

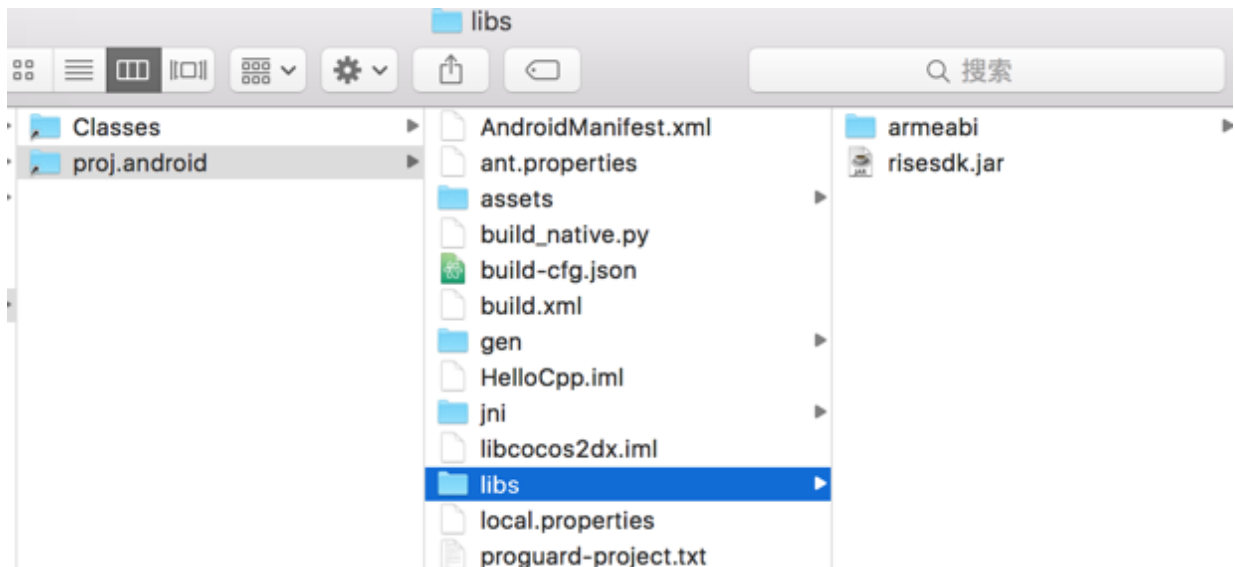
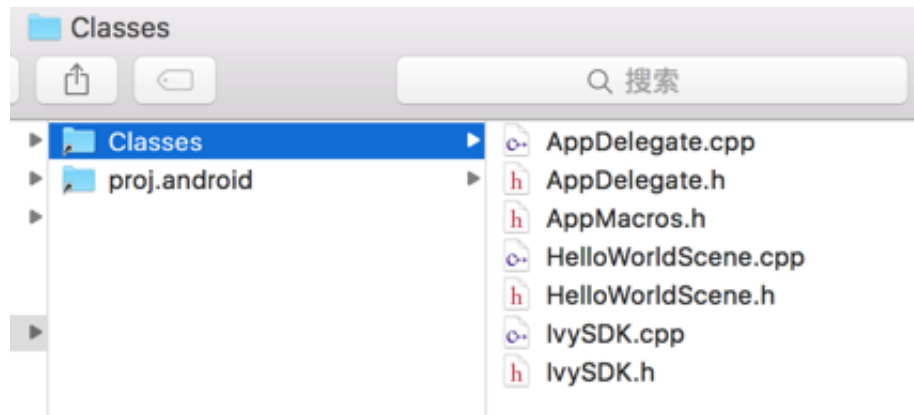
RiseSDK for cocos2d-x

1, Copy files and init in Activity

- copy Classes and proj.android to your Cocos2d-x project folder corresponding
 - before:



– after:



- init in your main activity

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    Cocos.onCreate(this);
}

@Override
protected void onResume() {
    super.onResume();
    Cocos.onResume(this);
}

