

主要实现步骤

- 1, 将代码打包成 jar 包。
- 2, 整合资源文件
- 3, 将 jar 包打包成 exe
- 4, 将 jdk、资源文件、jar 包转换后的 exe 三者再次打包成最终的 exe。

准备软件

- 1, Idea : 将代码打包成 jar 包 (java 形式的压缩包)
- 2, exe4j : 将 jar 包转换成 exe 的工具。
- 3, innosetup : 将游戏用到的图片, Java 的运行环境和第二步打包的代码, 组合成最终的安装程序。

备注 1 : exe4j 和 innosetup 安装非常简单, 傻瓜式下一步即可。

备注 2 : exe4j 支持的 JDK 版本是 8~11, 其他版本的 JDK 不行, 所以我也给大家提供了 JDK11 版本。

备注 3 : 64 位的操作系统用 exe4j_windows-x64_6_0_2.exe

32 位的操作系统用 exe4j_windows_6_0_2.exe

备注 4 : 打开我的电脑, 右键空白处, 点击属性, 在弹出界面的中央就能查看自己电脑是 64 位的还是 32 位的。

软件下载地址

在《day18-API》--- 《资料》--- 《拼图游戏打包成 exe》中

软件安装

- 1, 确保 idea 已经安装完毕
- 2, 安装 exe4j
- 3, 安装 innosetup

傻瓜式安装，直接下一步即可，但是要记住安装路径。

代码改写：

代码中所有图片用到的路径需要把模块名删除，改写之后如下：

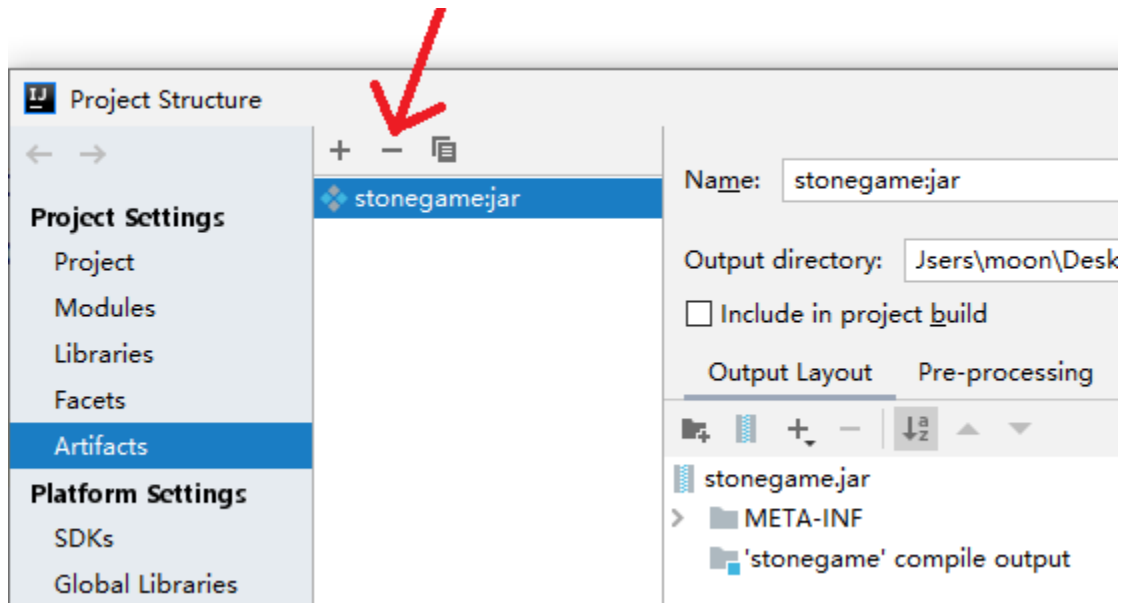
在不同的情况下，相对路径的相对位置是不一样的。

```
public void initView() {  
    //1. 添加用户名文字  
    JLabel usernameText = new JLabel(new ImageIcon( filename: "image\\login\\用户名.png"));  
    usernameText.setBounds( x: 116, y: 135, width: 47, height: 17);  
    this.getContentPane().add(usernameText);  
  
    //2. 添加用户名输入框  
  
    username.setBounds( x: 195, y: 134, width: 200, height: 30);  
    this.getContentPane().add(username);  
  
    //3. 添加密码文字  
    JLabel passwordText = new JLabel(new ImageIcon( filename: "image\\login\\密码.png"));  
    passwordText.setBounds( x: 130, y: 195, width: 32, height: 16);  
    this.getContentPane().add(passwordText);  
}
```

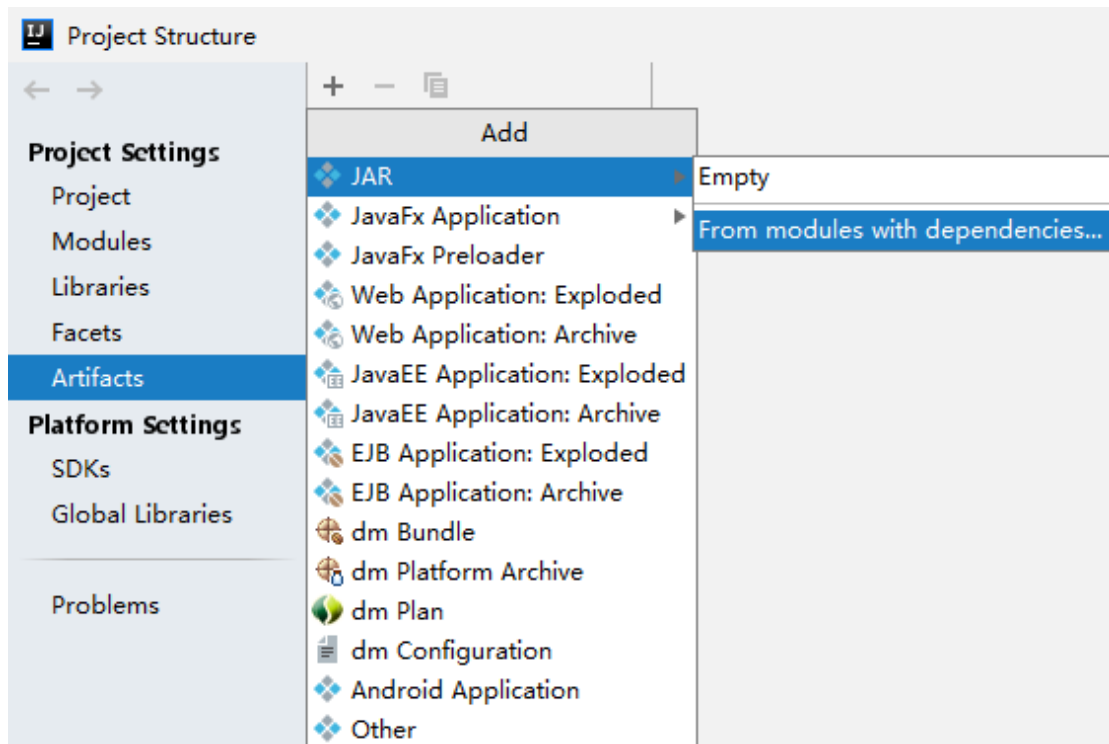
打包步骤：

第一步：代码打包成 jar 包

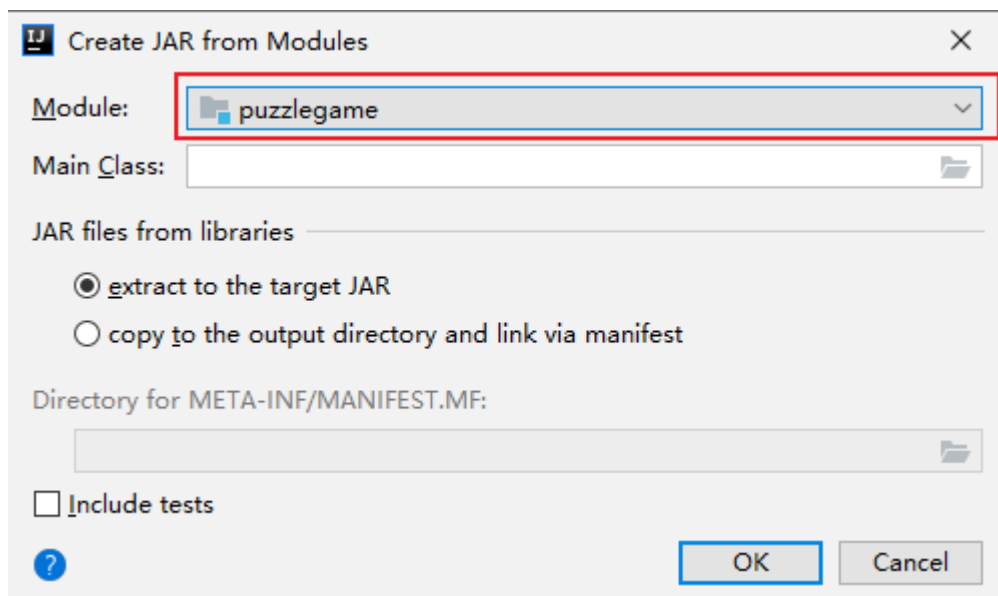
- 1， 点击 File，再点 Project Structure。
- 2， 点击 Artifacts。
- 3， 下方图片箭头指向位置应为空白 ,如果有其他内容 ,可以选中之后，点击减号删除。



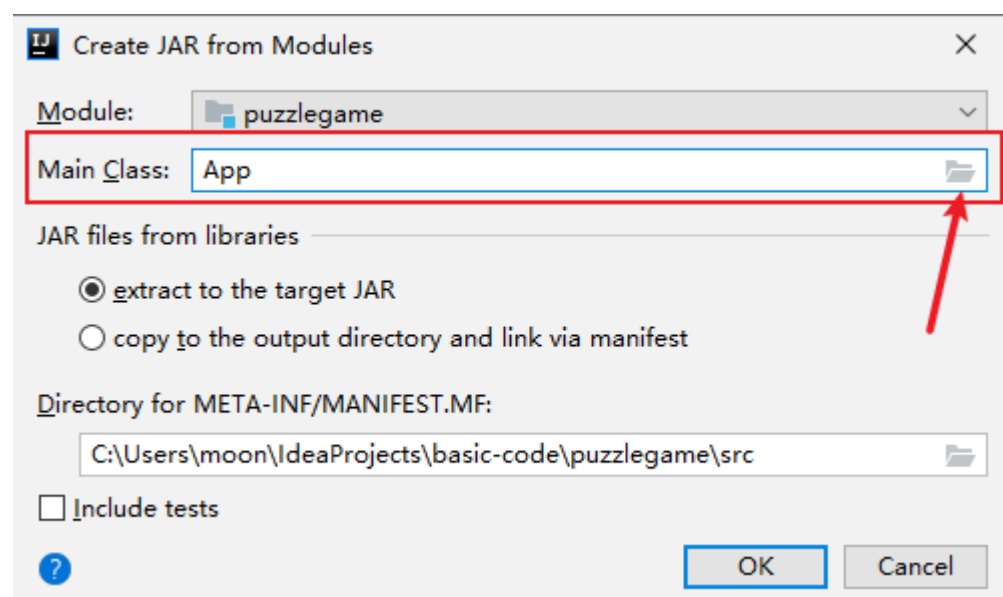
- 4， 选中左侧的 Artifacts，
点击中间上方位置的+
点击 JAR
点击 From modules with dependencies...
如下图所示：



