

# Unity 4.x

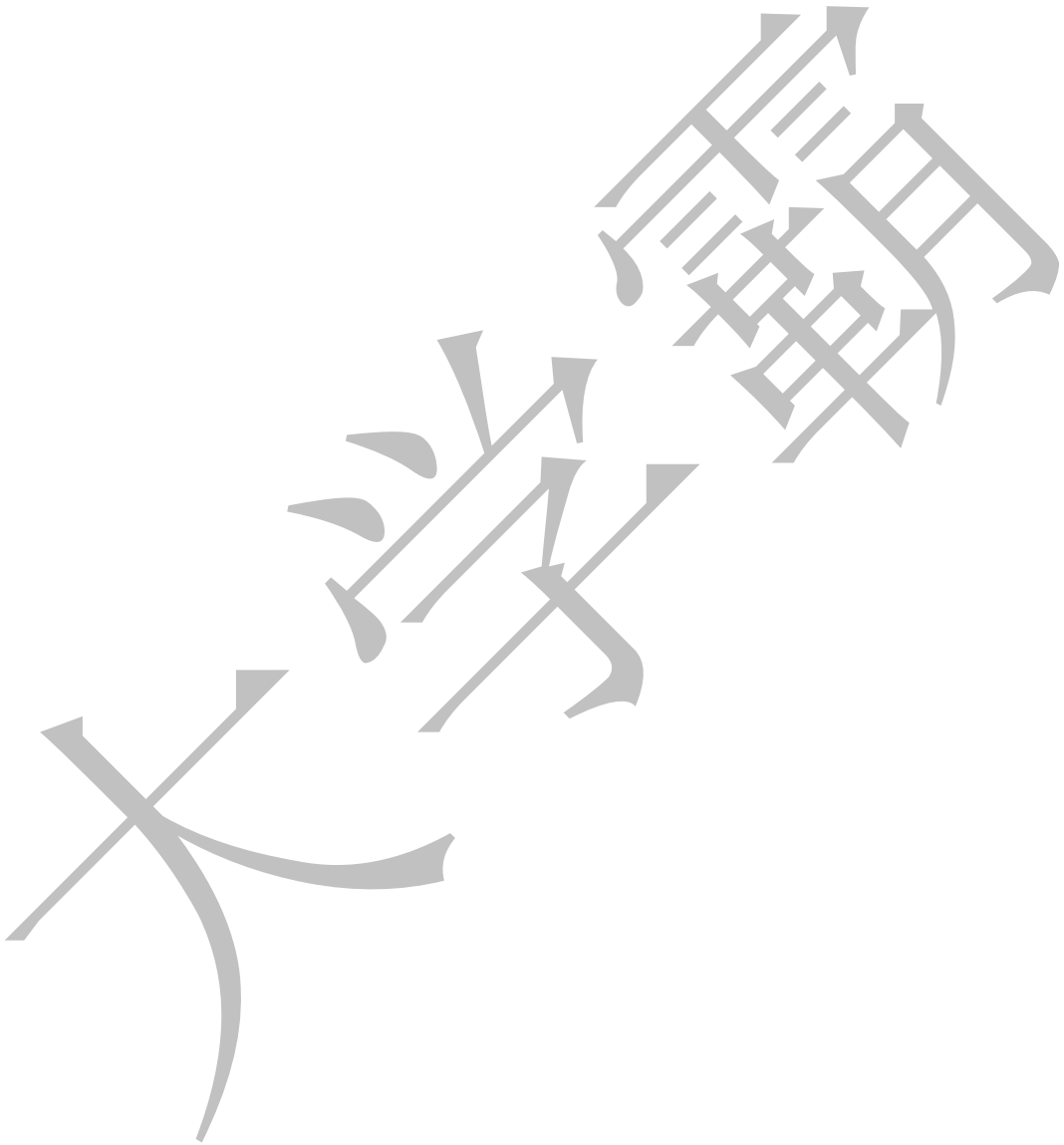
## 游戏开发技巧集锦

（内部资料）



大学霸

[www.daxueba.net](http://www.daxueba.net)



## 前 言

Unity 是一款世界知名的游戏开发工具，也是一款全面整合的专业游戏引擎。使用 Unity 开发的游戏，可以部署到所有的主流游戏平台，而无需任何修改，如 Windows、Linux、Mac OS X、iOS、Android、Xbox 360、PS3、WiiU 和 Web 等。据权威机构统计，国内 53.1% 的人使用 Unity 进行游戏开发；有 80% 的手机游戏是使用 Unity 开发的；苹果应用商店中，有超过 1500 款游戏使用 Unity 开发。

强大的工具还需要灵活的应用。现在的游戏种类繁多，其中的声光效果更是精彩炫目。作为游戏开发初学者，往往被别人的游戏效果和功能所惊叹，但往往又为自己的游戏所汗颜。

本书分析世界各类知名游戏，如《仙剑奇侠传》、《红警》、《使命召唤》、《穿越火线》、《劲舞团》、《极品飞车》、《斗地主》、《植物大战僵尸》、《天天跑酷》等。从这些游戏中选择大量经典应用功能和特效进行讲解，如：

- 《极品飞车》的后视镜功能
- 《红警警戒》的士兵巡逻功能
- 《荣誉勋章》的罗盘功能
- 《拳皇》的倒计时功能
- 《超级玛丽》的消失文字效果
- 《星际争霸》的士兵响应效果

相信读者从中学到的将不只是各种特效的实现方法，还会从中感受到无限的成就感和欢乐。

### 1. 学习所需的系统和软件

- ☐ 安装 Windows 7 操作系统
- ☐ 安装 Unity 4.5.2

### 2. 学习建议

大家学习之前，可以致信到 [XXXXXXXXXX](mailto:XXXXXXXXXX)，获取相关的资料 and 软件。如果大家在学习过程遇到问题，也可以将问题发送到该邮箱。我们尽可能给大家解决。



# 目录

第 1 章 熟悉Unity及其简单操作.....	1
1.1 安装Unity.....	1
1.2 编辑器的偏好设置.....	4
1.3 熟悉Unity的编辑器界面.....	5
1.4 将Unity中的资源保存到预设体中.....	8
1.5 使用Unity内置的资源包.....	10
1.6 导入自己的资源.....	11
1.7 导出Unity中的资源.....	11
1.8 导入自己的资源包.....	12
1.9 添加资源包到资源包列表中.....	13
1.10 使用Project视图检索器.....	13
第 2 章 摄像机的应用.....	15
2.1 设置双游戏视图.....	15
2.1.1 环境准备.....	15
2.1.2 编写脚本.....	16
2.1.3 实现效果.....	18
2.2 在多个游戏视图间切换.....	19
2.2.1 环境准备.....	19
2.2.2 编写脚本.....	19
2.2.3 实现效果.....	20
2.3 制作镜头光晕效果.....	21
2.4 制作游戏的快照.....	24
2.5 制作一个望远镜.....	27
2.6 制作一个查看器摄像机.....	30
2.7 使用忍者飞镖创建粒子效果.....	34
2.7.1 粒子基本属性.....	34
2.7.2 粒子的值.....	34
2.7.3 创建粒子效果.....	35
2.7.4 了解粒子系统的初始化模块.....	36
第 3 章 材质的应用.....	39
3.1 创建反射材质.....	39
3.2 创建自发光材质.....	41
3.2.1 创建并配置材质.....	41
3.2.2 制作应用于发光材质的纹理.....	42
3.2.3 效果展示.....	44
3.3 创建部分光滑部分粗糙的材质.....	45
3.3.1 创建并配置材质.....	45
3.3.2 制作兼具光滑和粗糙效果的纹理.....	46
3.3.3 效果展示.....	47
3.4 创建透明的材质.....	48