@Override
protected void onPause() {
    super.onPause();
    Cocos.onPause();
}

@Override
protected void onDestroy() {
    Cocos.onDestroy();
    super.onDestroy();
}

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    Cocos.onActivityResult(requestCode, resultCode, data);
    super.onActivityResult(requestCode, resultCode, data);
}

```

- if you use proguard to obfuscate your java source code, you should add these rules to your proguard rules file:

```

-keep class com.risesdk.client.** {
    <methods>;
}

-keep class android.support.** {
    *;
}

```

2, add IvySDK.cpp to your Android.mk file

```
LOCAL_SRC_FILES := hellocpp/main.cpp \  
                  ../../Classes/AppDelegate.cpp \  
                  ../../Classes/HelloWorldScene.cpp \  
                  ../../Classes/IvySDK.cpp
```

3, call these functions when you want

```
//show pass level full screen ad when you want  
IvySDK::showInterstitial(IvySDK::AD_POS_GAME_PASSLEVEL);  
  
// show pause ad when you want  
IvySDK::showInterstitial(IvySDK::AD_POS_GAME_PAUSE);  
  
// show banner at the top center position of your phone  
IvySDK::showBanner(IvySDK::AD_POS_MIDDLE_TOP);  
  
// close banner when needed  
IvySDK::closeBanner();  
  
// when you want to quit the game  
IvySDK::onQuit();  
  
// ask the player to share your game with his friends  
IvySDK::share();  
  
// ask the player to give you a 5-star-rating  
IvySDK::rateUs();  
  
// show more game to the player  
IvySDK::showMoreGame();  
  
// when you want to track the player behavior  
IvySDK::trackEvent("your category", "your action", "your label", 1);  
  
// get your custom data from server, return "{}" default  
const char* data = IvySDK::getExtraData();
```

4, When you want to use IAP, you should do this:

- register your payment callback

```

// define your bill ids
#define BILLING_ID_ACTIVE_GAME 1
#define BILLING_ID_BUY_CAR 2
... etc.

static void onPaymentSuccess(int billId) {
    switch (billId) {
        case BILLING_ID_ACTIVE_GAME:
            CCLOG("game actived!");
            // do your logic here
            break;

        case BILLING_ID_BUY_CAR:
            // do your logic here
            break;
    }
}

// define your callback function, this should be a global function or
// a member function of a class
void HelloWorld::onPaymentResult(int resultCode, int billingId) {
    CCLOG("billing %i result code %i", billingId, resultCode);
    switch (resultCode) {
        case IvySDK::PAYMENT_RESULT_SUCCESS:
            onPaymentSuccess(billingId);
            break;

        default:
            CCLOG("billing %i result code %i", billingId, resultCode);
            break;
    }
}

// call register in your initialize function
bool HelloWorld::init() {
    ...
    IvySDK::registerPaymentCallback(onPaymentResult);
    ...
}

```

- call doBilling when you want

```
IvySDK::doBilling(BILLING_ID_ACTIVE_GAME);
```

5, when you want to use reward ad, you should do this:

- register your reward ad callback

```
#define REWARD_ID_GOLD 1
#define REWARD_ID_CAR 2
... etc.

// define your callback function, this should be a global function or
// a member function of a class
void onFreecoinResult(int rewardId) {
    CLOG("receive reward %i", rewardId);
    // do your logic here
    switch(rewardId) {
        case REWARD_ID_CAR:
            // do your logic
            break;

        case REWARD_ID_GOLD:
            //do your logic
            break;

        ...
    }
}

// call register in your initialize function
bool HelloWorld::init() {
    ...
    IvySDK::registerFreecoinCallback(onFreecoinResult);
    ...
}
```

- call showFreeCoin when you want

```
// launch reward ad for gold
IvySDK::showFreeCoin(REWARD_ID_GOLD);

// determine whether the reward ad is available
bool has = IvySDK::hasFreeCoin();
if (has)...
```

6, Congratulations, done. You will see some toasts when you run your game in your android phone or emulator:

