷键是“T”键，通常用于操作 2D 元素（如精灵或 UI 元素），但也可用于操作 3D 游戏对象。当切换到矩形工具后，单击场景中的游戏对象，游戏对象上就会出现矩形工具图标，如图 1-4-14 所示，在矩形工具图标中单击并拖动可移动游戏对象。单击并拖动矩形工具图标的任意角或边可缩放游戏对象，将光标放在矩形的某个角之外，当光标变为旋转图标时单击并拖动可旋转游戏对象。

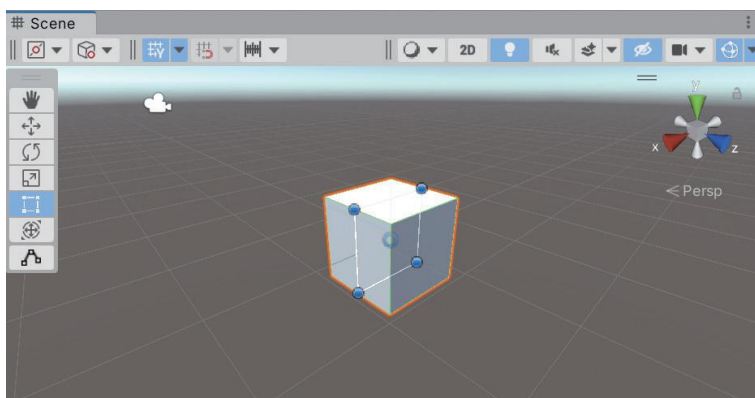



图 1-4-14 矩形图标

变换组件工具 ：变换组件工具的快捷键是“Y”键，它组合了移动、旋转和缩放三个工具的功能，如图 1-4-15 所示。

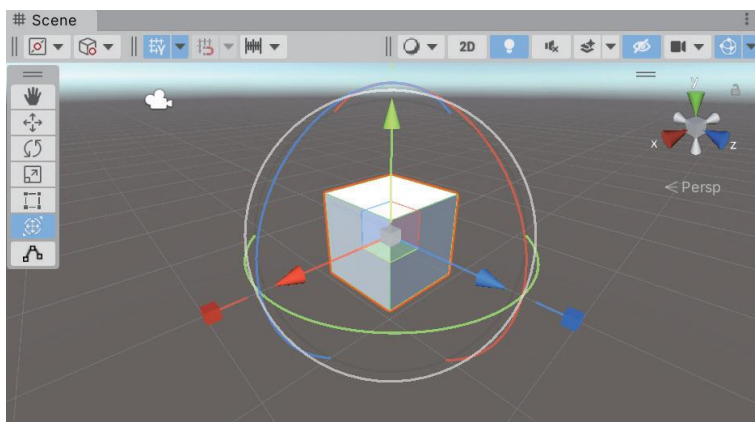


图 1-4-15 变换组件图标

(2) 工具设置 (Tools Settings)。

Tools Settings 包含 Toggle Tool Handle Position (切换工具手柄位置) 和 Toggle Tool Handle Rotation (切换工具手柄旋转) 两个工具, 如图 1-4-16 所示。

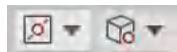



图 1-4-16 Tools Settings

Toggle Tool Handle Position : 当选择 Pivot 模式时, 操作原点定位在游戏对象的实际坐标原点, 当操作对象有子对象时, 操作原点为父对象的坐标原点, 如图 1-4-17 所示。当选择 Center 模式时, 操作原点定位在游戏对象的中心位置, 当操作对象有子对象时, 操作原点为该对象及其子对象共同的中心点, 如图 1-4-18 所示。

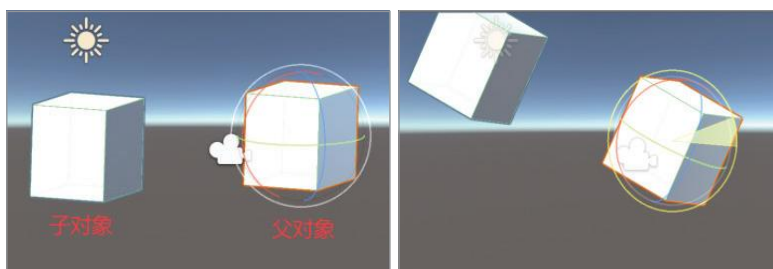


图 1-4-17 Pivot 模式

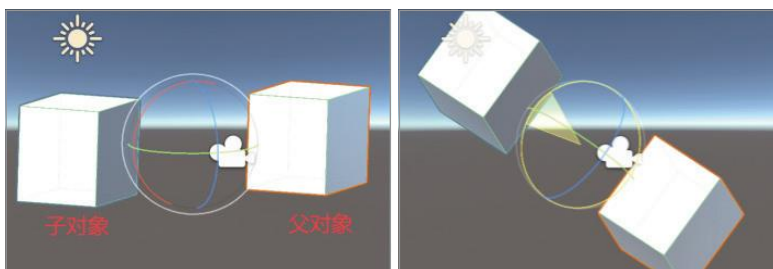



图 1-4-18 Center 模式

Toggle Tool Handle Rotation : 当选择 Global 模式时, 会忽略游戏对象的旋转, 此时进行移动操作, 游戏对象将沿世界坐标系坐标轴方向移动, 如图 1-4-19 所示。当选择 Local 模式时, 会使用游戏对象的旋转, 此时进行移动操作, 游戏对象将沿自身坐标系坐标轴方向移动, 如图 1-4-20 所示。