3.4.1	创建并配置材质.....	48
3.4.2	制作有透明效果的纹理.....	48
3.4.3	效果展示.....	49
3.5	使用cookie类型的纹理模拟云层的移动.....	50
3.5.1	制作云层效果的纹理.....	50
3.5.2	在Unity中完成的准备工作.....	51
3.5.3	编写控制云层移动的脚本.....	52
3.5.4	效果展示.....	53
3.6	制作一个颜色选择对话框.....	53
3.7	实时合并纹理——拼脸小示例.....	56
3.8	创建高亮材质.....	59
3.9	使用纹理数组实现动画效果.....	61
3.10	创建一个镂空的材质.....	64
第4章	GUI的应用.....	67
4.1	绘制一个数字时钟.....	67
4.2	制作一个模拟时钟.....	68
4.3	制作一个罗盘.....	71
4.4	使用雷达说明对象的相对位置.....	74
4.5	在游戏视图上显示指定数量的纹理.....	77
4.6	使用不同的纹理表示数值.....	79
4.7	显示一个数字倒计时.....	82
4.8	显示一个图片数字倒计时.....	83
4.9	显示一个饼状图倒计时.....	85
4.10	逐渐消失的文字信息.....	88
4.11	显示一个文字财产清单.....	89
4.12	显示一个图片财产清单.....	91
4.13	丰富图片清单的内容.....	93
4.14	允许鼠标滚轮控制滚动条的滚动.....	96
4.15	使用自定义鼠标取代系统鼠标.....	98
第5章	Mecanim动画系统的应用.....	102
5.1	给人物模型加Avatar和动画.....	102
5.1.1	添加Avatar.....	102
5.1.2	添加动画.....	104
5.1.3	添加动画控制器.....	106
5.1.4	人物模型动作效果展示.....	106
5.1.5	将动画应用于其它的人物模型.....	107
5.2	自由控制人物模型做各种动作.....	108
5.2.1	人物模型以及动画属性设置.....	109
5.2.2	动画控制器的设置——添加混合树.....	111
5.2.3	动画控制器的设置——建立过渡.....	114
5.2.4	创建脚本.....	116
5.2.5	运行效果展示.....	118
5.3	动画的融合——动画层和身体遮罩.....	119
5.4	使用脚本代替根动作.....	124

5.4.1	根动作的应用.....	124
5.4.2	脚本代替根动作做出处理.....	126
5.5	添加道具到人物模型上.....	132
5.6	配合人物模型的动作来投掷对象.....	135
5.7	应用布娃娃物理系统的人物模型.....	139
5.8	旋转人物模型的上半身去瞄准.....	143
第 6 章	声音的应用.....	148
6.1	声音音调配合动画播放速度.....	148
6.2	添加音量控制.....	152
6.3	模拟隧道里的回声效果.....	158
6.4	防止音乐片段在播放的过程中重播.....	161
6.5	音乐播放结束后销毁游戏对象.....	163
6.6	制作可动态改变的背景音乐.....	166
第 7 章	外部资源的应用.....	173
7.1	使用Resources加载外部资源.....	173
7.2	使用Resources文件夹加载外部资源.....	177
7.3	使用网址加载外部资源.....	182
7.4	使用静态属性存储和加载玩家数据.....	183
7.4.1	一个游戏的雏形.....	184
7.4.2	给游戏增加玩家数据存储的功能.....	186
7.5	使用PlayerPrefs存储和加载玩家数据.....	190
7.6	为游戏添加截图功能.....	193
第 8 章	TXT和XML文件的应用.....	197
8.1	使用TextAsset加载外部文本文件.....	197
8.2	使用C#文件流加载外部文本文件——读取数据.....	198
8.3	使用C#文件流加载外部文本文件——写入数据.....	201
8.4	加载并解析外部的XML文件.....	202
8.5	使用XMLTextWriter创建XML文件中的数据.....	204
8.6	使用串行化的方式自动创建XML文件中的数据.....	208
8.7	使用XMLDocument直接创建包含数据的XML文件.....	212
第 9 章	角色移动和状态切换.....	215
9.1	由玩家控制对象的移动.....	215
9.2	控制对象的朝向.....	219
9.3	控制对象与对象间的相对移动.....	222
9.3.1	相对移动——寻找.....	222
9.3.2	相对移动——靠近对象时减速.....	227
9.3.3	相对移动——保持距离.....	229
9.4	控制对象群组的移动.....	231
9.5	控制角色向前投掷物体.....	236
9.6	控制角色在一个随机的点出现.....	242
9.7	控制角色在指定点出现.....	246
9.8	控制角色按照指定路线行进.....	248
9.9	控制游戏不同状态间的切换.....	252
9.10	使用多个类来管理游戏的多个状态.....	255

第 10 章 完善和优化游戏.....	260
10.1 让游戏处于暂停状态.....	260
10.2 让游戏在指定时间内减速运行.....	263
10.3 使用偏振投影实现 3D 立体效果.....	267
10.4 阻止你的游戏在未知的网站上运行.....	272
10.5 优化原则：使用代码分析技术查找游戏性能瓶颈.....	273
10.6 优化原则：减少对象的数量——不需要的时候就销毁.....	277
10.7 优化原则：使用委托和事件提升效率.....	279
10.8 优化原则：使用协同程序有规律的执行逻辑代码.....	281
10.9 优化原则：将计算量大的任务分到多个帧执行.....	282
10.10 优化原则：尽量减少对象和组件的查找.....	284
第 11 章 Unity 收费版提供的功能.....	290
11.1 让摄像机聚焦不同的对象——景深效果.....	290
11.2 为汽车加后视镜.....	294
11.3 使用声音过滤器模拟水中的音效.....	298
11.4 在场景对象上播放视频.....	301
11.5 在 Game 视图上播放外部的视频文件.....	304



## 第 1 章 熟悉 Unity 及其简单操作

如果读者是刚接触到 Unity 的话，本章可以让你更加的熟悉 Unity。而且本章所介绍的一些简单的设置操作，可以让这个游戏引擎在读者看来更舒服一些。

### 1.1 安装 Unity

Unity 的安装包可以从它的官方网站下载到，如图 1-1 所示。

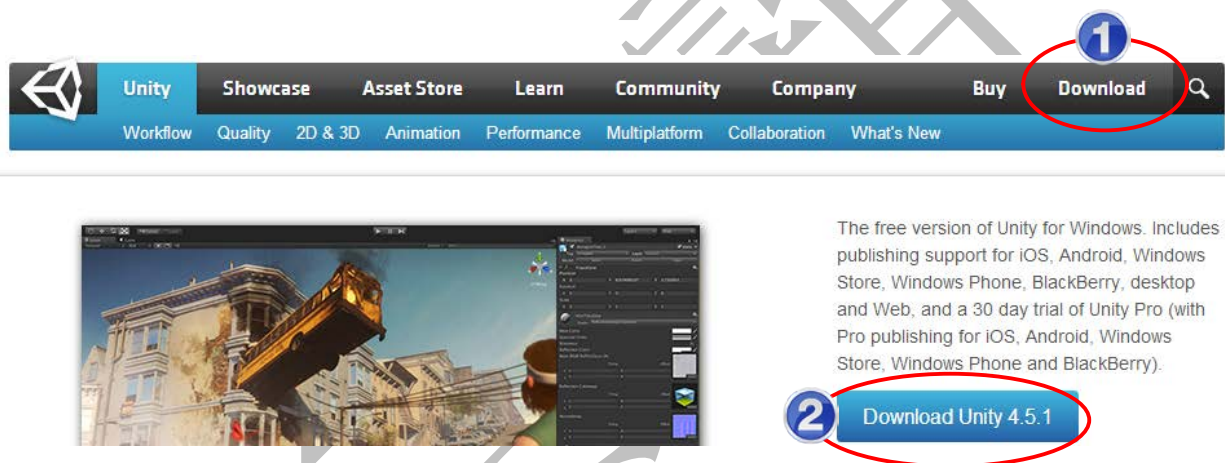


图 1-1 从官网下载 Unity

链接是：<http://unity3d.com/unity/download>。写作本书时，可以下载到的最新版是 Unity 4.5.1，大小是 1.09G，下载到的安装包如图 1-2 所示。

