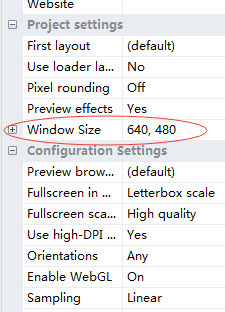
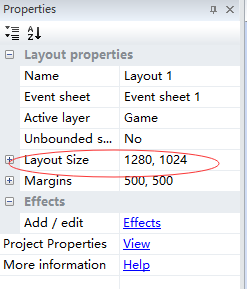
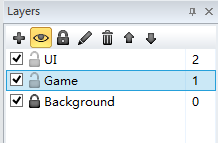
Top-down shoot实验指导书

首先还是新建一个空白的project，将layout size改为1280, 1024 ，窗口大小window size 改为640, 480



此次project有三个图层，分别是background（背景）、game（游戏）和UI（界面层）

与背景相关的素材放在背景层，与游戏内容相关的素材如玩家、敌人之类的让游戏层，辅助性的素材如按钮，提示语之类的放界面层。



接下来，自定义添加游戏背景，同样使用TiledBackground

然后添加player（玩家）和enemy（敌人）还有bullet（子弹）（如果自己有素材，可以直接拖入编辑区，GIF动画也使用）。

在本次的游戏中，通过点击鼠标能使player发射子弹，按键盘WSAD能上下左右移动。

因此，我们的输入包含了mouse（鼠标）和keyboard（键盘），点击编辑区空白处，添加和

OK！素材都添加完毕了，那接下来就开始添加事件吧！  
首先，分析一下我们需要的事件：

1.每个时刻，player都要朝向鼠标

2.；当鼠标左键按下的时候，player位置处会生成一个子弹。

3. 按住W键，player上移；按住S键，player下移；按住A键，player左移；按住D键，player右移

4.当子弹射中敌人时，即子弹和敌人碰撞时，子弹消失，敌人的血量减10.

5.当敌人的血量<=0时，敌人消失。

添加第一个事件

每个时刻，玩家朝向鼠标



与之前朝向鼠标的事件有一点不同的是，在位置坐标中添加了“Game”，此处（Mouse.X(“Game”) , Mouse.Y(“Game”)）Game表示在Game图层，因为我们的player在游戏曾，因此事件在Game执行，添加(”Game”)。

添加第二个事件

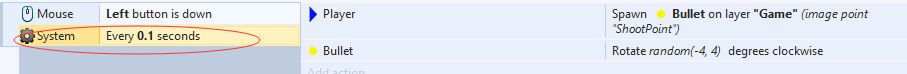
鼠标左键按下，玩家位置处生成一颗子弹，子弹在（-4,4）角度中随机顺时针旋转



下面这条事件表示：bullet（子弹）在（-4,4）的角度之间随机选择一个度数做顺时针旋转，但是我们这里的子弹是圆的，所以这条事件可以删掉，不会影响。

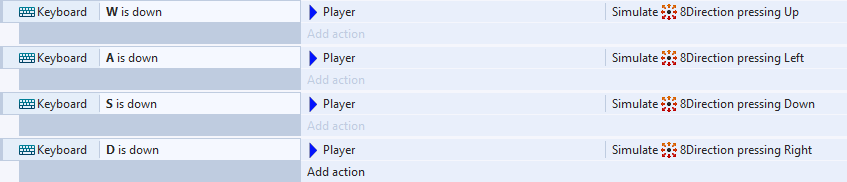


如果这样写条件，我们会发现一个问题，子弹会一连串地射出，效果不好。解决方法在之前的课堂上说过，我们需要添加一个条件：每隔0.1秒，子弹射出，使子弹之间产生时间间距，从而决定子弹连在一起射出的问题



添加第三个事件

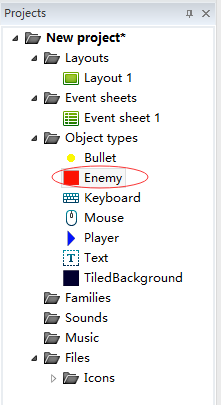
第三条事件分为四部分，大家应该对这四条事件已经熟悉了，不做具体解释



添加第四个事件

在添加事件之前，我们需要为敌人添加一个名为“血量”的标签

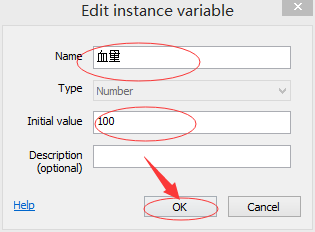
在右侧的工程列表栏中找到object types，点击其中的enemy（敌人），此时编辑区中所有的敌人都会被选中



在属性栏中，找到instance variable，添加标签

F:\电脑\1918455819\FileRecv\MobileFile\Image\PTD`Q9}SH30O4H6OFHCHYTJ.png

将标签命名为“血量”，类型选择number（数字），initial value(初始值)设为100

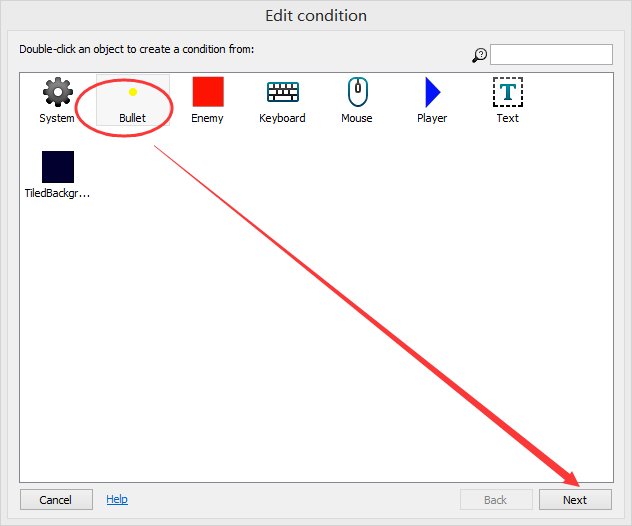


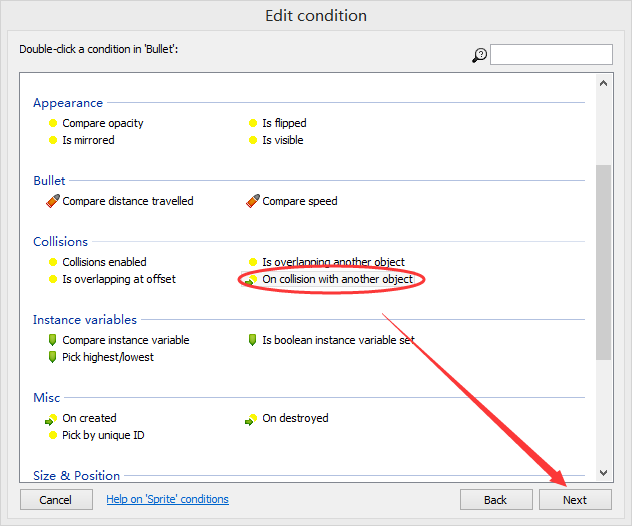
子弹和敌人发生碰撞，敌人血量减10，子弹消失

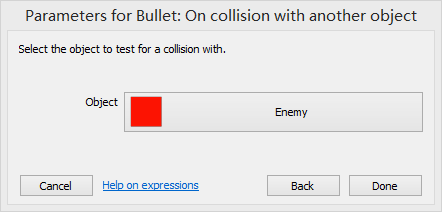


血量减少添加操作

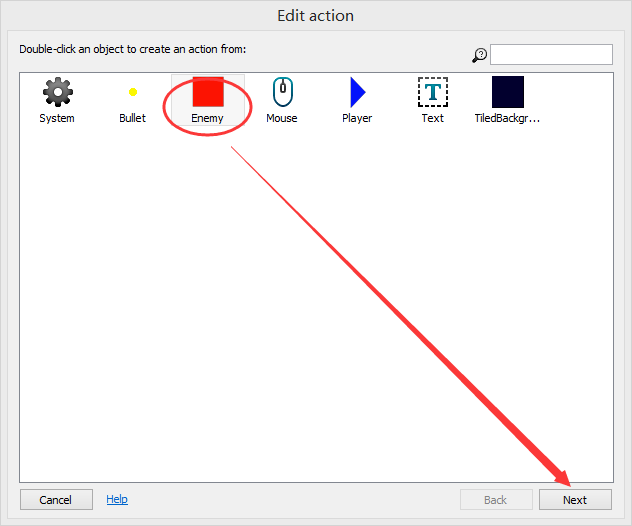
条件添加：

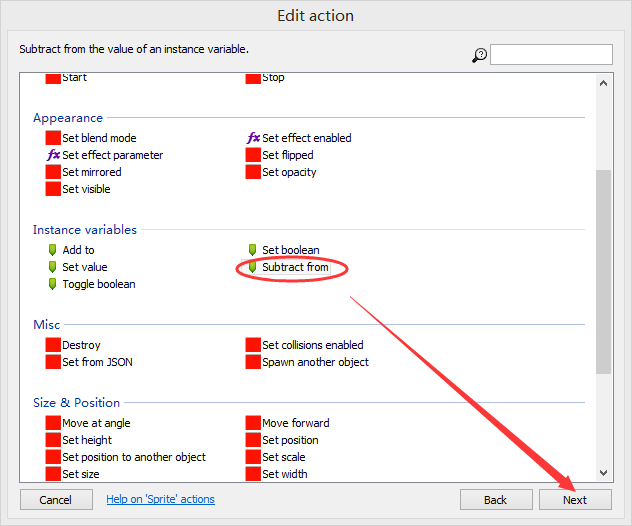




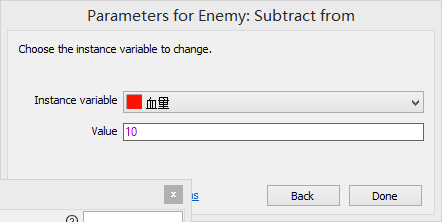


（action）动作添加





Subtract（减少）