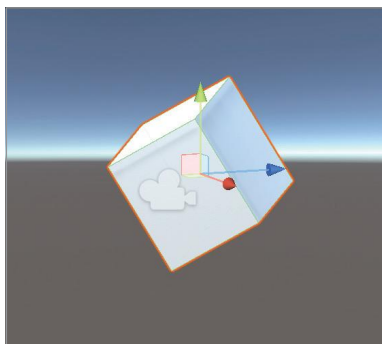


图 1-4-19 Global 模式

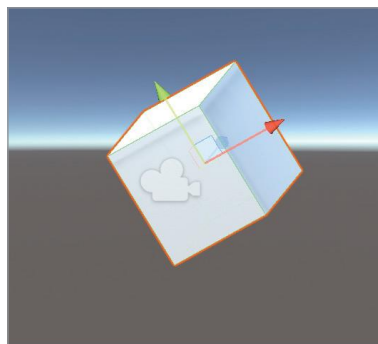


图 1-4-20 Local 模式

(3) 视图选项 (View Options)。

View Options 主要用于设置场景的查看方式、启用 / 禁用照明和音频等，如图 1-4-21 所示。

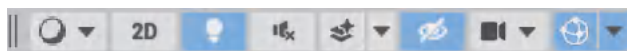



图 1-4-21 View Options 工具

Shading Mode ：控制场景中游戏对象的绘制模式，默认为 Shaded 模式，除此之外 Unity 还支持 Wireframe 模式和 Shaded Wireframe 模式，如图 1-4-22 所示。