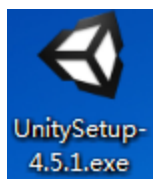


图 1-2 下载到的 Unity 安装包

双击此安装包，即可开始 Unity 的安装过程，如图 1-3、图 1-4 和图 1-5 所示。

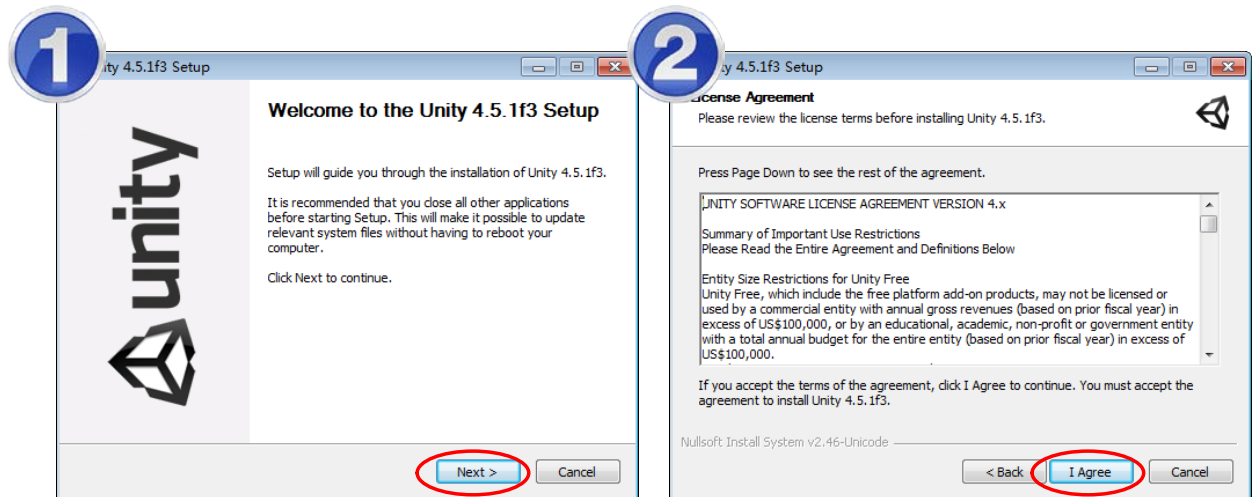


图 1-3 Unity 安装过程 1

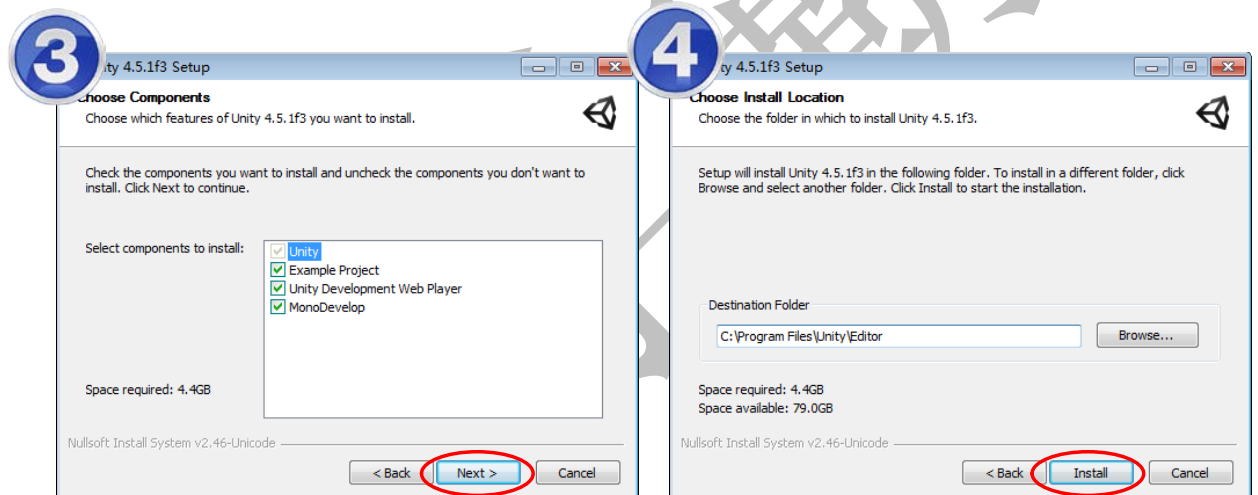


图 1-4 Unity 安装过程 2

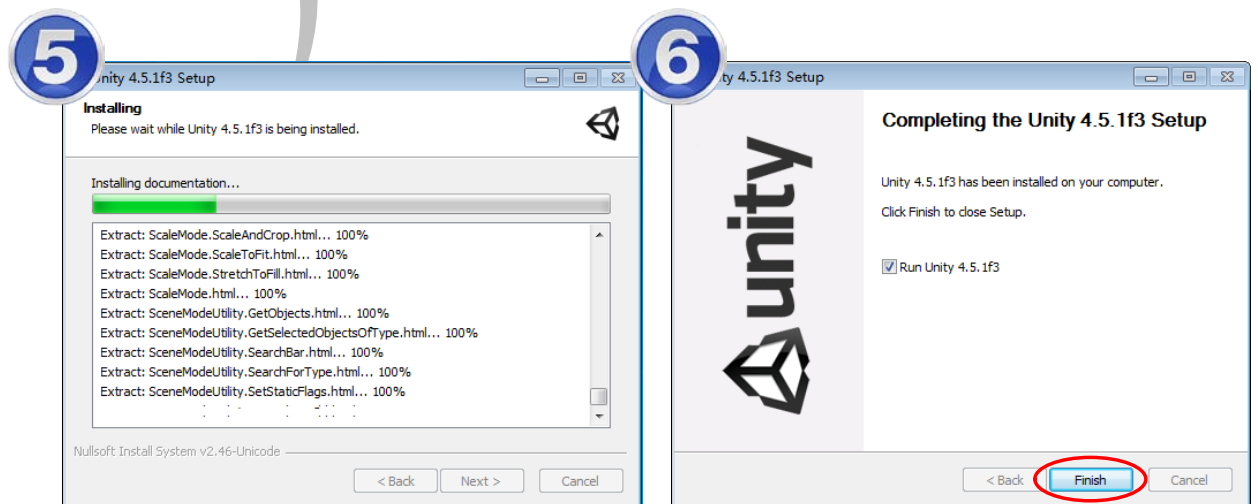


图 1-5 Unity 安装过程 3

整个安装过程十分的简单，除了指定一个软件的安装位置外，其它保留默认设置即可，大约用 15 分钟即可完成。双击安装好的 Unity 软件应用程序图标，选择免费的版本，对于初学者而言，这个版本的功能足够使用，等到以后有需求了再安装收费但是功能更强大的版本也不迟，如图 1-6 所示。

提示：建议初学 Unity 的读者选择免费的版本即可，但是本书为了尽可能全面的介绍 Unity 的所有有趣功能，使用的是 Unity 的收费版，毕竟收费版提供了更多的功能。