5, Module : 选择要打包的项目。

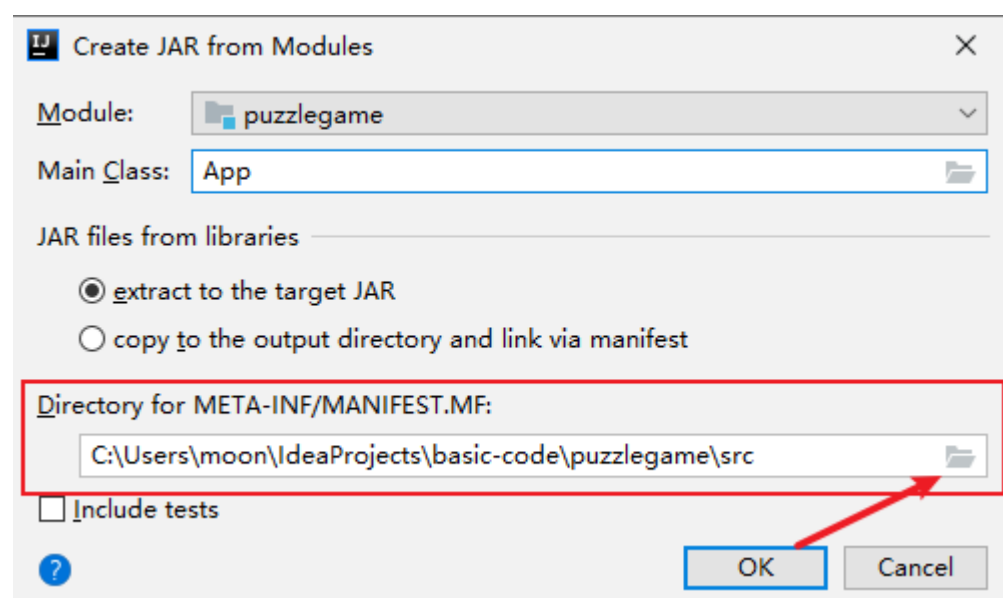


6 ,Main Class : 点击红框最后的那个小图标 项目中是哪个 main 方法所在的类去启动项目，这里就选择哪个类。我的项目是由 app 里面的 main 方法启动的，所以我就选择 App



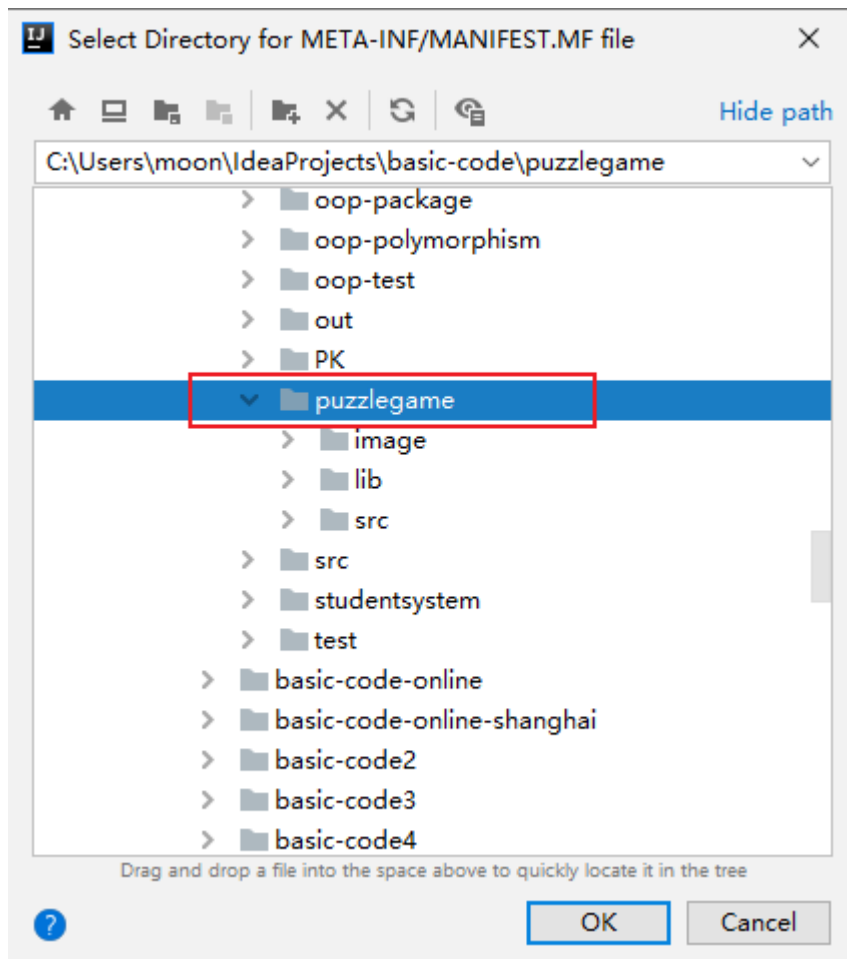
7, 设置 META-INF/MANIFEST.MF。

点击箭头指向的图标进行设置。

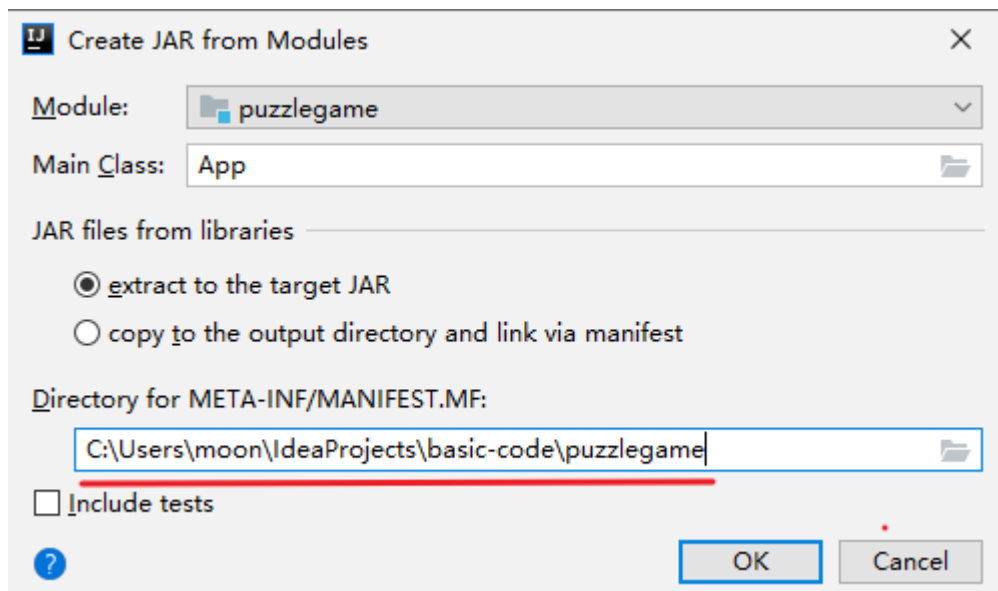


8, 选择当前模块

点击 OK



9， 此时这里的路径就是模块所在路径，点击 OK。



10， 如果没有弹框报错，第十步可以跳过，直接看十一步。

如果出现弹框报错，表示当前模块下已存在 META-INF 文件夹

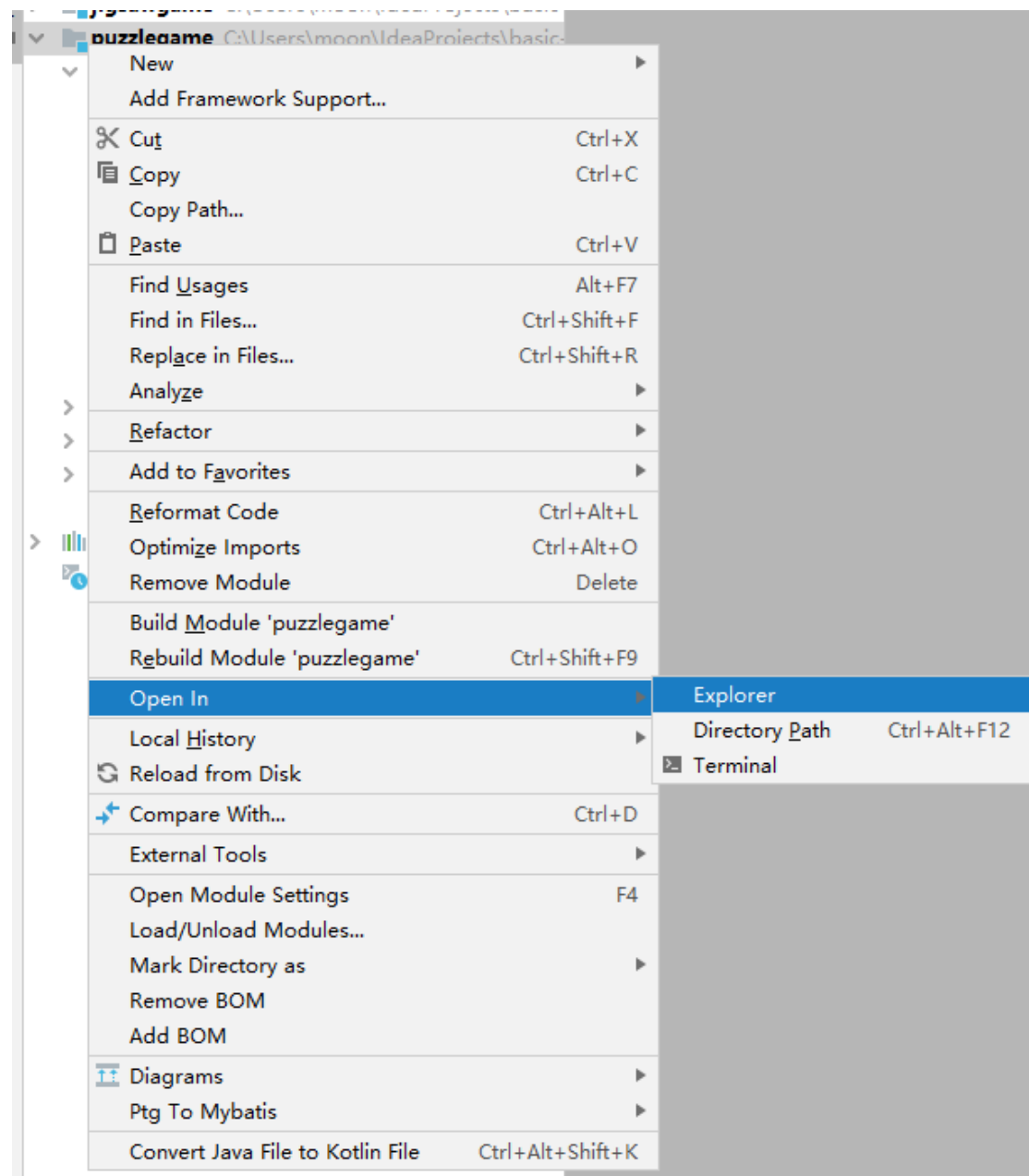
了。可以在本地找到已存在的 META-INF 文件夹，右键点击 DELETE 删除即可。