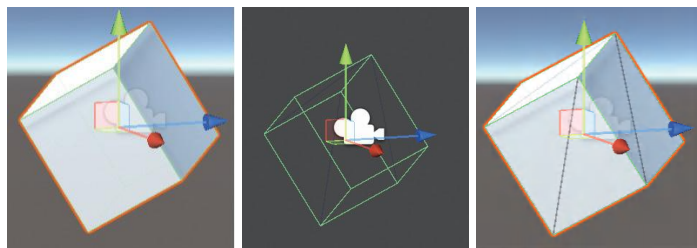





图 1-4-22 Shaded、Wireframe 和 Shaded Wireframe 模式

2D ：控制场景显示为 2D 视图还是 3D 视图。在 2D 视图模式下，摄像机朝向 z 轴正方向，x 轴指向右方，y 轴指向上方。


场景照明 ：打开或关闭 Scene 视图光照（光源、对象着色等），默认是打开的，如果取消，场景会变得暗淡。

场景音频 ：打开或关闭 Scene 视图音频效果。

渲染效果 ：启用或禁用 Scene 视图的渲染特效，如天空盒、雾等其他特效。

场景可见性 ：切换场景中隐藏对象的可见性。

场景相机 ：设置场景相机参数。

Gizmos 菜单 ：启用或隐藏 Scene 视图的一些图标。如摄像机图标、光源图标等，如图 1-4-23 所示。

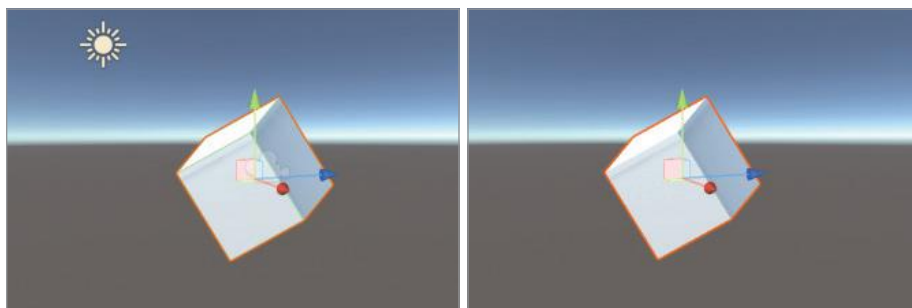


图 1-4-23 打开和隐藏 Gizmos 图标

1.4.4 Game 视图

Game 视图用来呈现游戏的最终运行效果，方便游戏开发人员进行测试，如图 1-4-24 所示。

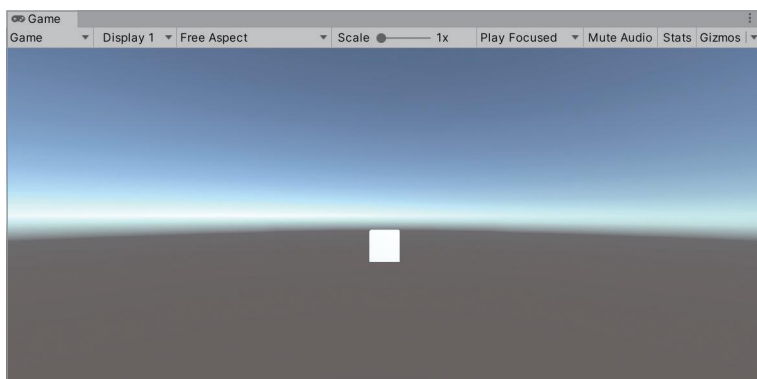


图 1-4-24 Game 视图

1. Game/Simulator Game

切换 Game 和 Simulator（模拟器）视图模式，如图 1-4-25 所示。



图 1-4-25 Simulator 视图

2. 显示屏 Display 1

当场景中有多个摄像机时，通过此按钮可设置显示不同的摄像机画面，默认值为 Display 1。可以在摄像机目标显示下拉菜单中指定显示屏。

3. Free Aspect Free Aspect

设置分辨率，游戏开发者可以通过设置不同分辨率测试游戏在不同设备上的显示效果。默认值为 Free Aspect。

4. 缩放 Scale 1x

放大 Game 视图，方便游戏开发者观察游戏运行结果。

5. 播放聚焦模式 Play Focused

当选择 Play Focused 模式时，运行游戏会自动聚焦 Game 视图；当选择 Play Maximized 模式时，运行游戏将全屏显示 Game 视图；当选择 Play Unfocused 模式时，在运行游戏时不会聚焦 Game 视图。

6. 音频静音 Mute Audio

设置运行状态下游戏中的音频是否静音。

7. 状态 Stats

单击此按钮，弹出 Statistics 面板，显示当前运行场景的性能信息。

1.4.5 Inspector 视图

Inspector 视图用于显示和编辑游戏对象、游戏资源的组件参数，如图 1-4-26 所示，其显示内容会根据选定对象的不同而变化。

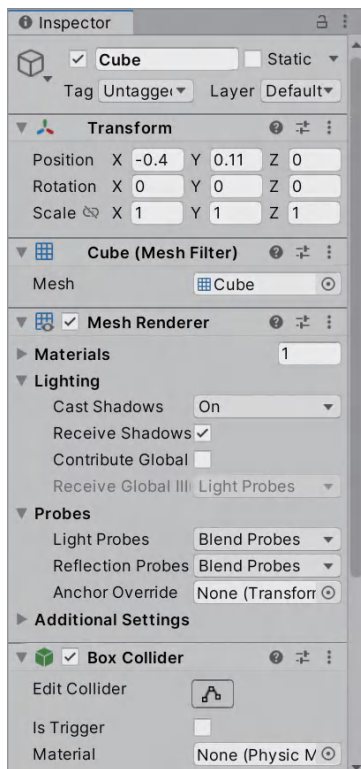



图 1-4-26 Inspector 视图

在 Inspector 视图中，每个组件都有相应的帮助图标，单击帮助图标会自动打开 Unity 手册中关于该组件的介绍页面，方便游戏开发者了解组件各个参数的含义。