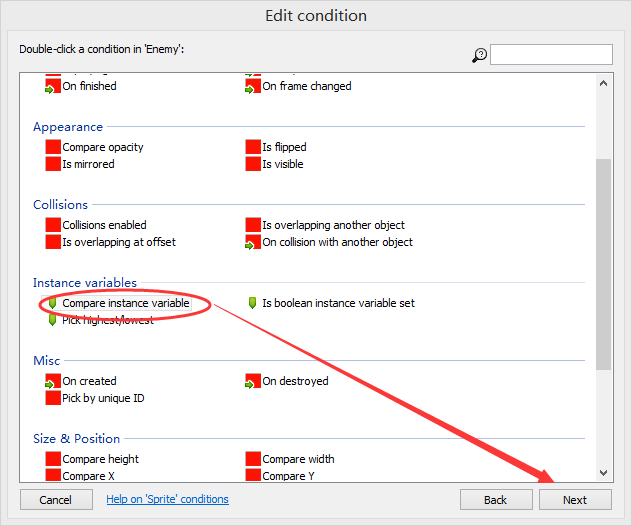


添加第五个事件

敌人血量<=0，敌人消失



条件添加具体操作



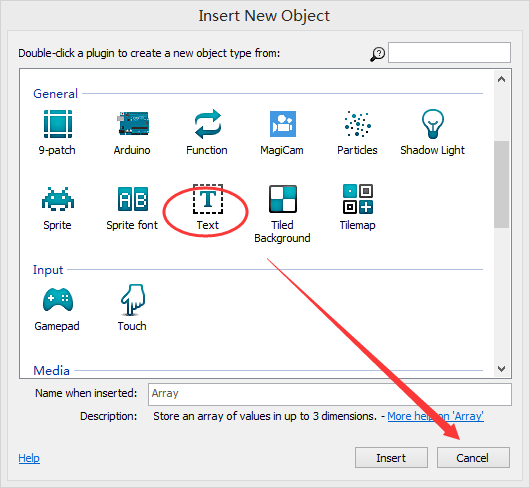
![F:\电脑\1918455819\FileRecv\MobileFile\Image\7]H~EH2~(9OCQT_70V](SBN.png](data:image/png;base64,)

OK！我们的游戏就完成了！

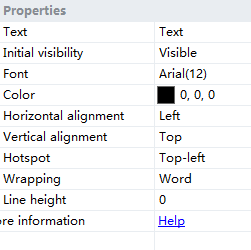
当然，大家还可以将之前所学的也应用进去，比如给敌人添加一些sin的behavior，或者8-direction，让敌人也能够自主地运动起来

分数计数器添加

点击空白处，在UI层添加文本（text），放在左上角的位置（如果不是添加到UI层，比如Game层，会出现在窗口移动，计数器消失的情况，这是因为在计数器放在Game层的左上角，而窗口运动时，窗口会到game层的其他位置，计数器是静止不动的，所以会消失，而UI层是与窗口绑定的，窗口移动，UI层也做相同的移动）



参数设置



Text：文本（设置文本的内容）

Initial visiblity ：初始可见

Font：字体（可以设置字体、字形、字号）

Color：颜色

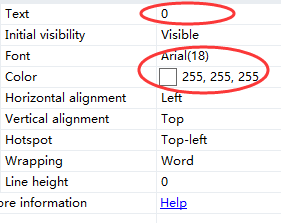
Horizontal alignment：水平方向上的对齐（左对齐、居中对齐、右对齐）

Vertical alignment：垂直方向上的对齐（顶对齐、居中对齐、底对齐）

此处，我们的分数初始值为0

思路：为文本设置一个名为“分数”的标签，子弹射到敌人，即子弹和敌人发生碰撞时，“分数”标签加10，然后将“分数”标签的值赋给文本

具体操作：

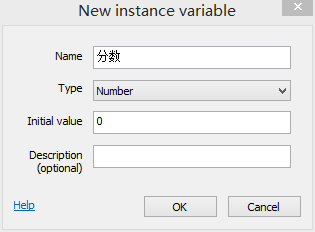


字体变大后，会出现字不见的情况。

解决方法：在编辑区把文本框拉大即可

添加“分数”标签

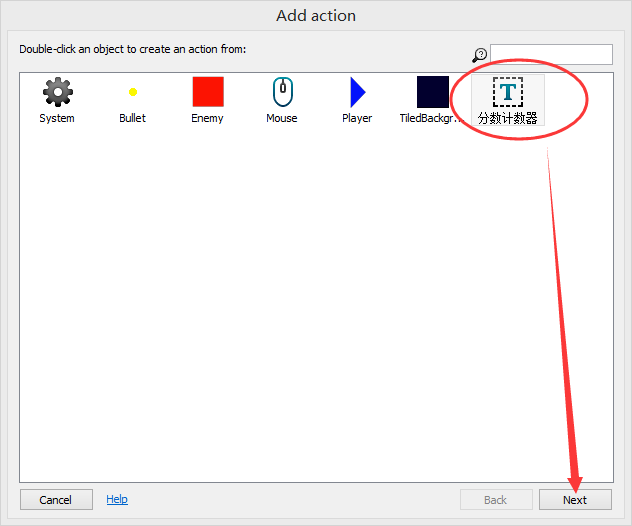
F:\电脑\1918455819\FileRecv\MobileFile\Image\%}F`K0UN$E5R]OJH`2L9BQX.png

