1.菜单继承自layer

|  |
| --- |
|  |

练习1.创建一个开关按钮MenuItmeToggle

|  |  |
| --- | --- |
| // MyGameScene.cpp  #include "MyGameScene.h"  USING\_NS\_CC;  using namespace ui;  cocos2d::Scene \* MyGameScene::createScene()  {  return MyGameScene::create();  }  // Print useful error message instead of segfaulting when files are not there.  static void problemLoading(const char\* filename)  {  printf("Error while loading: %s\n", filename);  printf("Depending on how you compiled you might have to add 'Resources/' in front of filenames in HelloWorldScene.cpp\n");  }  bool MyGameScene::init()  {  if (!Scene::init())  return false;  //获取显示尺寸  auto visibleSize = Director::getInstance()->getVisibleSize();  //获取显示原点  auto origin = Director::getInstance()->getVisibleOrigin();  //创建关闭按钮  //auto closeBtn = MenuItemImage::create("closeNormal.png", "closeSelected.png", CC\_CALLBACK\_1(MyGameScene::menuCloseCallback, this));  ////判断是否创建成功  //if (closeBtn == nullptr || closeBtn->getContentSize().width < 0 || closeBtn->getContentSize().height < 0)  //{  // problemLoading("'closeNormal.png' and 'closeSelected.png'");  //}  //else  //{  // //设置按钮显示位置  // float x = origin.x + visibleSize.width - closeBtn->getContentSize().width / 2;  // float y = origin.y + closeBtn->getContentSize().height / 2;  // closeBtn->setPosition(x, y);  //}  auto spriteNormal = Sprite::create("closeNormal.png");  auto spriteSelected = Sprite::create("closeSelected.png");  auto selectedItem = MenuItemSprite::create(spriteSelected, spriteSelected);  auto normalItem = MenuItemSprite::create(spriteNormal, spriteNormal);    //使用toggle菜单项目    auto toggleBtn = MenuItemToggle::createWithCallback(CC\_CALLBACK\_1(MyGameScene::menuCloseCallback, this), selectedItem,normalItem,NULL);  auto menu = Menu::create(toggleBtn, NULL);  float x = origin.x + visibleSize.width - toggleBtn->getContentSize().width / 2;  float y = origin.y + toggleBtn->getContentSize().height / 2;  toggleBtn->setPosition(x, y);  menu->setPosition(Vec2::ZERO);  this->addChild(menu, 1);    ////定义一个位置基准点  //Vec2 base = Vec2(visibleSize.width / 2, 0);  ////创建→按钮  //auto rightBtn = createDirButton(0, base);  //this->addChild(rightBtn);  //float width = rightBtn->getContentSize().width;  //float height = rightBtn->getContentSize().height;  //base += Vec2(0, height);  //rightBtn->setPosition(base + Vec2(width, 0));  ////创建↓  //auto downBtn = createDirButton(90, base);  //this->addChild(downBtn);  ////创建←按钮  //auto leftBtn = createDirButton(180, base+Vec2(-width,0));  //this->addChild(leftBtn);  ////创建↑按钮  //auto upBtn = createDirButton(270, base+Vec2(0,height));  //this->addChild(upBtn);  return true;  }  void MyGameScene::menuCloseCallback(cocos2d::Ref \* pSender)  {    Director::getInstance()->end();  }  Button \* MyGameScene::createDirButton(float rotation, Vec2 position)  {  Button \* btn = Button::create("right1.png", "right2.png", "right3.png");  //设置按钮文本字体大小  btn->setTitleFontSize(12);  //设置按钮文本  btn->setTitleText("Click me");  btn->setTitleColor(Color3B::MAGENTA);  btn->setColor(Color3B::ORANGE);  btn->addTouchEventListener([&](Ref\* sender,Widget::TouchEventType type) {  switch (type)  {  case cocos2d::ui::Widget::TouchEventType::BEGAN:  break;  case cocos2d::ui::Widget::TouchEventType::MOVED:  break;  case cocos2d::ui::Widget::TouchEventType::ENDED:  CCLOG("you just clicked me...");  break;  case cocos2d::ui::Widget::TouchEventType::CANCELED:  break;  default:  break;  }  });  btn->setRotation(rotation);  btn->setPosition(position);  return btn;  } | //MyGameScene.h  #pragma once  #include"cocos2d.h"  #include"ui/CocosGUI.h"  USING\_NS\_CC;  using namespace ui;  class MyGameScene :public cocos2d::Scene {  public:  static cocos2d::Scene\* createScene();  virtual bool init();  // a selector callback  void menuCloseCallback(cocos2d::Ref\* pSender);  // implement the "static create()" method manually  CREATE\_FUNC(MyGameScene);  Button\* createDirButton(float rotation, Vec2 position);  }; |