1.4.6 菜单栏

Unity 的菜单栏由文件、编辑、资源、游戏对象、组件、窗口和帮助 7 个菜单项组成，菜单栏中几乎涵盖了 Unity 编辑器的全部功能。

1. File (文件) 菜单

File (文件) 菜单主要用于新建、打开和保存场景与项目，以及项目的生成设置和构建与运行，其子菜单及对应快捷键如图 1-4-27 所示。

2. Edit (编辑) 菜单

Edit (编辑) 菜单主要用于对场景中游戏对象的基本操作（如撤销、剪切、复制、粘贴等）以及项目的相关设置等，其子菜单及对应快捷键如图 1-4-28 所示。

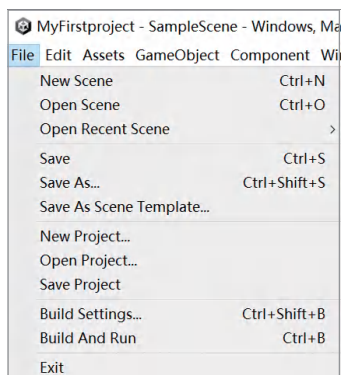


图 1-4-27 File 子菜单

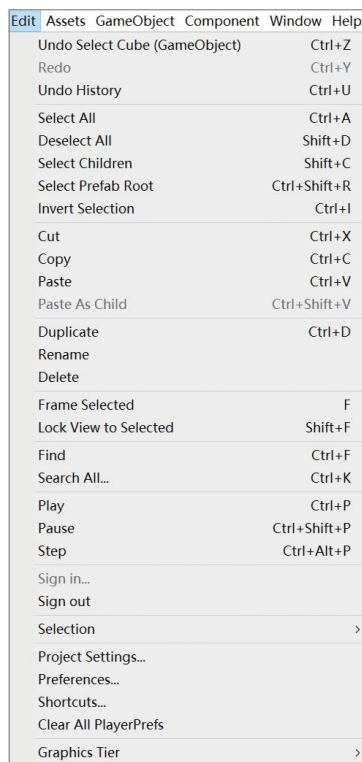


图 1-4-28 Edit 子菜单

3. Assets (资源) 菜单

Assets (资源) 菜单主要用于资源的创建、打开、删除、导入和导出，其子菜单及对应快捷键如图 1-4-29 所示。

4. GameObject (游戏对象) 菜单

GameObject (游戏对象) 菜单主要用于创建 3D 物体、UI、灯光、音频和视频等游戏对象，其子菜单及对应快捷键如图 1-4-30 所示。

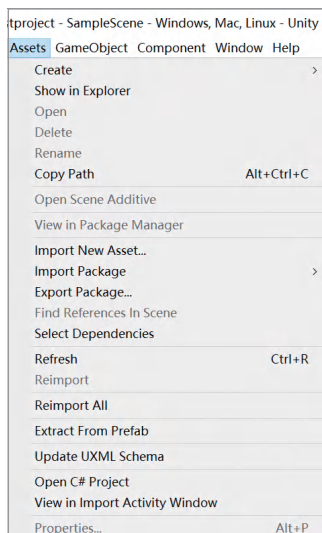


图 1-4-29 Assets 子菜单

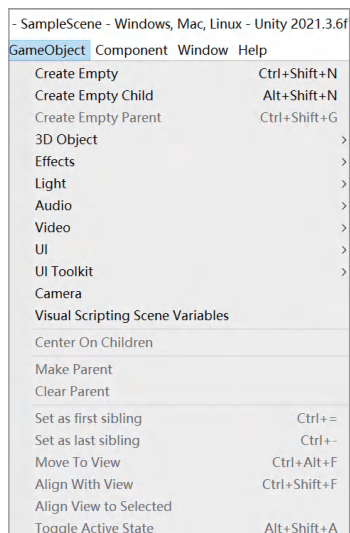


图 1-4-30 GameObject 子菜单

5. Component (组件) 菜单

Component (组件) 菜单主要用于为游戏对象等添加网格、效果、物理、音频等组件，并以此来实现相应功能，其子菜单及对应快捷键如图 1-4-31 所示。

6. Window (窗口) 菜单

Window (窗口) 菜单主要用于显示场景及游戏的 Unity 主视图、设置窗口布局、打开资源商店等，其子菜单及对应快捷键如图 1-4-32 所示。

7. Help (帮助) 菜单

Help (帮助) 菜单主要包含 Unity 用户手册、脚本参考、Unity 论坛和管理许可证等功能，其子菜单及对应快捷键如图 1-4-33 所示。

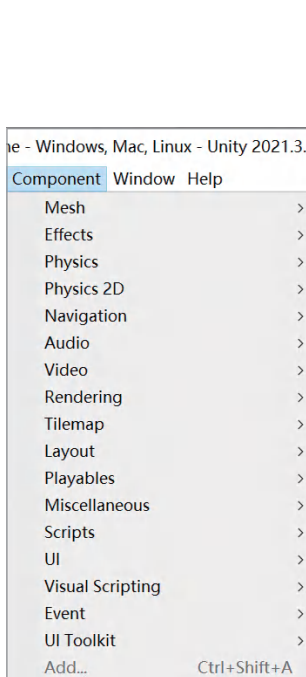


图 1-4-31 Component 子菜单

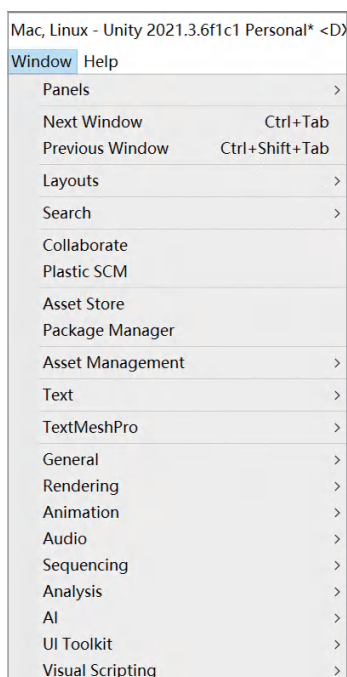


图 1-4-32 Window 子菜单

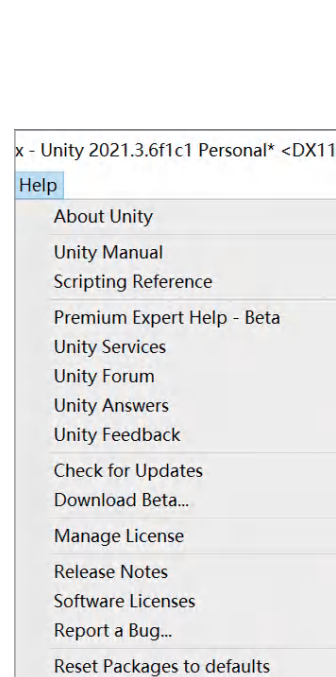


图 1-4-33 Help 子菜单

1.4.7 工具栏

Unity 的菜单栏下面就是工具栏，工具栏不是窗口，是 Unity 界面中唯一无法重新排列的部分。

Unity 工具栏提供了 Unity 账号服务，Manage Services，Plastic SCM，Play、Pause 和 Step，Undo History，Global Search，Layer 下拉列表和 Layout 下拉列表功能。