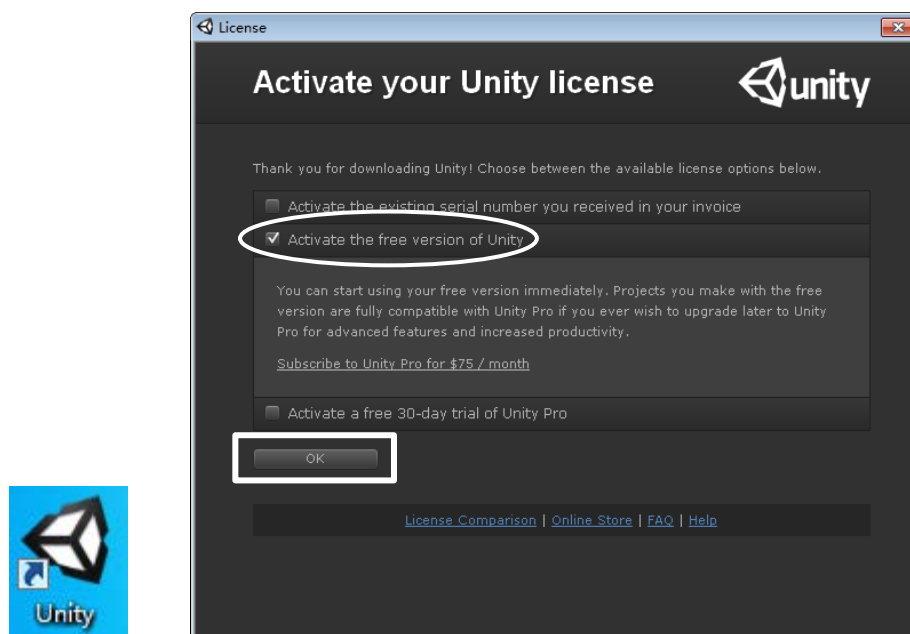


图 1-6 双击软件应用图标，选择免费版本

接下来，读者可以用不到 1 分钟的时间，使用邮箱注册一个帐号，并且在登录这个帐号以后，就可开始使用 Unity 这个软件了，如图 1-7 所示。

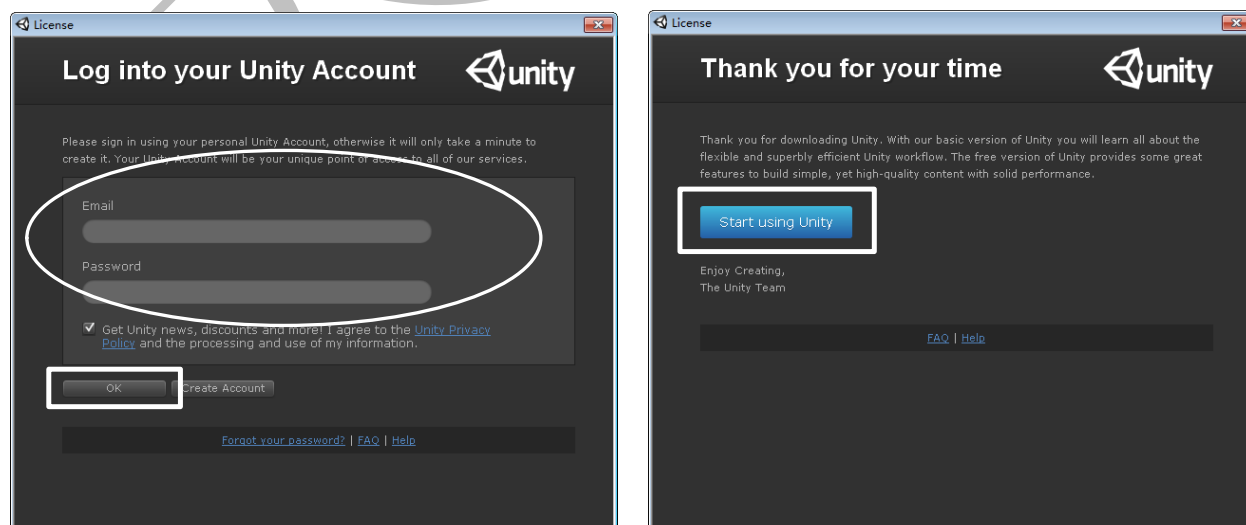


图 1-7 注册并登录帐号，然后就可以开始使用 Unity 了

提示：帐号的用途还是很多的，例如，在登录了帐号以后，可以在官网上下载一些游戏示例来研究。

## 1.2 编辑器的偏好设置

在开始使用 Unity 这个编辑器之前，读者最好了解一下，如何按照自己的意愿和喜好，设置它的一些参数。这样将不仅使得编辑器更适合读者自己使用，从而加快你的开发速度，还能让你的开发过程变的更加愉快一些。要调整 Unity 的参数，可以按照下面的步骤来操作：

（1）在编辑器内，单击 **Edit|Preferences...** 命令，然后会出现名为 **Unity Preferences**（Unity 偏好设置）的对话框，如图 1-8 所示。对话框包含 5 个标签，在每个标签下都有一些可以设置的参数。

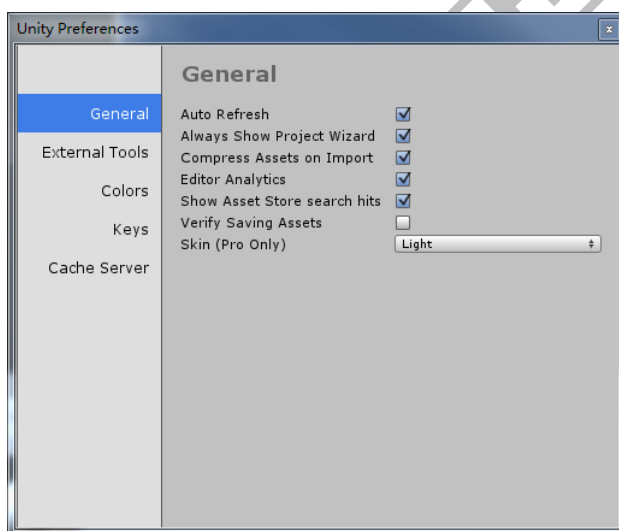


图 1-8 单击命令，弹出用于设置参数的对话框

（2）双击运行 Unity 的时候，它总是默认打开，关闭 Unity 时正在编辑的游戏项目。如果你想打开的是另一个游戏项目，就不得不在编辑器内，单击 **File|Open Project...** 命令，然后再打开指定的游戏项目。这样显得麻烦。在 **General**（综合设置）标签下，选中 **Always Show Project Wizard**（是否总是开启项目向导）复选框，如图 1-9 所示，运行 Unity 的时候，它会先要求你指定要打开的游戏项目，然后再显示 Unity 的编辑器界面。

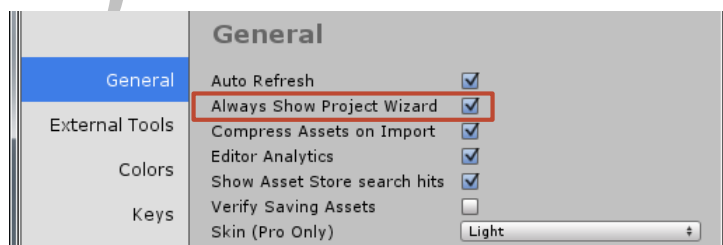


图 1-9 General 标签下，偏好的设置

（3）在 Unity 中编写脚本时，默认使用自带的 **MonoDevelop**。如果你想用其它的软件编写脚本，可以在 **External Tools**（外部工具）标签下，设置 **External Script Editor**（外部脚本编辑器）属性，从它的下列列表中选择喜欢的脚本编辑器。同理，Unity 默认按照图片文件的扩展名，指定外部对应的程序打开文件，如果想指定一款软件打开所有类型的图片，可

以设置 Image application（图像应用）属性，如图 1-10 所示。

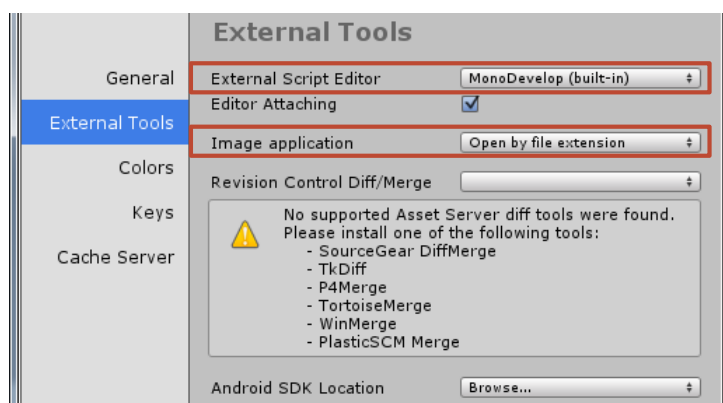


图 1-10 External Tools 标签下，偏好的设置

（4）在 Colors 标签下，可以设置 Unity 编辑器界面各组成部分的颜色；在 keys 标签下，可以设置许多操作的快捷键；读者可以依据需要设置，如图 1-11 所示。

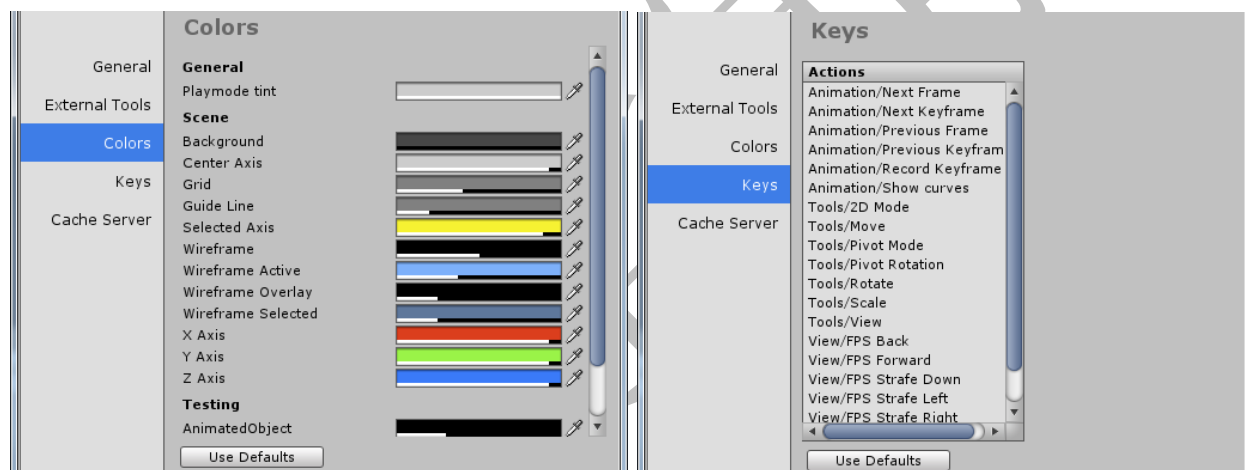


图 1-11 Colors 和 Keys 标签下的各设置项

## 1.3 熟悉 Unity 的编辑器界面

虽然 Unity 的编辑器界面被设计的很友好，但是你仍然可能对这些陌生对话框感到恐惧。本节就带你熟悉一下 Unity 的界面。首次打开 Unity 编辑器时，默认的界面如图 1-12 所示。