查找方式如下：

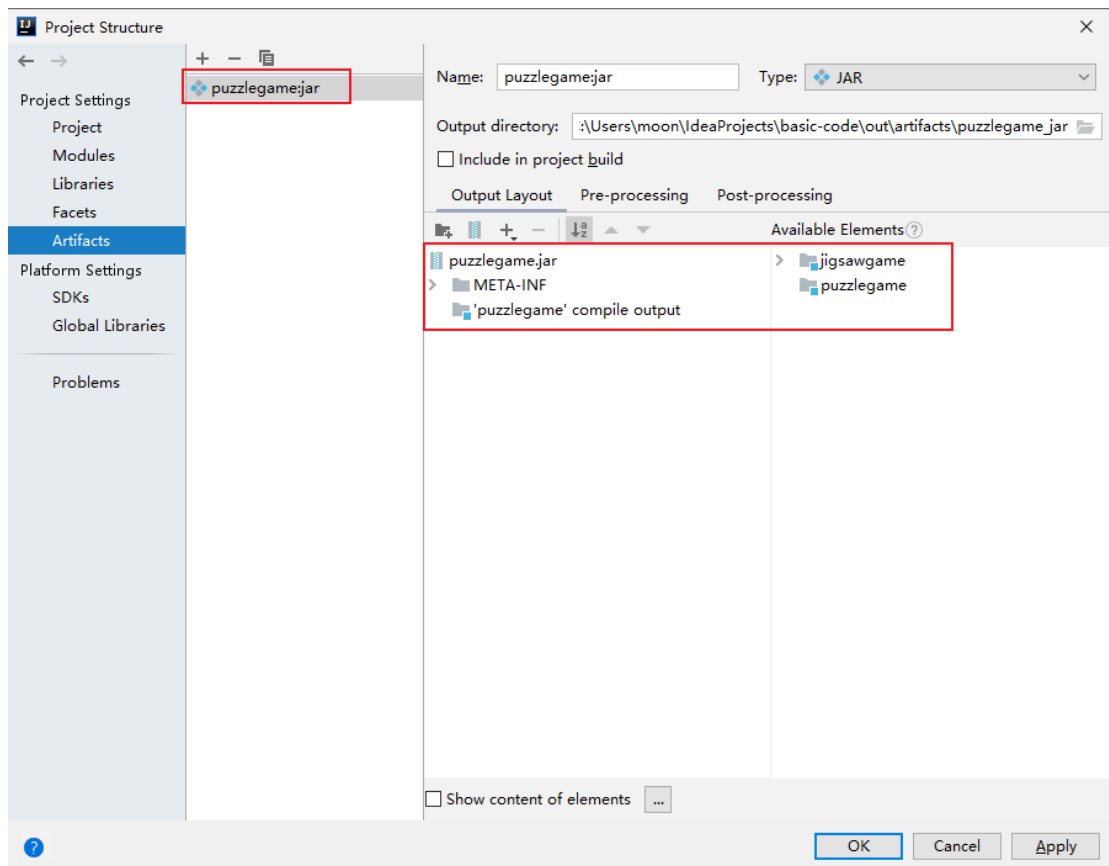
右键模块点击 open in

点击 Explorer

在模块里面找到 META-INF 文件删除即可。

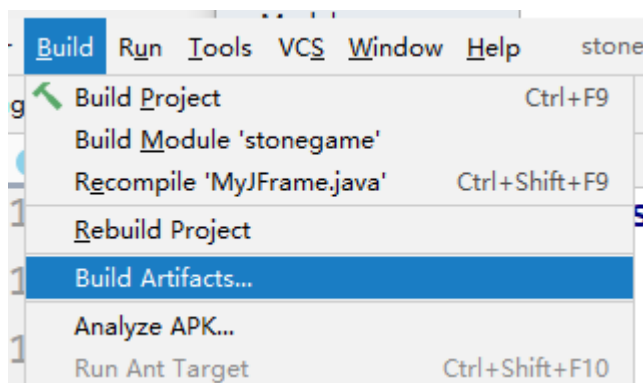


11，如果没有报错，就出现一个 puzzlegame : jar 的提示。
点击右下角的 OK。



12，在 idea 主界面上方，点击 Build 里面的 Build Artifacts。
如果是灰色的不能按，请确定在上面第一步~第九步有没有配置好 jar 包的信息。