1. Unity 账号服务

单击下拉三角可实现账号登录和注销，查看账号信息以及升级至 Unity 增强版和专业版。

2. Manage Services

打开服务面板（具有 Unity 项目 ID 时可用）。

3. Plastic SCM

当登录账号后，单击此按钮可以使用 Unity 的项目版本控制系统。

4. Play

运行游戏项目。注意：当游戏处于运行状态时，对场景做的所有操作不会被保存。

5. Pause

暂停游戏项目。

6. Step

逐帧运行游戏项目。



7. Undo History

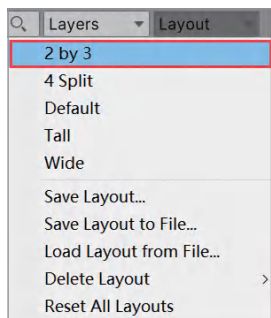
显示所有撤销的历史记录。

8. Global Search

与编辑菜单栏中的搜索全部功能相同。

9. Layer 下拉列表

设置场景中的显示对象，单击图标  会开启或关闭对应层级下游戏对象的显示，单击图标  会开启或禁用 Scene 视图中选取对应层级下游戏对象的功能。



10. Layout 下拉列表

设置窗口布局，Unity 内置了 2×3、4 分屏、默认、高和宽 5 种常用的窗口布局，除此之外还可以自定义布局样式。

例如，设置一款在开发项目时便于观察实际运行效果的布局样式，则需要先在 2×3 布局模式的基础上稍作改动。首先单击工具栏 Layout 下拉列表中的“2 by 3”选项，将视图切换成 2×3 模式，如图 1-4-34 所示。

单击 Project 视图，将其拖拽到 Hierarchy 视图下方。当将光标移动到两个视图交界处时，光标变成上下箭头形状，调整 Project 视图的高度，效果如图 1-4-35 所示。

图 1-4-34 切换布局模式

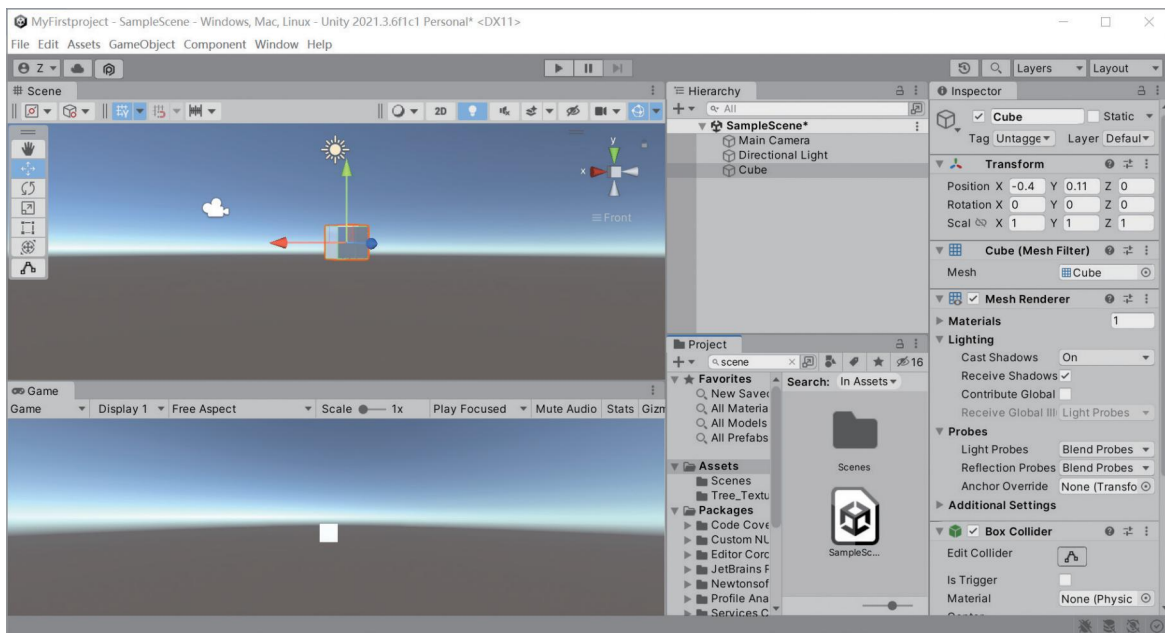


图 1-4-35 设置自定义布局模式

任务 1.5 综合实践——创建时钟

前面学习了 Unity 编辑器的界面布局和基本操作，接下来以创建一个时钟为例，讲解 Unity 编辑器的具体使用方法。

1.5.1 创建表盘

首先创建一个 Unity 3D 项目并将其命名为“Clock Project”，然后在 Hierarchy 视图中单击鼠标右键创建一个空对象，命名为“Clock”，如图 1-5-1 所示。

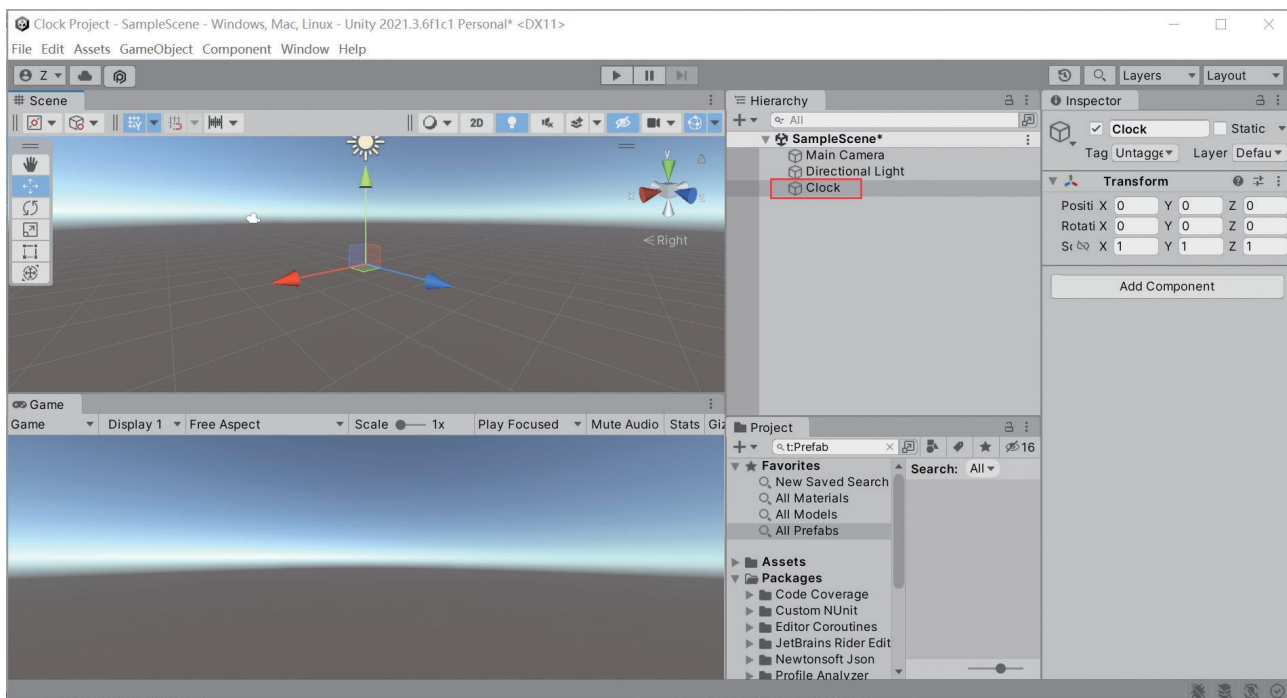


图 1-5-1 创建“Clock”空对象

选中“Clock”，单击 Inspector 视图中 Transform 组件右侧的图标，重置位置、旋转和缩放的数值，如图 1-5-2 所示。

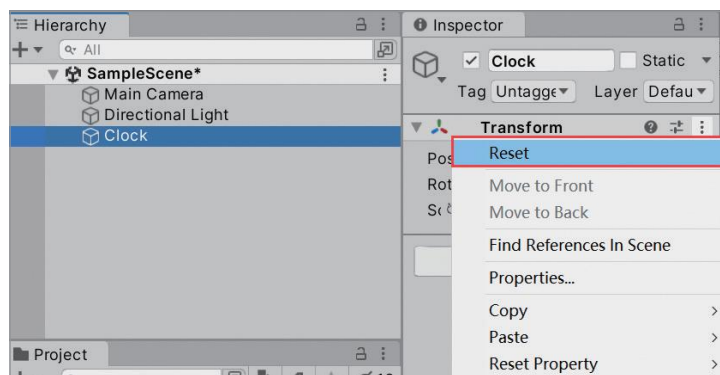


图 1-5-2 重置“Clock”对象的 Transform 组件

在 Hierarchy 视图中选择“Cube”选项，单击鼠标右键执行“3D Object”→“Cylinder”命令，并将其命名为 Clock Border，作为时钟的边框。将 Scene 视图切换为 Back 视图，选中“Clock Border”选项，修改其“Transform”组件中的“Rotation”属性为（90，0，0），“Scale”属性为（5，0.1，5），如图 1-5-3 所示。



图 1-5-3 设置“Clock Border”对象的“Transform”组件

在 Project 视图中选择“Assets”文件夹，单击鼠标右键执行“Create”→“Folder”命令创建文件夹，将其命名为“Materials”，用来存储材质。双击打开“Materials”文件夹，单击鼠标右键执行“Create”→“Material”命令创建材质，将其命名为“Black”。单击 Inspector 视图中“Albedo”属性右侧的白色色块，打开“Color”面板，将颜色设置为黑色，如图 1-5-4 所示。