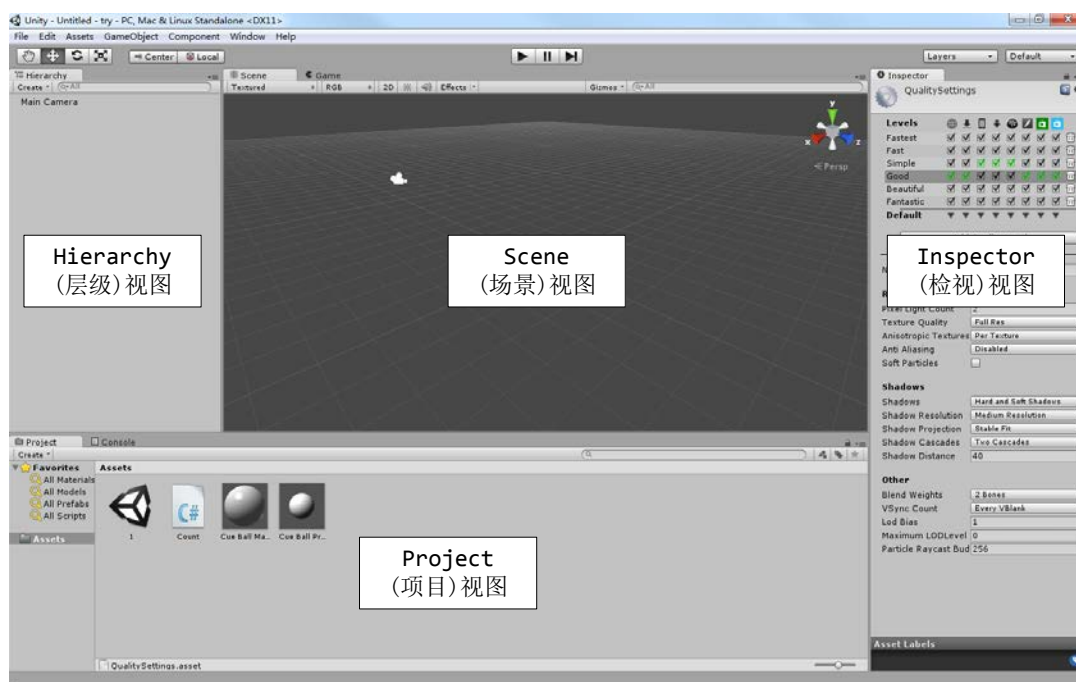
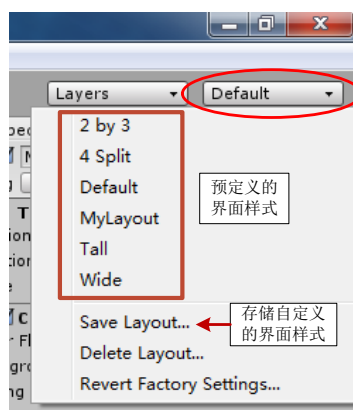


图 1-12 Unity 默认使用的编辑器界面

除了熟悉的菜单和工具栏以外，还有包括了 4 个视图：

- ❑ Scene（场景）视图：在此视图中，可以使用鼠标手动修改场景中游戏对象的位置、朝向和大小。
- ❑ Hierarchy（层级）视图：当前场景中所有游戏对象的名称，都被列在此视图中。
- ❑ Project（项目）视图：这是创建、访问游戏资源的地方。无论网格、纹理，还是预置体，都被列在这个视图里。
- ❑ Inspector（检视）视图：在其它 3 个视图中选中的任何游戏对象、资源，都可以在此视图中被配置。因为此视图中列出了所选对象的所有属性。

除了默认的界面样式，Unity 还预定义了其它的样式，单击工具栏最右边的按钮即可查看和修改，如图 1-13 所示。各窗口可以自由拖动到任何地方，甚至是单独作为一个窗口，如图 1-14 所示，然后你可以定义并命名自己的界面样式。



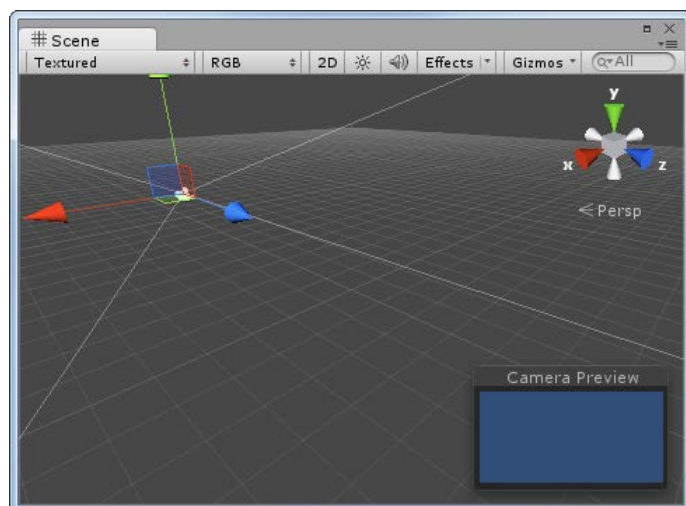


图 1-13 改变 Unity 的界面样式

图 1-14 成为了单独窗口场景视图

在默认的 Unity 界面中, 还有两个视图被“隐藏”了, 它们是 Game (游戏) 视图和 Console (控制台) 视图, 如图 1-15 所示。单击相应的标签即可显示相应的视图。



图 1-15 默认界面中“隐藏”的 Game 视图和 Console 视图

Game 视图在 Unity 运行游戏的时候, 会自动出现, 因为 Game 视图所展示的就是游戏实际运行时所看到的效果。视图上边框从左到右有 4 个按钮, 如图 1-16 所示。



图 1-16 Game 视图上边框的 4 个按钮

各按钮的作用是:

- ❑ Free Aspect, 可以随意调节 Game 视图的分辨率, 如图 1-17 所示, 但是要调节实际运行时, 游戏的分辨率不能用此方法。

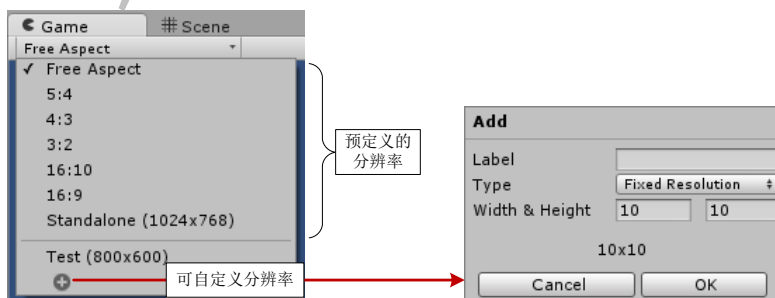


图 1-17 调节 Game 视图的分辨率

- ❑ Maximize on Play, 运行游戏的时候, Game 视图的窗口将最大化显示, 而视图将以



Free Aspect 中设置的分辨率显示。

- ❑ Stats，单击此按钮，会弹出 Statistics 面板，里面会显示运行场景的渲染速度、Draw Call 的数量、帧率、贴图占用的内存等参数，如图 1-18 所示。

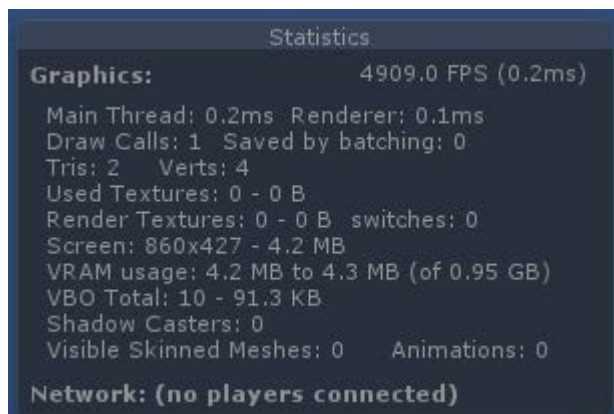


图 1-18 Statistics 面板，及其各个参数

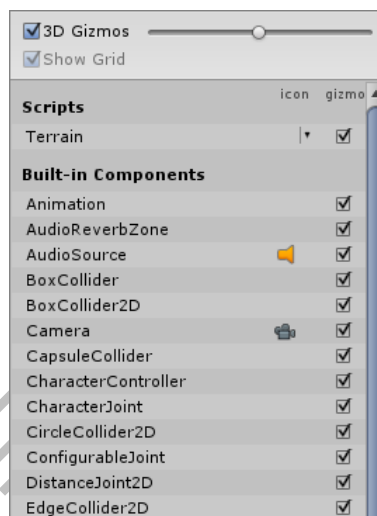


图 1-19 Gizmos 中的部分选项

- ❑ Gizmos，可以设置是否显示场景中的灯光、声音、相机等游戏对象的图标，如图 1-19 所示。

Console 视图，在开发游戏的过程中，用于显示一些调试信息，包括错误（error）、警告（warning），及其一些输出信息，如图 1-20 所示。

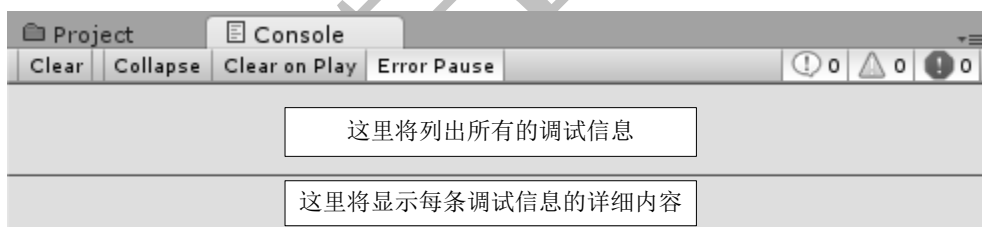


图 1-20 Console 视图

## 1.4 将 Unity 中的资源保存到预设体中

本节将会把游戏对象保存到 Project 视图的预设体（prefab）中。目的是使得游戏对象能被重复利用。先创建一个游戏对象，然后对此游戏对象做些简单的设置：

（1）在 Unity 中，单击 GameObject>Create Other|Sphere 命令，在游戏场景中会创建一个球体。它的默认名称 Sphere 会被列在 Hierarchy 视图中。鼠标右击此名称，在弹出的快捷菜单中选择 Rename 命令，更改球体的名称为 Ball，如图 1-21 所示。