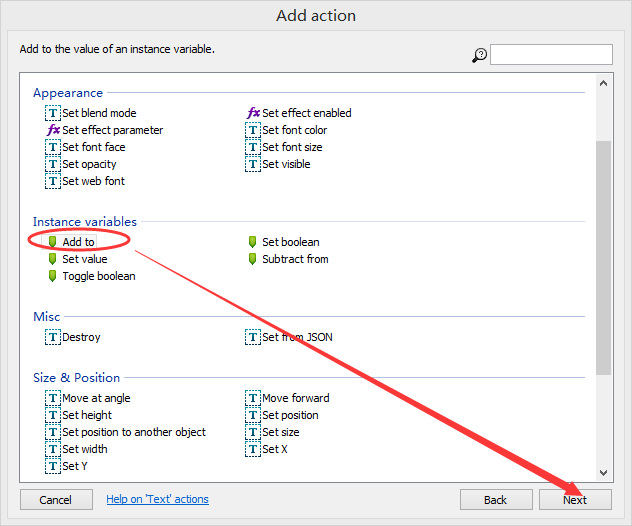


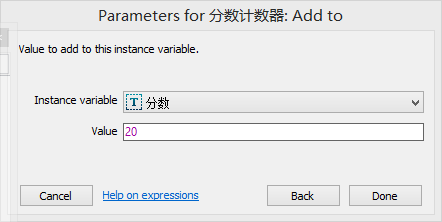
事件添加：

当子弹和敌人发生碰撞，计数器分数增加20



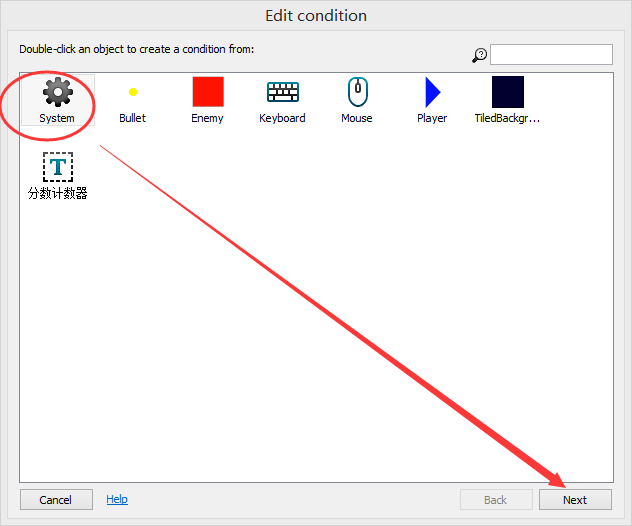


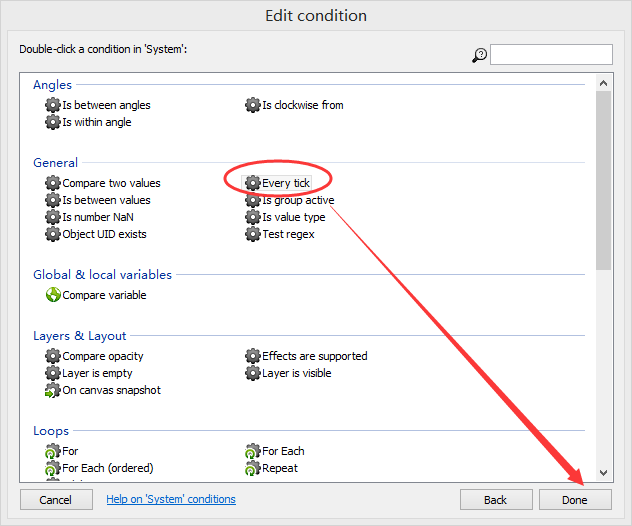


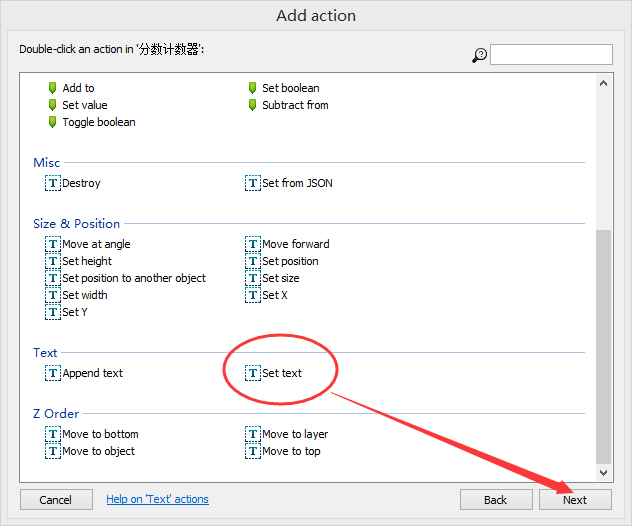


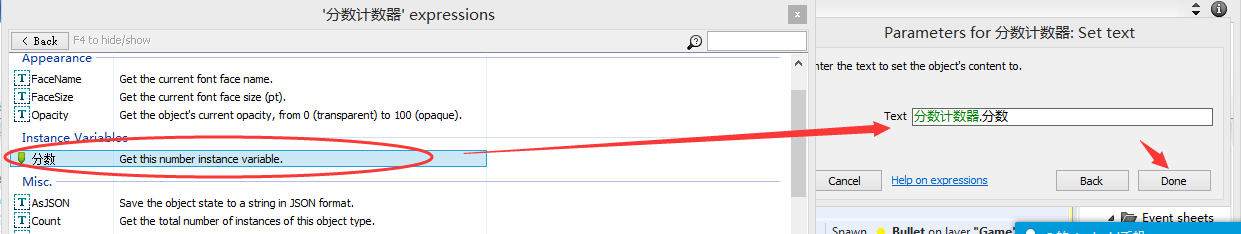
每个时刻，将“分数”标签的值赋给文本

F:\电脑\1918455819\FileRecv\MobileFile\Image\VIDHXVA3`14~ZGJ)~$NFTOK.png









这样，计数器就添加完成了~