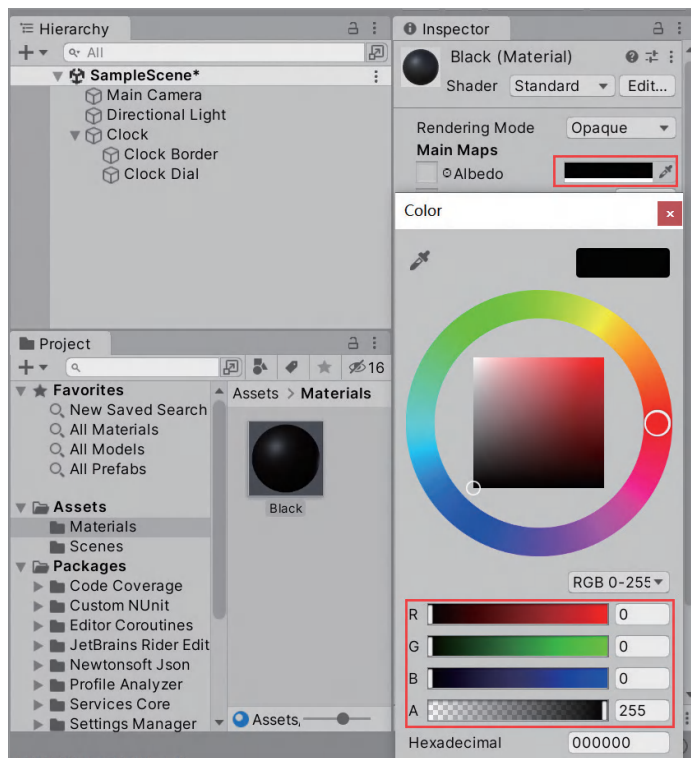


图 1-5-4 创建黑色材质

选中 Black 材质，按住鼠标左键将其拖动到 Hierarchy 视图中的 Clock Border 上，此时 Clock Border 变成了黑色。

在 Hierarchy 视图中选择“Clock Border”选项，按“Ctrl+D”组合键进行复制和粘贴并将其命名为“Clock Dial”，作为时钟的表盘，并修改其“Position”属性中的“Z”坐标为 -0.01，使表盘稍微浮出边框，修改其“Scale”属性为（4.8，0.1，4.8），使其稍小于边框，如图 1-5-5 所示。

再次在 Materials 中创建一个材质并将其命名为“White”，将其“Albedo”属性的颜色调整为白色，并拖动到 Hierarchy 视图中的“Clock Dial”对象上，如图 1-5-6 所示。



图 1-5-5 修改“Clock Dial”属性



图 1-5-6 为“Clock Dial”添加白色材质

1.5.2 创建时间刻度

在 Hierarchy 视图中选择“Clock”选项，单击鼠标右键创建空对象并将其命名为 Clock Scale，将其作为时间刻度，重置其 Transform 组件。选择“Clock Scale”选项，单击鼠标右键执行“3D Object”→“Cube”命令创建对象，将其命名为 Scale，修改其“Position”属性为 (2.05, 0, -0.1)，“Scale”属性为 (0.4, 0.08, 0.15)，如图 1-5-7 所示。然后选中“Black”材质，并将其拖拽到 Scale 对象上。

选择“Clock Scale”选项，按“Ctrl+D”组合键复制 11 个副本，作为时间刻度，如图 1-5-8 所示。

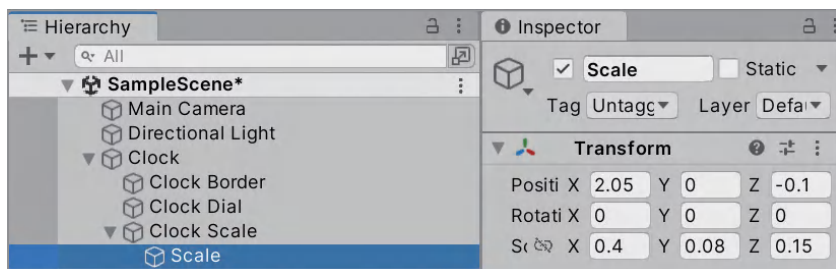


图 1-5-7 创建“Scale”并设置其 Transform 组件属性

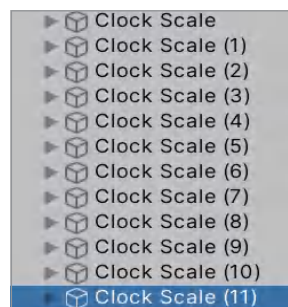


图 1-5-8 复制“Clock Scale”

选中创建好的 12 个刻度对象，在 Inspector 视图中修改“Rotation”属性中的“Z”值为 L (0, 330)，如图 1-5-9 所示，通过线性插值函数实现各个刻度对象的旋转，然后按照刻度指向的具体时间修改其命名，并通过拖动将其刻度按照从小到大排序。为了方便管理，可以在 Hierarchy 视图中选中 12 个时间刻度，单击鼠标右键选择“Create Empty Parent”选项，为所有刻度对象创建一个父对象，并将其命名为“Clock Scales”。