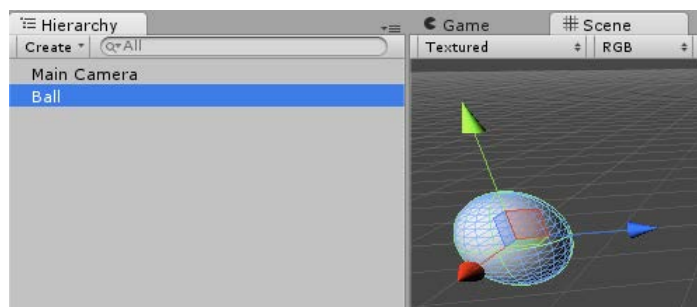


图 1-21 Hierarchy 视图中的球体名与 Scene 视图中的球体

(2) 在 Project 视图里，单击 Create 按钮，选择 Material 选项，可在 Project 视图中创建名为 New Material 的材质（Material），将其命名为 Ball Material，如图 1-22 所示。

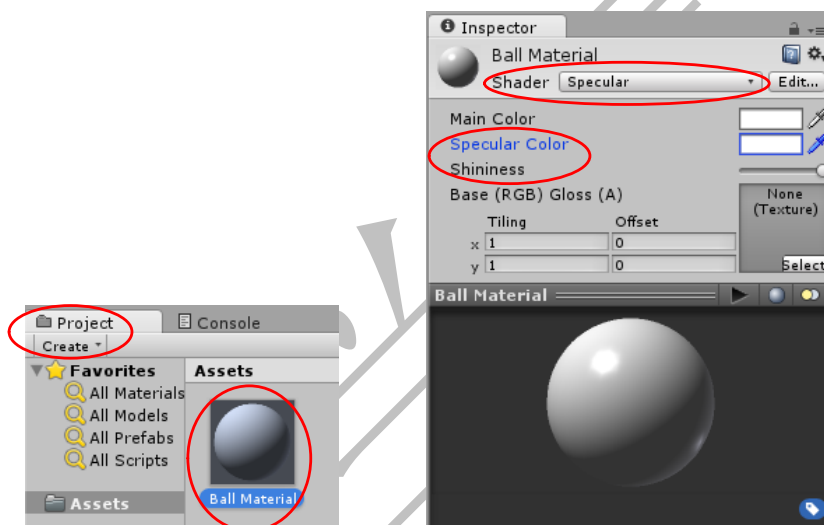


图 1-22 在 Project 视图里创建材质

图 1-23 在 Inspector 视图中，修改材质属性

(3) 选中在 Project 视图里新创建的材质，然后在 Inspector 视图里改变此材质的属性：设置 Shader（着色器）为 Specular（镜面反射）；Specular Color（镜面反射的颜色）为白色；Shininess（光泽）调到最大，如图 1-23 所示。

(4) 将 Project 视图中的 Ball Material，用鼠标拖动到 Scene 视图中的 Ball 上。

(5) 选中 Ball 后，单击 Component|Physics|Rigidbody 命令，为 Ball 添加 Rigidbody（刚体）组件，如图 1-24 所示。

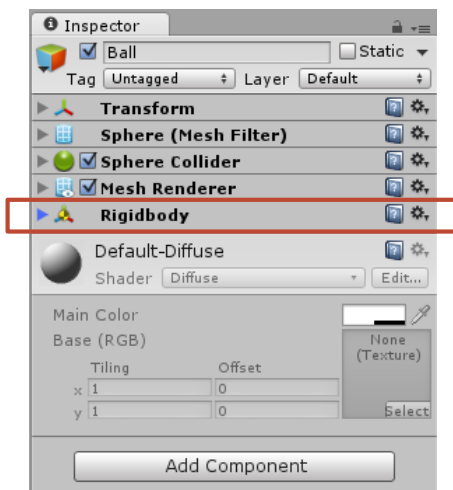


图 1-24 为 Ball 添加 Rigidbody 组件

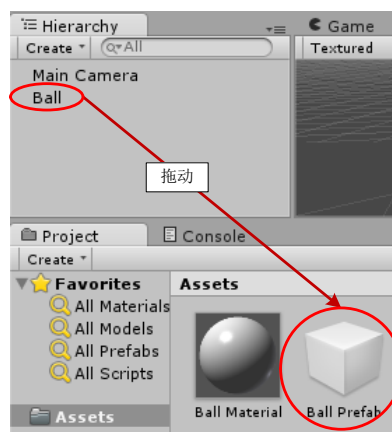


图 1-25 拖动的过程

游戏对象已经创建好了，接着需要创建预设体，最后把游戏对象保存到 Project 视图的预设体（prefab）中即可，具体操作是：

- (1) 在 Project 视图中，单击 Create 按钮，选择 Prefab 选项，则会创建一个名为 New Prefab 的预设体，修改它的名称为 Ball Prefab。
- (2) 拖动 Hierarchy 视图里的 Ball，到 Project 视图里的 Ball Prefab，如图 1-25 所示。

## 1.5 使用 Unity 内置的资源包

Unity 内置了资源包，设计的目的是用于满足一些普遍的需求。因此使用 Unity 不需要你创建所有的东西。那么 Unity 提供了哪些类型的资源包呢？在 Unity 里，单击 Assets|Import Package 命令，即可查看，如图 1-26 所示。

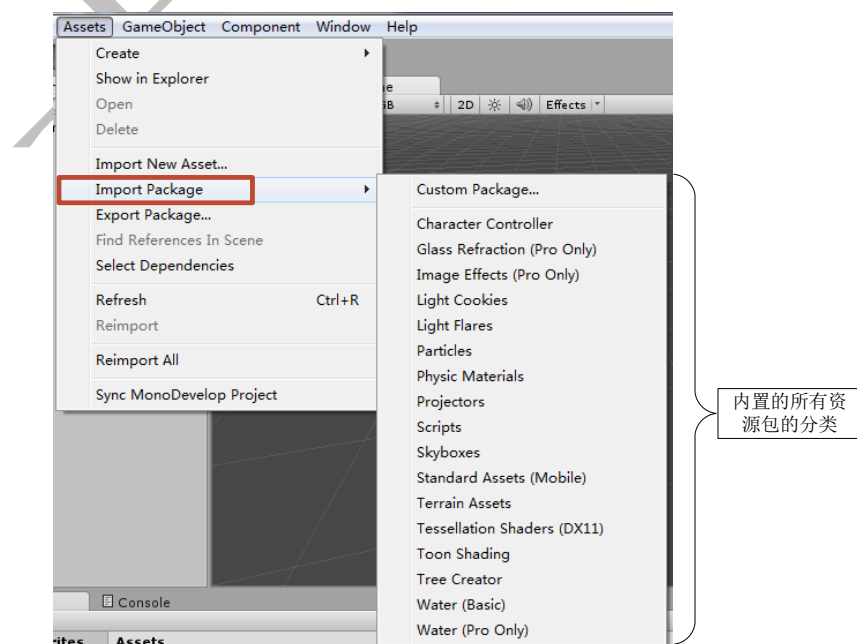


图 1-26 Unity 内置的各类资源包

只要把这些内置的资源包导入到当前的游戏项目中，就可以任意使用包中的资源了。例如，选择 **Water(Basic)**，会弹出相应的对话框要求选择导入这个包中的那些资源，默认是导入此包中的所有资源。单击 **Import** 按钮后，**Water(Basic)** 就会被导入且显示在 **Project** 视图里，如图 1-27 所示。

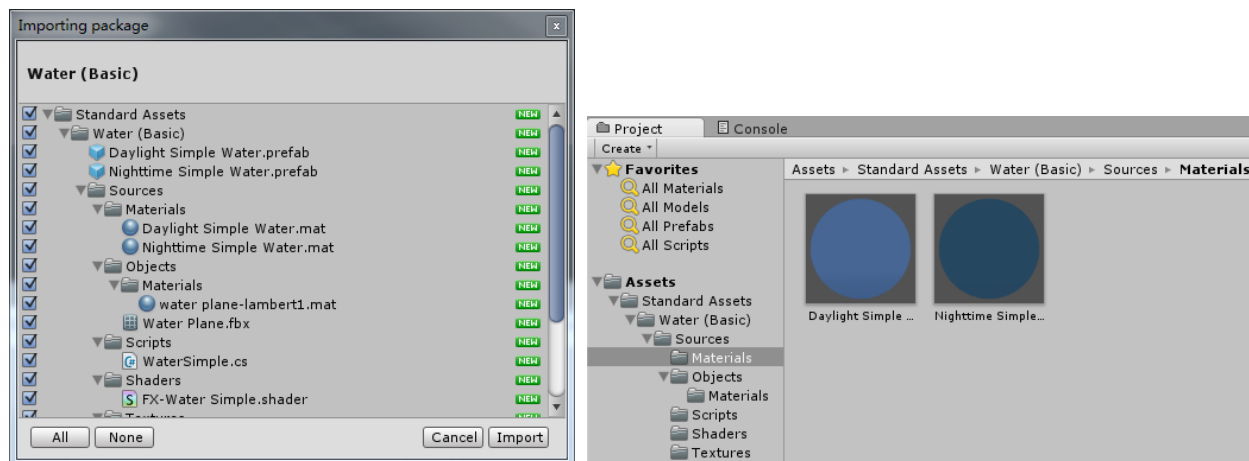


图 1-27 导入的 Water(Basic)

## 1.6 导入自己的资源

在开发游戏的时候，一定也准备了不少使用其它软件制作的各类资源。Unity 允许你将这些资源导入到当前的游戏项目中。在 Unity 里，单击 **Assets|Import New Asset...** 命令，在弹出的 **Import New Asset** 对话框中，找到要导入的资源，然后单击 **Import** 即可。导入的资源会被列在 **Project** 视图中。如图 1-28 所示。

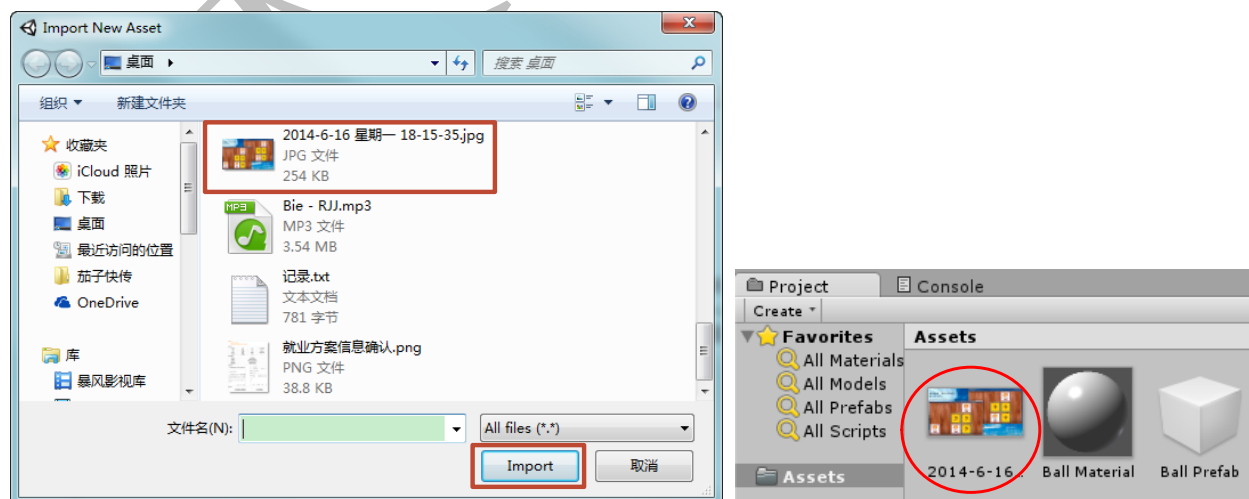


图 1-28 导入自己的资源到游戏项目

所有被导入到项目中的资源，都被存储到项目路径中，名为 **Assets** 的文件夹里。并且

Assets 文件夹中存放的是相应资源的拷贝，而不是原始文件。如图 1-29 所示。

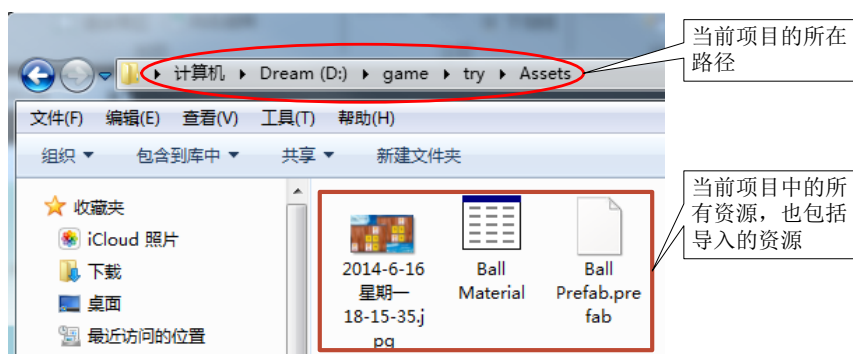


图 1-29 项目路径中 Assets 文件夹里的文件

提示：尽管所有的资源文件，都会被存放在 Assets 文件夹下，但是请不要使用 Windows 的资源管理器随意的为这个文件夹中添加文件，甚至是修改名字，这样的信息修改不会在项目中得到同步，况且还可能引入其它的问题。所以建议读者，只在 Unity 编辑器内编辑各项资源。

## 1.7 导出 Unity 中的资源

当前项目中的资源，以后可能会被用在其它的游戏项目中，因此若是可以把当前项目中的指定资源，打包并导出将最好不过。实际上 Unity 提供了这样的机制，导出的包也可以被 Unity 识别。具体的操作是：

- (1) 在 Unity 里，单击 Assets|Export Package...命令。
- (2) 在弹出的 Exporting package 对话框中选择要导出的资源，然后单击 Export...按钮。
- (3) 在为导出的资源包指定名称和路径以后，单击“保存”按钮即可，如图 1-30 所示。

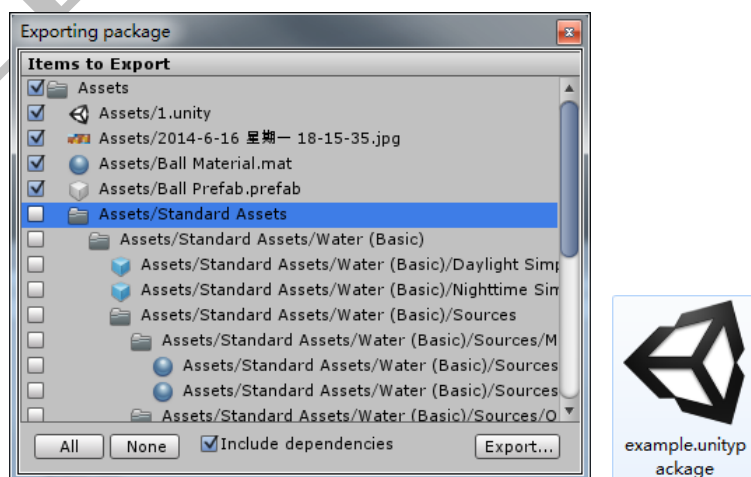


图 1-30 导出项目中的指定资源

## 1.8 导入自己的资源包

除了 Unity 内置的资源包以外，还有许多资源包是被我们自己导出的。本节来学习如何将我们自己导出的资源包导入到当前的游戏项目中。具体操作是：

- （1）在 Unity 里，单击 Assets|Import Package|Custom Package...命令。
- （2）在弹出的 Import package 对话框中找到要导入的资源包，然后单击“打开”按钮，在接下来出现的对话框中列出了此资源包中的所有资源，可以有选择的导入资源包中的部分资源，最后单击 Import 按钮即可，如图 1-31 所示。

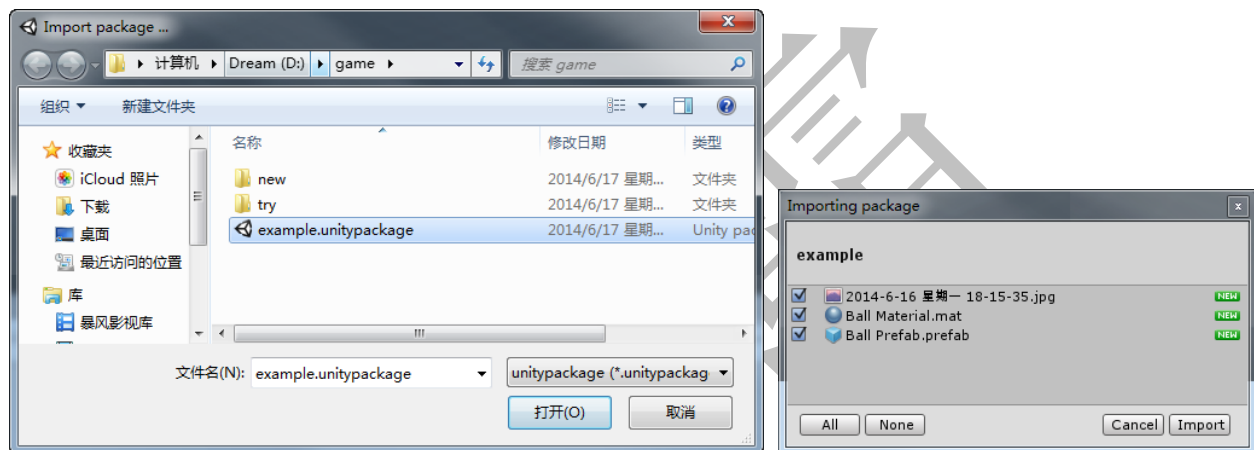


图 1-31 选择要导入的资源包，以及包中的哪些资源

## 1.9 添加资源包到资源包列表中

如果一个自定义的资源包，总是被频繁的加入到项目中，就可以考虑把这个资源包，添加到菜单项中的 Unity 资源包列表中。具体操作是：

- （1）复制资源包文件，然后粘贴到 Unity 安装目录里的 Editor\Standard Packages 文件夹下。
- （2）重启 Unity 后，单击 Assets|Import Package 命令，在子菜单中就可以看到自定义的资源包，如图 1-32 所示。

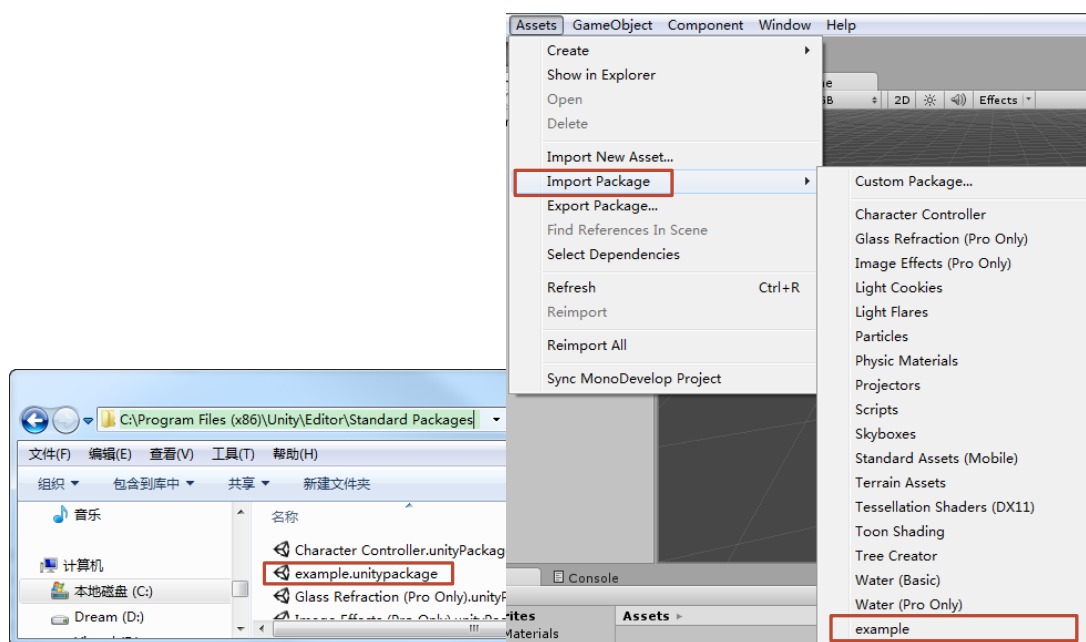


图 1-32 添加自定义的资源包到 Unity 资源包列表中

导入的资源同样会被列在 Project 视图中。

## 1.10 使用 Project 视图检索器

当前项目中有很多资源时，为了快速找到想使用的资源，可以使用 Project 视图检索器（browser）检索资源。Project 视图检索器如图 1-33 所示。

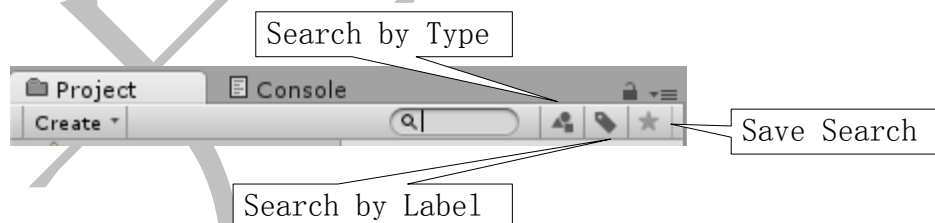


图 1-33 Project 视图检索器

（1）在检索器的文本框中输入 Palm 时，得到的检索结果如图 1-34 所示。

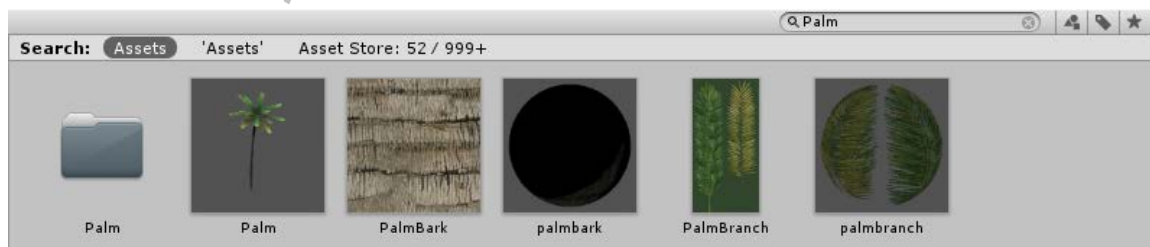


图 1-34 检索名称以 Palm 开头的资源

（2）除了名称以外，还可以设置检索资源的类型或者标签。单击对应的按钮，从弹出的列表中选中对应的项即可。如果想选择多项，可以在按下 Ctrl 的同时选择菜单项，如图

1-35 所示。

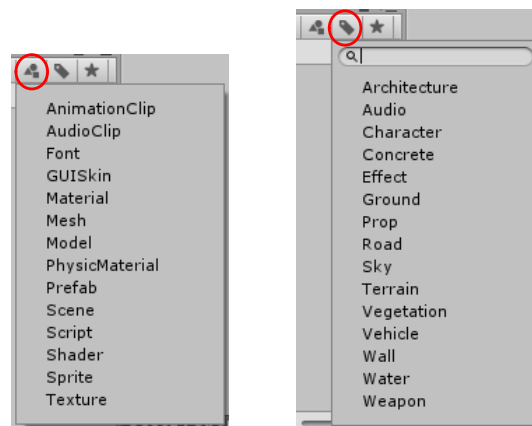


图 1-35 设置检索资源时的类型和标签

(3).检索器上最后一个按钮,可以保存当前的搜索结果。在 Project 视图左侧的 Favorites 列表中,如图 1-36 所示,还可以为此搜索结果随意命名。

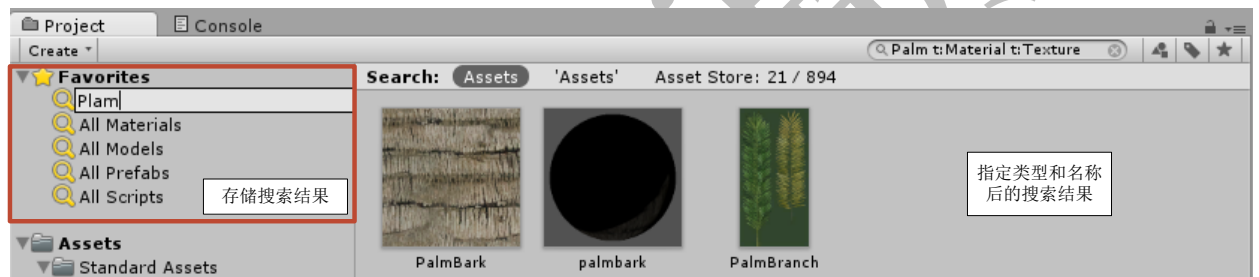


图 1-36 保存当前的搜索结果