练习2.实现静音功能

|  |
| --- |
|  |

|  |  |
| --- | --- |
| //MyGameScene.cpp  #include "MyGameScene.h"  USING\_NS\_CC;  using namespace ui;  cocos2d::Scene \* MyGameScene::createScene()  {  return MyGameScene::create();  }  // Print useful error message instead of segfaulting when files are not there.  static void problemLoading(const char\* filename)  {  printf("Error while loading: %s\n", filename);  printf("Depending on how you compiled you might have to add 'Resources/' in front of filenames in HelloWorldScene.cpp\n");  }  bool MyGameScene::init()  {  if (!Scene::init())  return false;  //获取显示尺寸  auto visibleSize = Director::getInstance()->getVisibleSize();  //获取显示原点  auto origin = Director::getInstance()->getVisibleOrigin();  //创建关闭按钮  //auto closeBtn = MenuItemImage::create("closeNormal.png", "closeSelected.png", CC\_CALLBACK\_1(MyGameScene::menuCloseCallback, this));  ////判断是否创建成功  //if (closeBtn == nullptr || closeBtn->getContentSize().width < 0 || closeBtn->getContentSize().height < 0)  //{  // problemLoading("'closeNormal.png' and 'closeSelected.png'");  //}  //else  //{  // //设置按钮显示位置  // float x = origin.x + visibleSize.width - closeBtn->getContentSize().width / 2;  // float y = origin.y + closeBtn->getContentSize().height / 2;  // closeBtn->setPosition(x, y);  //}  ////使用toggle菜单项目  //auto spriteNormal = Sprite::create("closeNormal.png");  //auto spriteSelected = Sprite::create("closeSelected.png");  //auto selectedItem = MenuItemSprite::create(spriteSelected, spriteSelected);  //auto normalItem = MenuItemSprite::create(spriteNormal, spriteNormal);  //  //  //  //auto toggleBtn = MenuItemToggle::createWithCallback(CC\_CALLBACK\_1(MyGameScene::menuCloseCallback, this), selectedItem,normalItem,NULL);  //auto menu = Menu::create(toggleBtn, NULL);  //float x = origin.x + visibleSize.width - toggleBtn->getContentSize().width / 2;  //float y = origin.y + toggleBtn->getContentSize().height / 2;  //toggleBtn->setPosition(x, y);  //menu->setPosition(Vec2::ZERO);  //this->addChild(menu, 1);    ////定义一个位置基准点  //Vec2 base = Vec2(visibleSize.width / 2, 0);  ////创建→按钮  //auto rightBtn = createDirButton(0, base);  //this->addChild(rightBtn);  //float width = rightBtn->getContentSize().width;  //float height = rightBtn->getContentSize().height;  //base += Vec2(0, height);  //rightBtn->setPosition(base + Vec2(width, 0));  ////创建↓  //auto downBtn = createDirButton(90, base);  //this->addChild(downBtn);  ////创建←按钮  //auto leftBtn = createDirButton(180, base+Vec2(-width,0));  //this->addChild(leftBtn);  ////创建↑按钮  //auto upBtn = createDirButton(270, base+Vec2(0,height));  //this->addChild(upBtn);  //实现静音功能  //1.创建精灵  auto unselected = Sprite::create("smmute.png");  auto selected = Sprite::create("smspeaker.png");  //创建精灵菜单项  auto unselectedItem = MenuItemSprite::create(unselected, unselected);  auto selectedItem = MenuItemSprite::create(selected, selected);    //利用两个精灵菜单项来创建一个开关菜单  auto toggleItem = MenuItemToggle::createWithCallback([&](Ref\* ref) {  //实现模拟静音/播放功能  //1.类型转换  auto item = static\_cast<MenuItemToggle\*>(ref);  if (item)  {  if (item->getSelectedIndex() == 0)  {  CCLOG("Playing");    }  else if (item->getSelectedIndex() == 1)  {  CCLOG("Silence");  }    }  },unselectedItem,selectedItem, NULL);    //调整位置    toggleItem->setAnchorPoint(Vec2::ANCHOR\_MIDDLE\_RIGHT);  toggleItem->setPosition(Vec2(visibleSize / 2)+Vec2(-20,0));  //创建一个label菜单  auto label = Label::createWithSystemFont("Silence", "Arial", 20);  auto labelItem = MenuItemLabel::create(label,NULL);  labelItem->setAnchorPoint(Vec2::ANCHOR\_MIDDLE\_LEFT);  labelItem->setPosition(Vec2(visibleSize / 2));  //将菜单添加到场景  auto menu = Menu::create(toggleItem,labelItem, NULL);  menu->setPosition(Vec2::ZERO);  this->addChild(menu, 1);  return true;  }  void MyGameScene::menuCloseCallback(cocos2d::Ref \* pSender)  {    Director::getInstance()->end();  }  Button \* MyGameScene::createDirButton(float rotation, Vec2 position)  {  Button \* btn = Button::create("right1.png", "right2.png", "right3.png");  //设置按钮文本字体大小  btn->setTitleFontSize(12);  //设置按钮文本  btn->setTitleText("Click me");  btn->setTitleColor(Color3B::MAGENTA);  btn->setColor(Color3B::ORANGE);  btn->addTouchEventListener([&](Ref\* sender,Widget::TouchEventType type) {  switch (type)  {  case cocos2d::ui::Widget::TouchEventType::BEGAN:  break;  case cocos2d::ui::Widget::TouchEventType::MOVED:  break;  case cocos2d::ui::Widget::TouchEventType::ENDED:  CCLOG("you just clicked me...");  break;  case cocos2d::ui::Widget::TouchEventType::CANCELED:  break;  default:  break;  }  });  btn->setRotation(rotation);  btn->setPosition(position);  return btn;  } | // MyGameScene.h  #pragma once  #include"cocos2d.h"  #include"ui/CocosGUI.h"  USING\_NS\_CC;  using namespace ui;  class MyGameScene :public cocos2d::Scene {  public:  static cocos2d::Scene\* createScene();  virtual bool init();  // a selector callback  void menuCloseCallback(cocos2d::Ref\* pSender);  // implement the "static create()" method manually  CREATE\_FUNC(MyGameScene);  Button\* createDirButton(float rotation, Vec2 position);  private:  //菜单  Menu\* prt\_menu;  //文本菜单项  MenuItemLabel\* prt\_labelItem;  //开关菜单项  MenuItemToggle\* prt\_toggleItem;  //选中精灵菜单项  MenuItemSprite\* prt\_selectedItem;  //未选中精灵菜单项  MenuItemSprite\* prt\_unselectedItem;  }; |

效果“

|  |  |
| --- | --- |
|  |  |

”