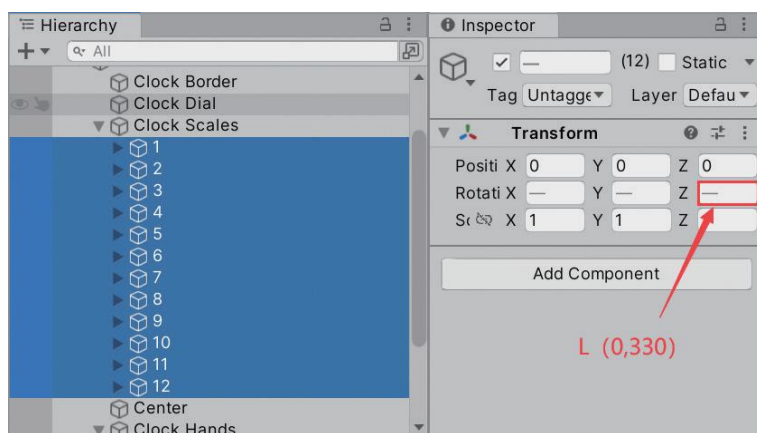


图 1-5-9 通过插值函数实现刻度旋转

！ 小提示

$L(a, b)$ 是 Unity 中的线性插值函数，可实现在 a 到 b 之间的线性插值。例如上例中的 $L(0, 330)$ 会生成 0, 30, 60, 90,300, 330 的等差数列。

1.5.3 创建表针

再次选择“Clock”选项，创建一个 Cylinder 并将其命名为“Center”，作为表针的固定器，修改其“Position”属性为 (0, 0, -0.18)、“Rotation”属性为 (90, 0, 0)、“Scale”属性为 (0.15, 0.01, 0.15)，并将 Black 材质拖动到“Center”上，如图 1-5-10 所示。

再次选择“Clock”选项，创建一个空对象并将其命名为“Clock Hands”，用来管理所有表针。选择“Clock Hands”选项，为其创建一个空子对象并将其命名为“Hour Hand”，重置其 Transform 组件。选择“Hour Hand”选项，为其创建一个 Cube 子对象，将其命名为“Pointer”，作为时针，修改其“Position”属性为 (0.38, 0, -0.13)、“Scale”属性为 (1.2, 0.07, 0.07)，并将 Black 材质拖动到 Pointer 上，如图 1-5-11 所示。



图 1-5-10 创建表针固定器



图 1-5-11 创建时针

选择“Hour Hand”选项，按“Ctrl+D”组合键复制一个副本，并将其名称修改为“Minute Hand”，修改其“Rotation”属性为 (0, 0, 90)。选中“Minute Hand”的子对象“Pointer”，修改其“Position”属性为 (0.5, 0, -0.14)、“Scale”属性为 (1.5, 0.04, 0.04)，如图 1-5-12 所示。

再次复制“Hour Hand”，并将其命名为“Second Hand”，修改其“Rotation”属性为 (0, 0, -120)。选中“Second Hand”的子对象“Pointer”，修改其“Position”属性为 (0.56, 0, -0.15)、“Scale”属性为 (1.8, 0.025, 0.025)，如图 1-5-13 所示。



图 1-5-12 创建分针



图 1-5-13 创建秒针

选择 Project 视图的 Materials 文件夹，创建一材质并将其命名为“Red”，修改“Albedo”属性的颜色为红色，并将其拖动到“Second Hand”的子对象“Pointer”上，此时钟表就做好了，单击“Play”按钮运行场景，运行效果如图 1-5-14 所示。



图 1-5-14 运行效果

想一想

如果需要将时钟的指示时间调整为 8:30，应该怎么设置表针的旋转角度呢？

任务 1.6 Unity 项目发布

1.6.1 设置文件图标和鼠标

前面已经完成了钟表的制作，接下来进行项目发布前的设置。首先在“Assets”文件夹中新建一个文件夹并将其命名为“Images”，将随书资源 Module01 中的“Cursor.png”和“Clock.jpg”拖入 Images 文件夹，如图 1-6-1 所示。

选中导入的“Cursor.png”，在 Inspector 视图中将其“Texture Type”（纹理类型）属性修改为“Cursor”，单击“Apply”按钮完成转换，如图 1-6-2 所示。

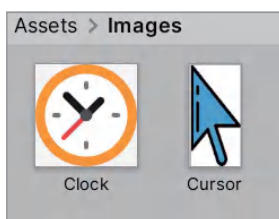


图 1-6-1 导入素材

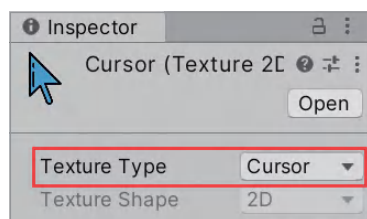


图 1-6-2 修改光标纹理类型

单击菜单栏执行“File”→“Build Settings...”→“Player Settings...”命令或单击菜单栏执行“Edit”→“Project Settings”→“Player”命令，打开项目设置窗口，“Player”具有以下几个基础参数：

Company Name: 公司名称，用户可以将其设置为自己所在公司的名称。

Product Name: 产品名称。

Version: 版本号，首次发布时通常设置为 1.0。

Default Icon: 默认图标。

Default Cursor: 默认光标。

Cursor Hotspot: 用来设置从默认光标左上角到光标热点位置的偏移值（以像素为单位）。