如果已经配置好了 jar 的信息，此处就可以正常点击。

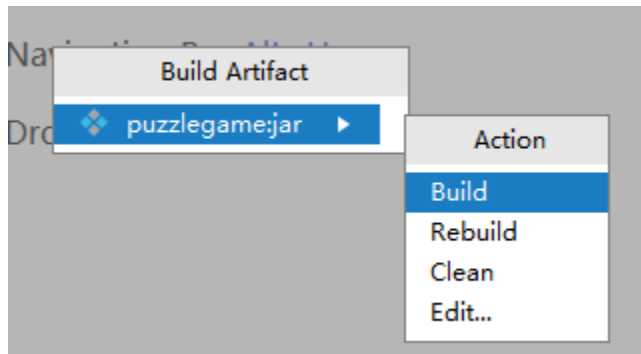


13，在主界面正中央。

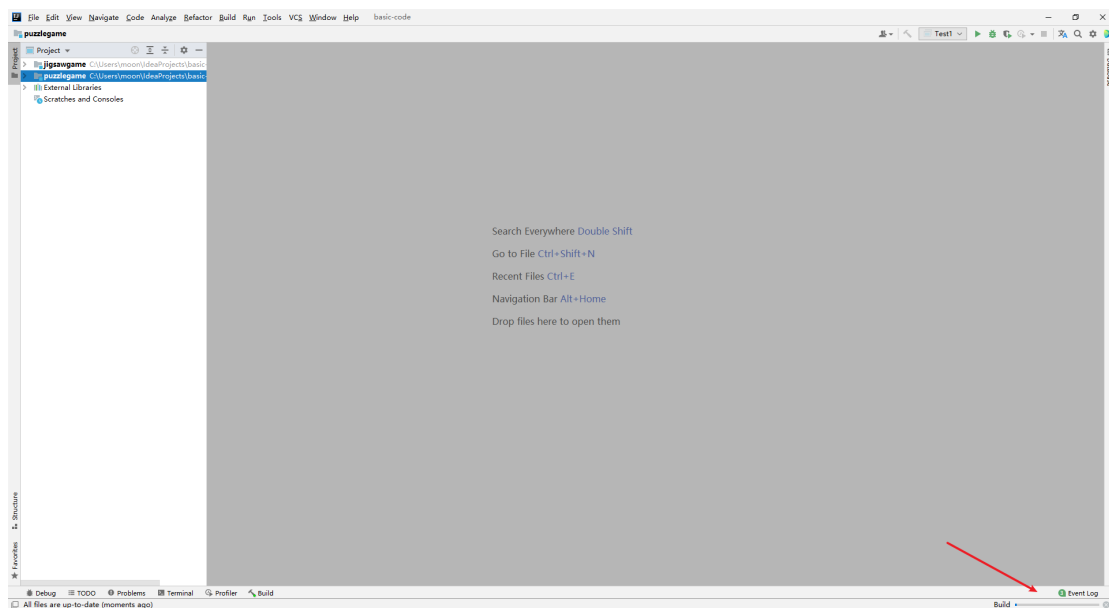
选择 puzzlegame:jar。

再选择 Build。

idea 会帮我们生成 jar 包。



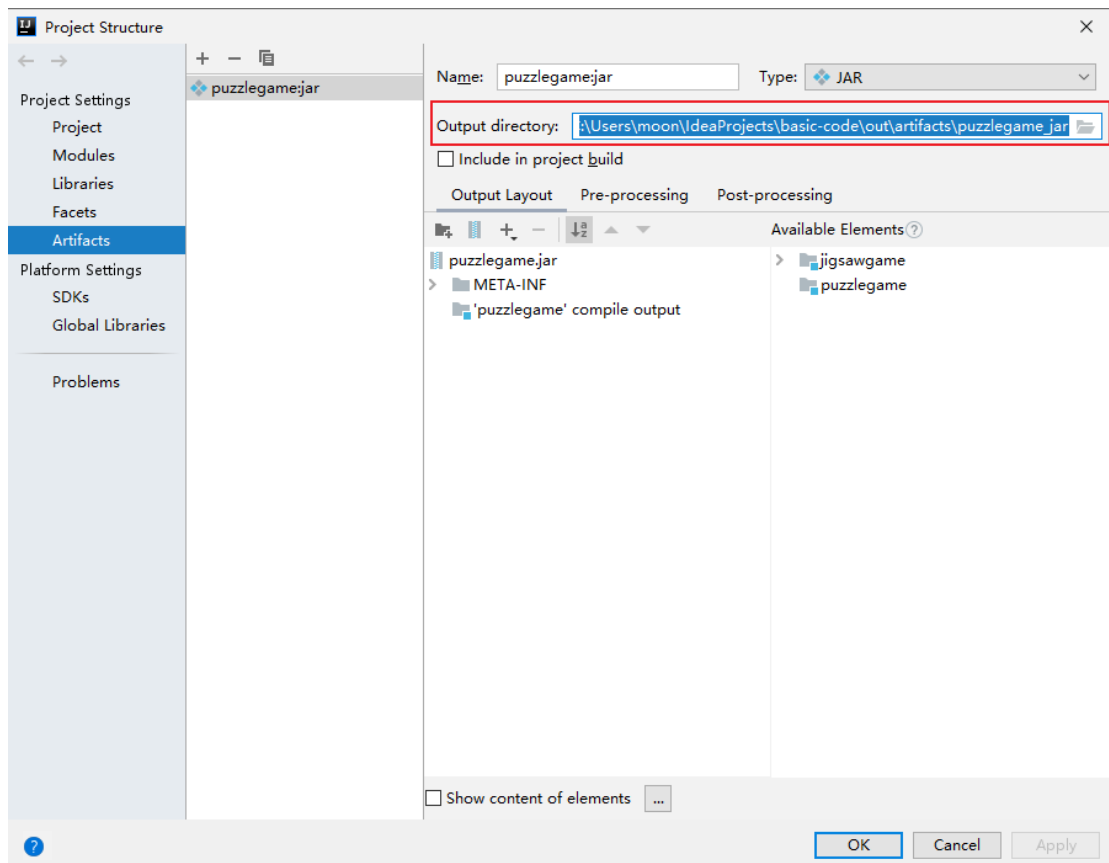
14，等页面右下角的进度条结束，jar 就已经生成完毕。



15，点击 File，再点 Project Structure。

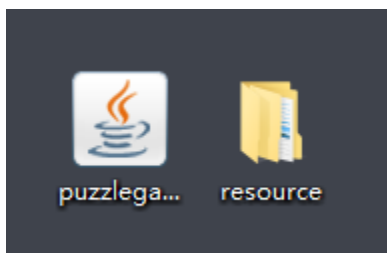
点击 Artifacts。

在下图红色框中的路径下，可以找到生成好的 jar 包。



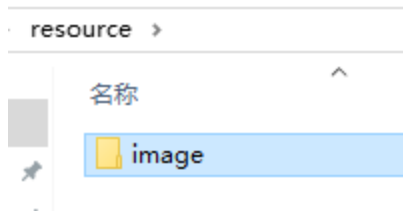
第二步：整合资源文件

- 1， 将第一步创建好的 jar 包拷贝到桌面上。
- 2， 在桌面上新建一个文件夹 resource。



- 3， 将《拼图游戏》中的 image 文件夹粘贴到 resource 文件夹当中。

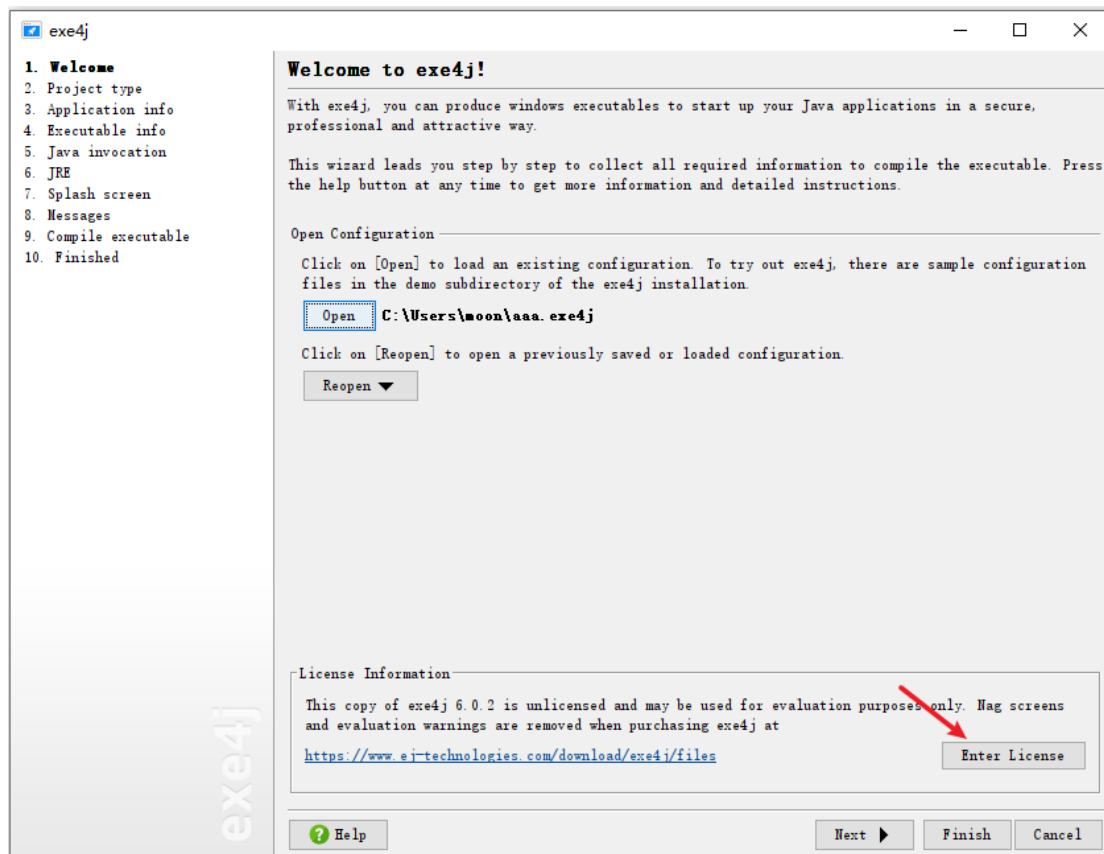
此时在桌面中的 resource 文件夹下的 image 文件夹下，就会看到游戏里面用到的所有图片。



第三步：将 jar 包打包成 exe

1， 双击打开安装好的 exe4j.exe，注册软件。

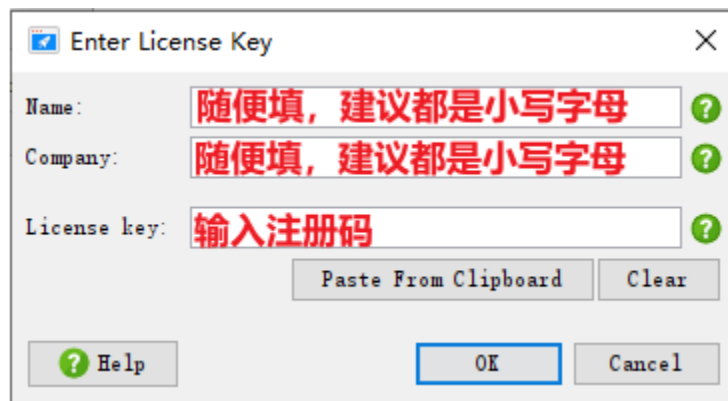
点击下面的 Enter License



2， 输入用户名，公司名和注册码后点击 ok

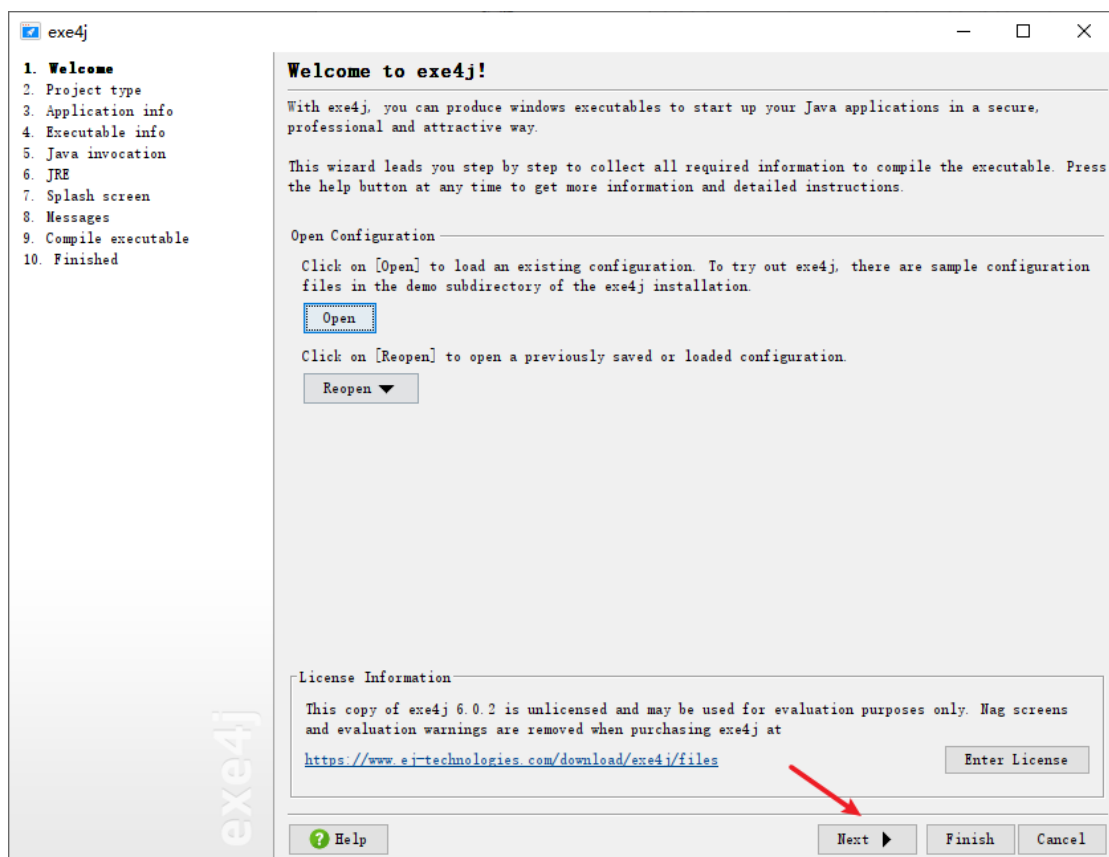
用户名，公司名随便填，最好都是小写字母。

注册码：L-g782dn2d-1f1yqxx1rv1sqd



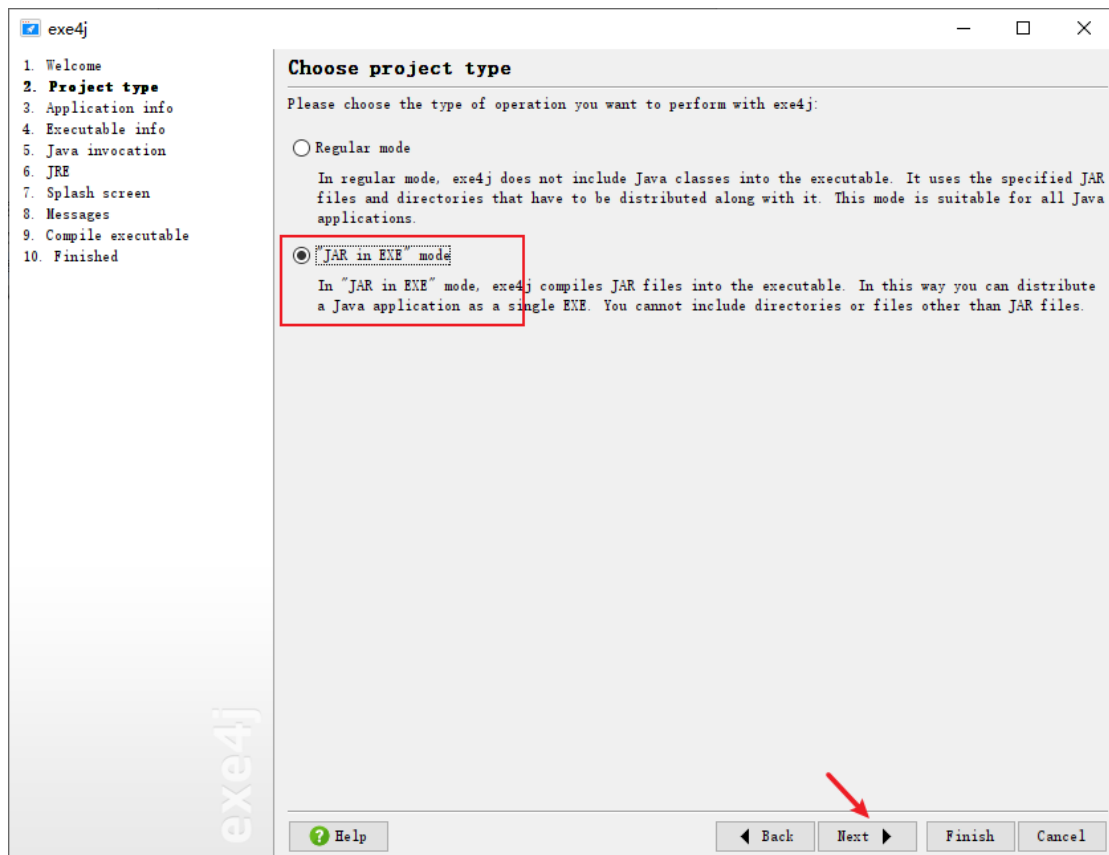
3 , 注册完毕之后点击右下角的 next

备注 :exe4j.exe 只识别 JDK8 到 JDK11 ,如果安装时出现弹框报错 ,请检查 JDK 版本 ,资料文件夹中给大家提供了 JDK11 ,有需要可以安装。(黑马程序员阿玮老师出品)



4 , 选择 JAVA 转 EXE。

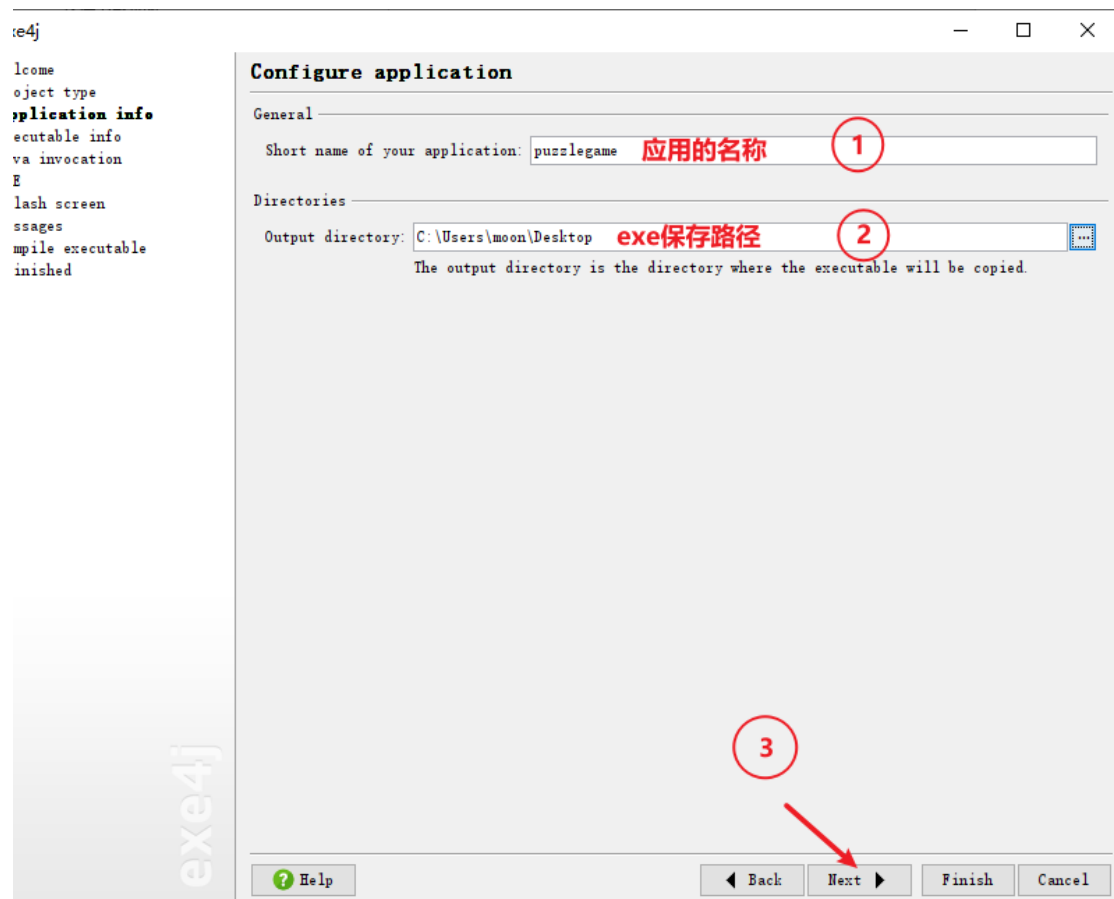
点击右下角的 next



5, 输入名称 puzzlegame

输出保存 exe 的路径, 建议选择到桌面上。

点击右下角的 next



6, 选择以图形界面的形式启动游戏

输入应用名称, puzzlegame

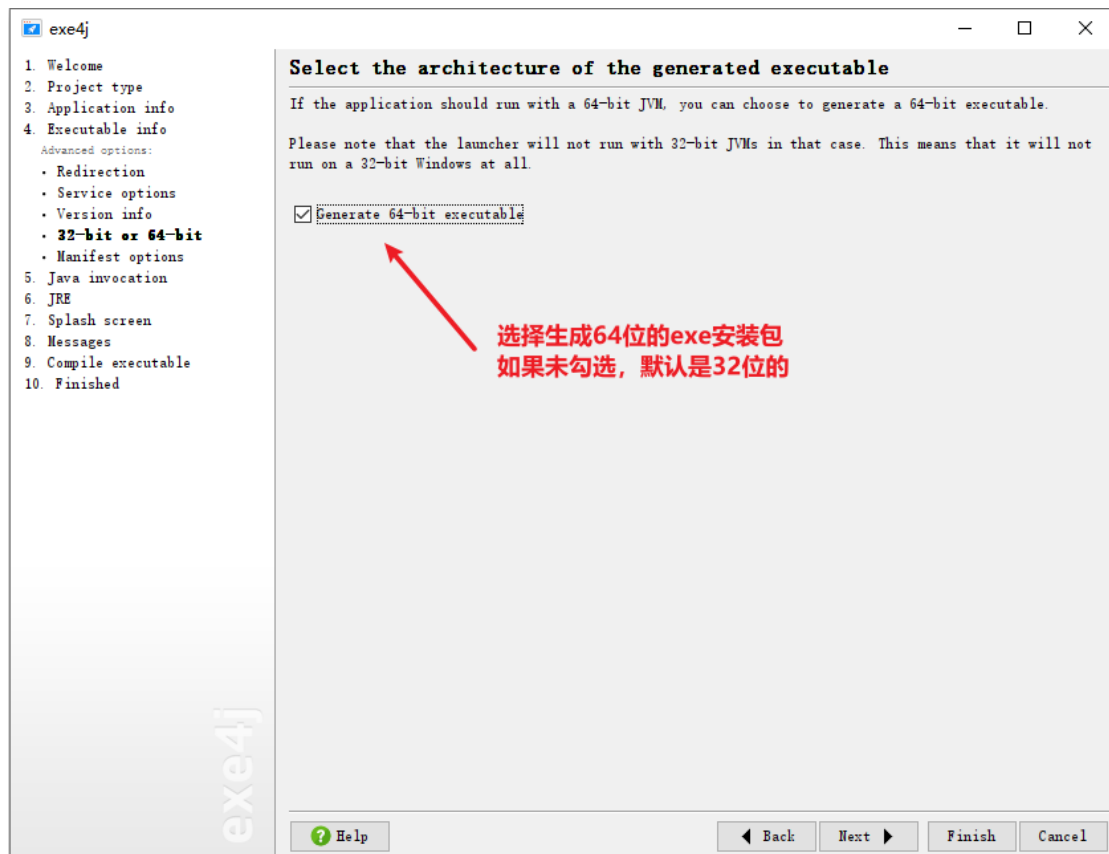
点击高级设置 Advanced Options, 选择 32-bit or 64-bit



7, 勾选 Generate 64-bit executable

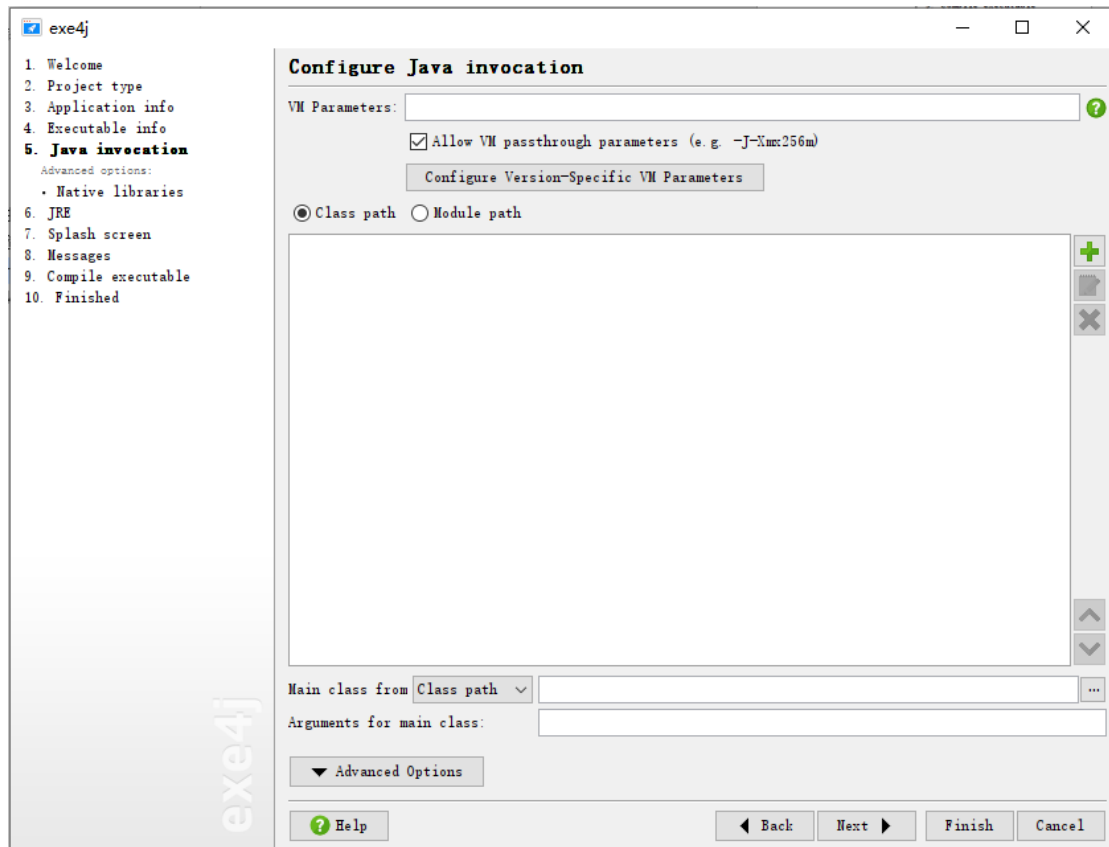
表示要生成 64 位的 exe 安装包, 如果未勾选默认生成 32 位的安装包。

点击右下角 next

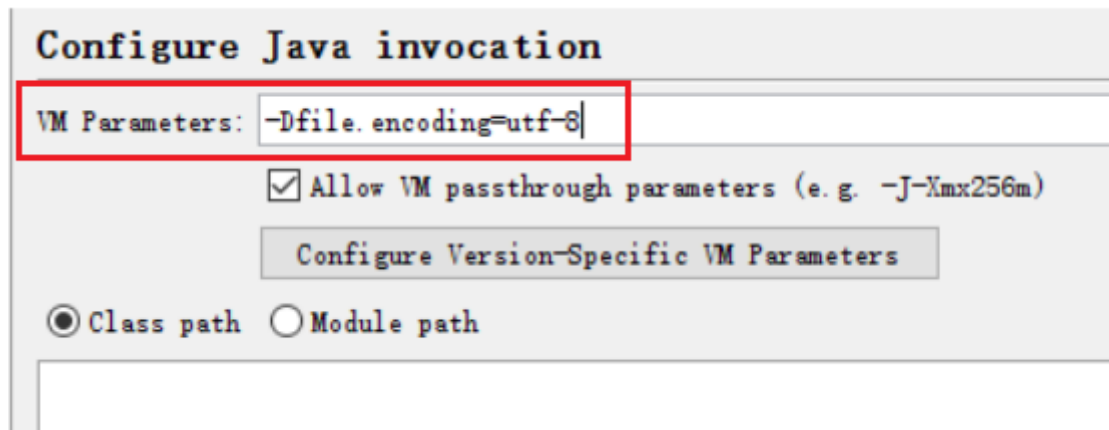


8, 然后一直下一步, 一直出现如下界面

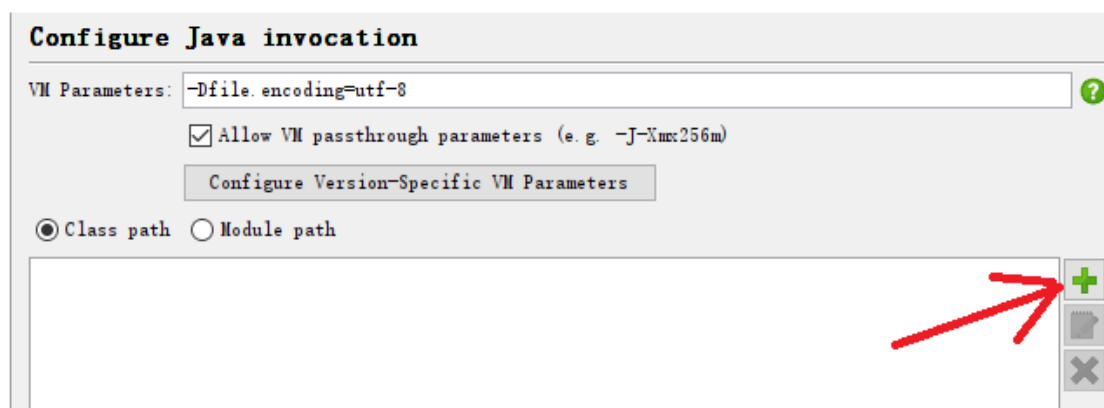
在本界面开始 jar 包并以及配置启动类。(黑马程序员阿玮老师出品)



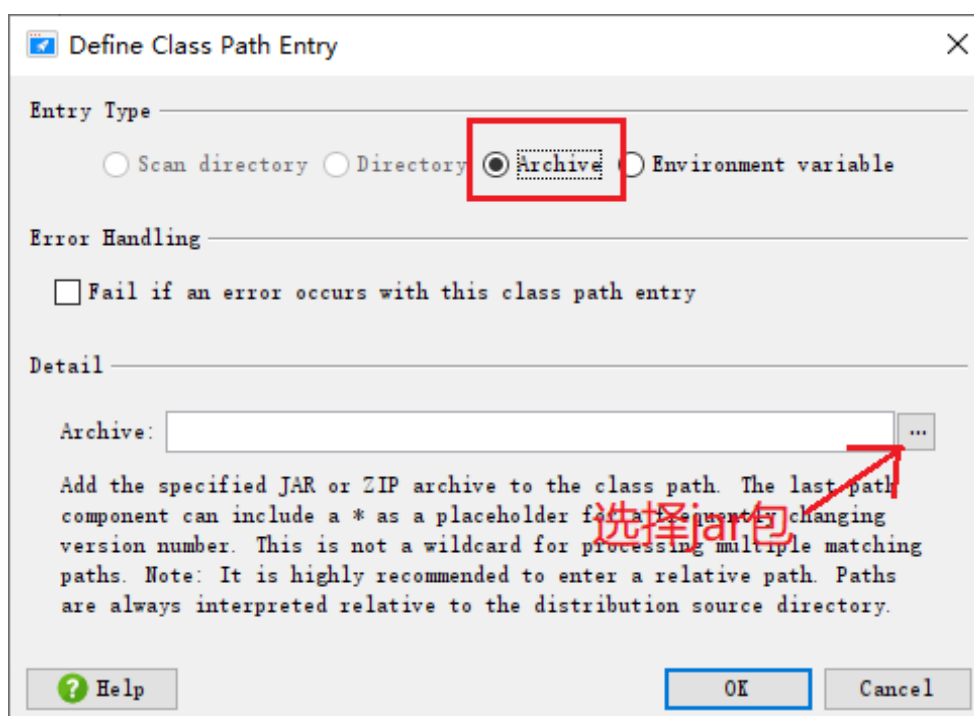
9, 第一行, VM 参数配置的地方加上: `-Dfile.encoding=utf-8`



10, 点击右侧绿色的+, 添加jar包

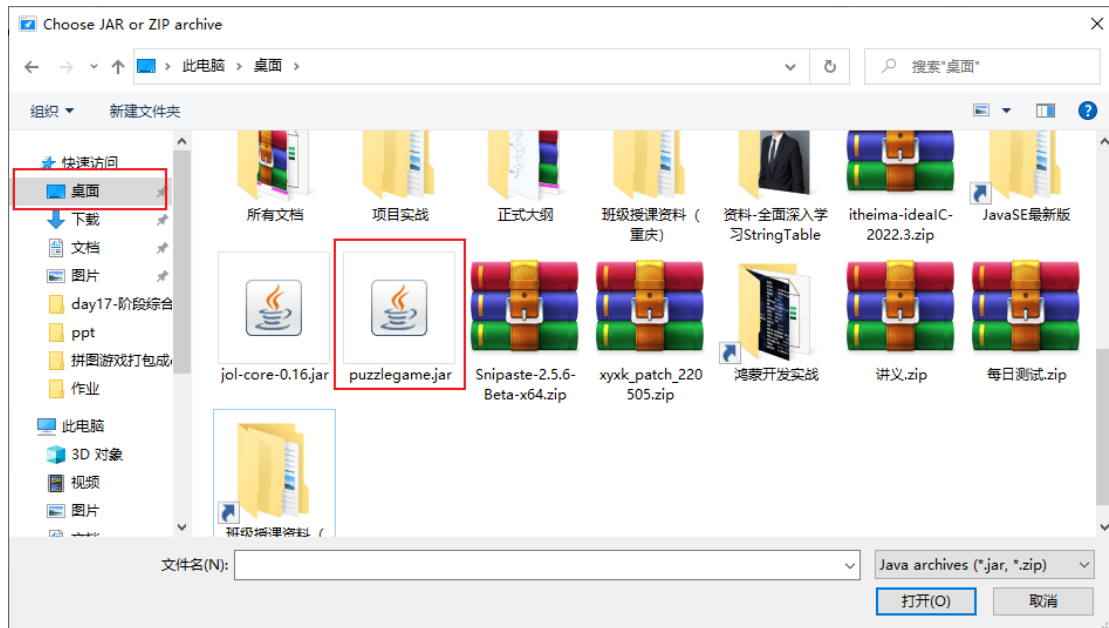


11，添加jar 包

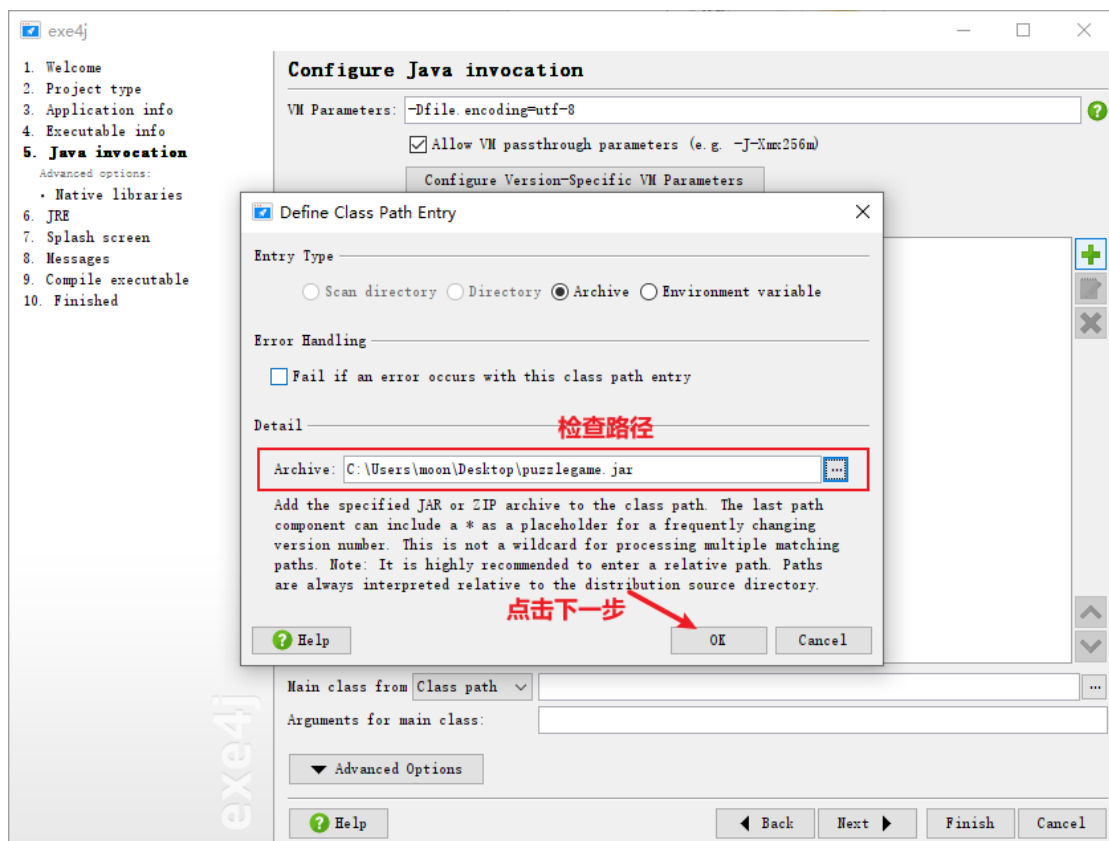


12，选择桌面上的 puzzlegame.jar。

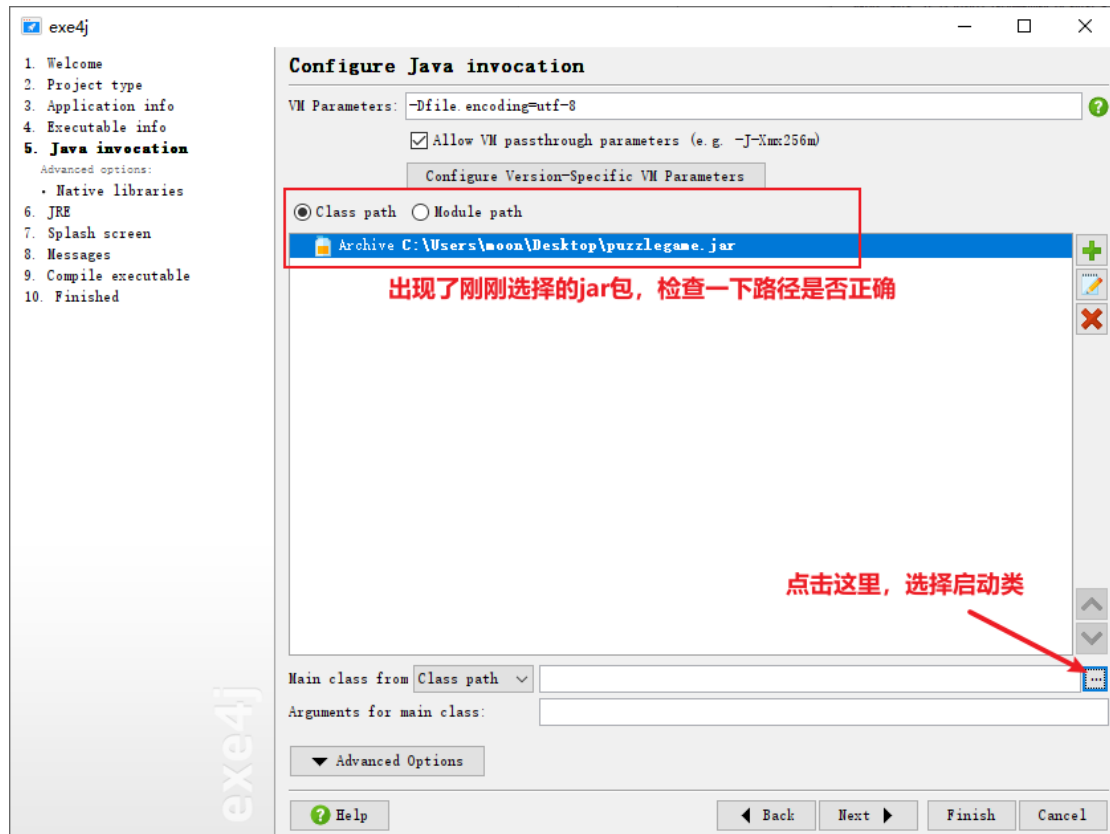
点击下面的打开。（黑马程序员阿玮老师出品）



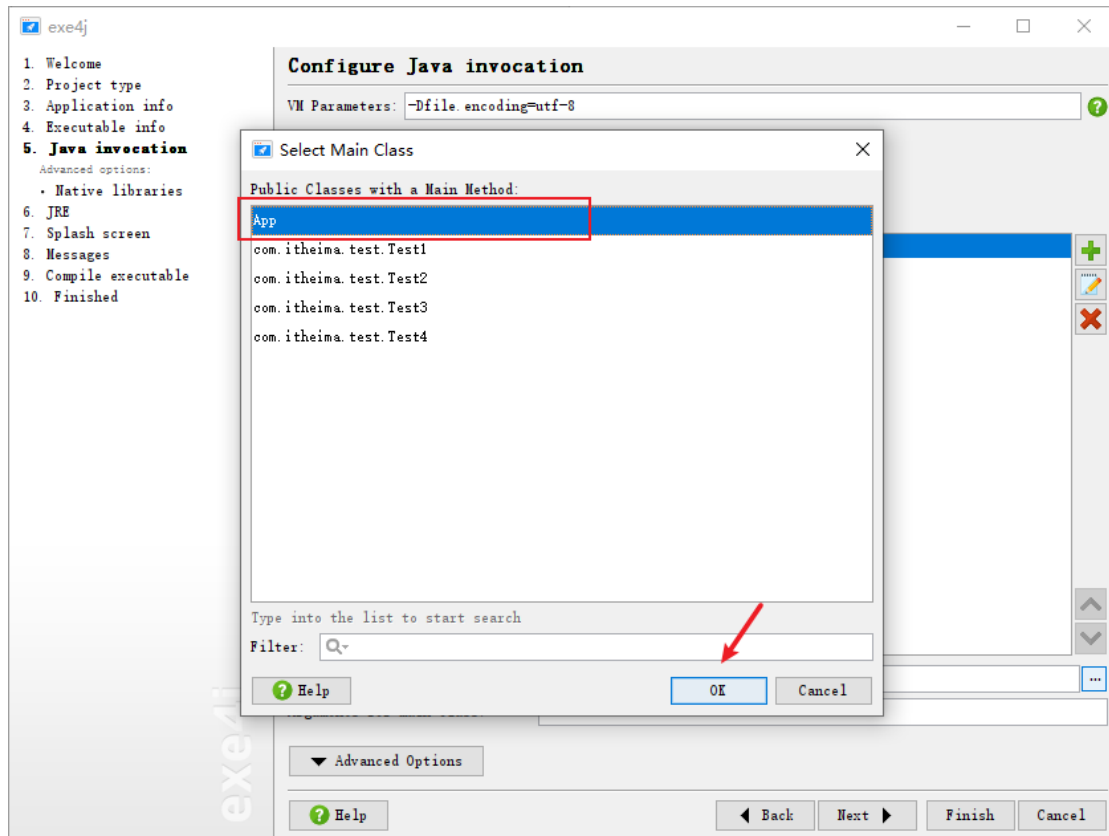
13, 检查路径, 如果无误点击右下角的 OK



14, 选择项目启动类

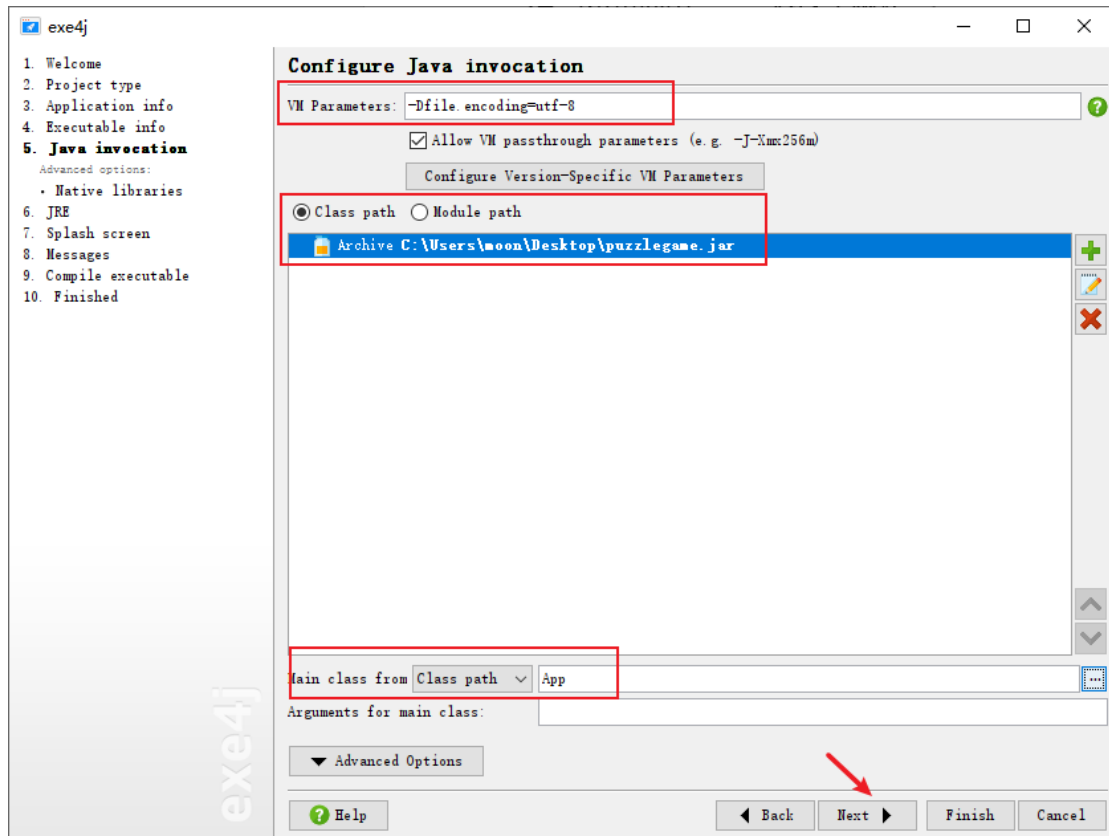


15, 因为程序主入口 main 方法写在 App 类中, 所有选择 app, 并点击 OK。



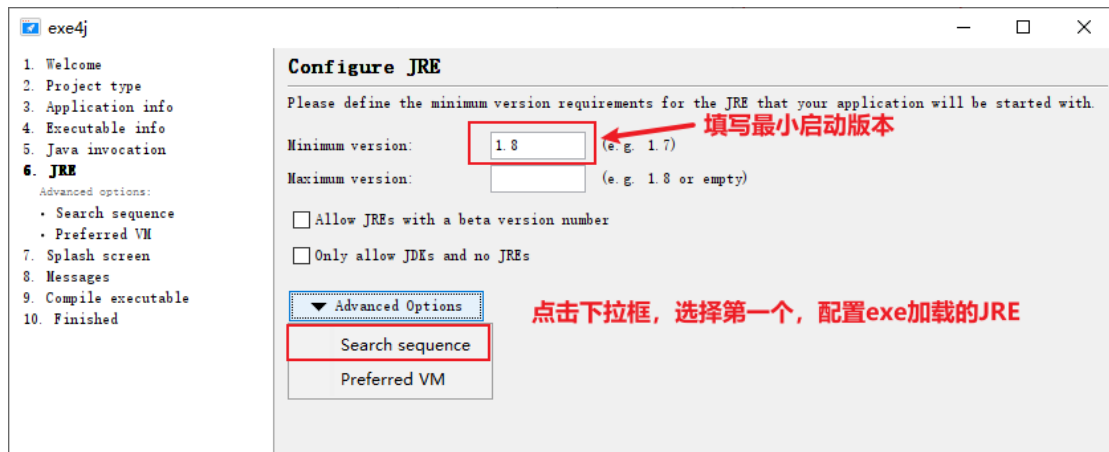
16，本页面中，一共修改了三处。

三处全部操作完毕。点击右下角的 next

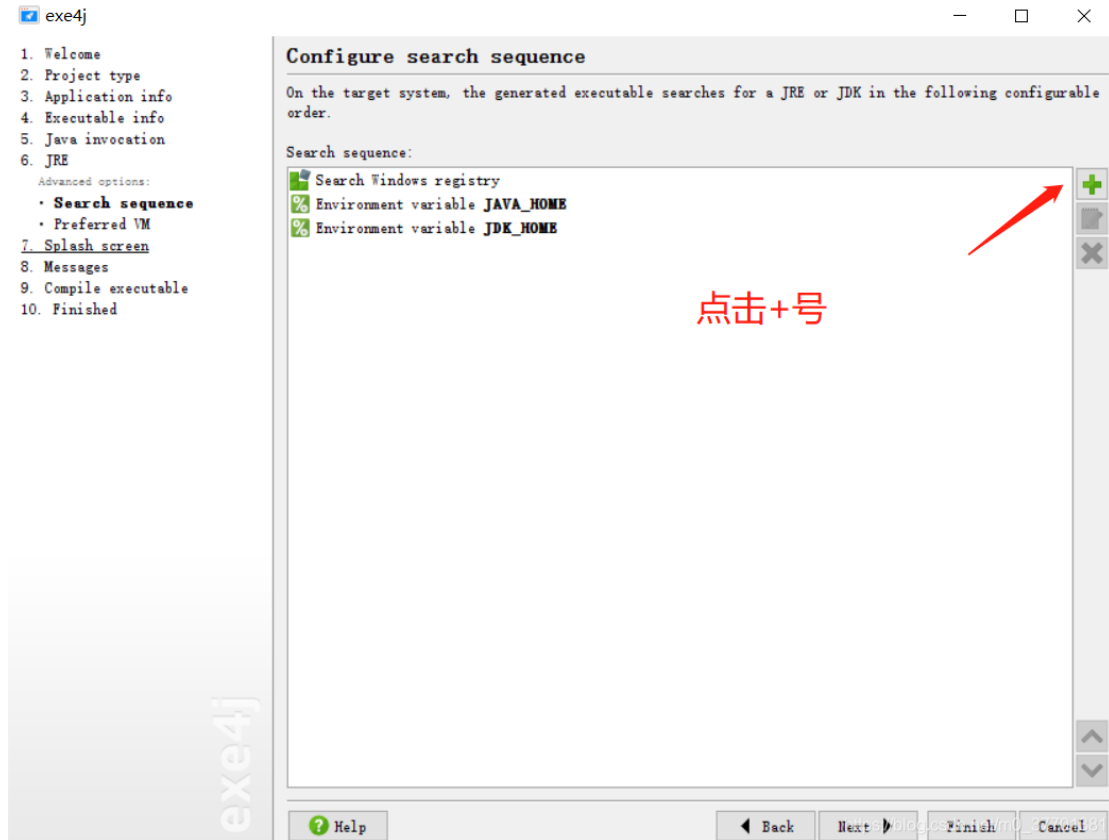


17, 填写最小启动的 JDK 版本。输入 1.8

配置 exe 加载的 JDK, 选择第一个。



18, 点击+

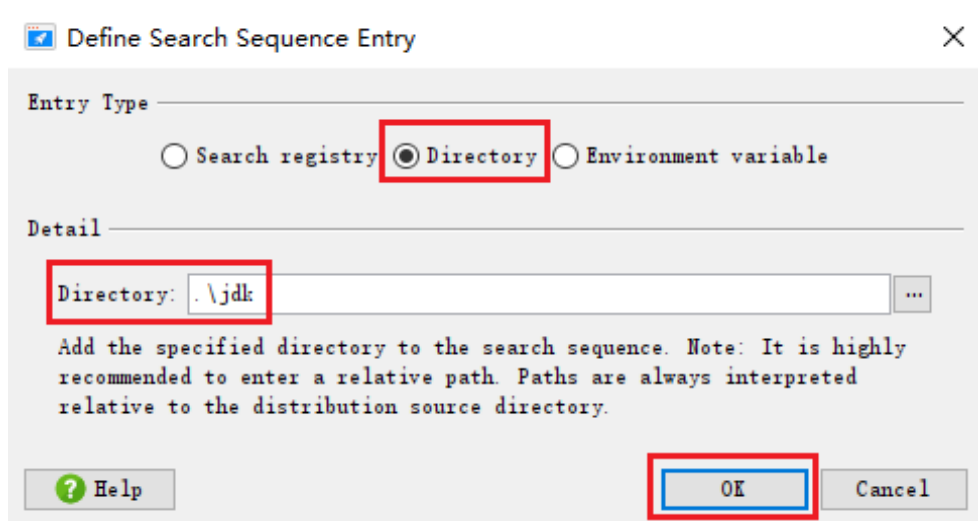


19，选择 Directory

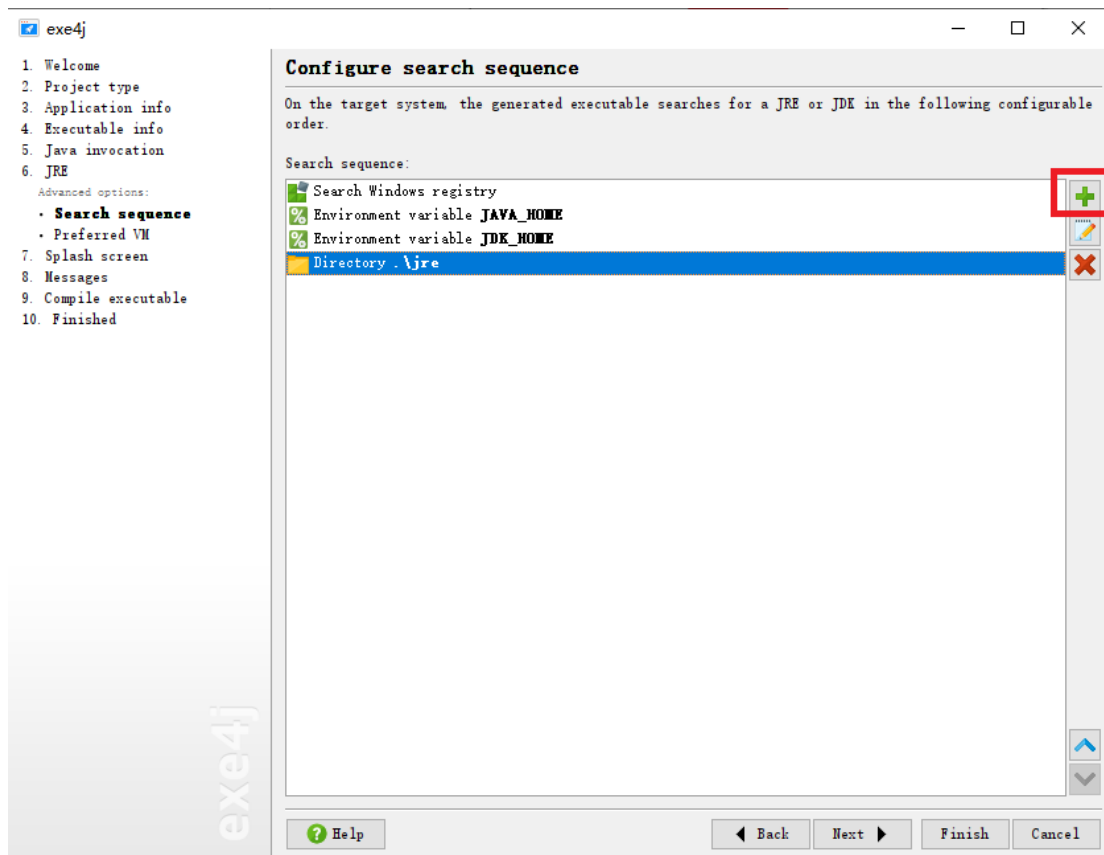
下面输入.\jdk

(注意：输入点杠 jdk，都要是英文状态下输入)

点击 OK



20，再次点击+

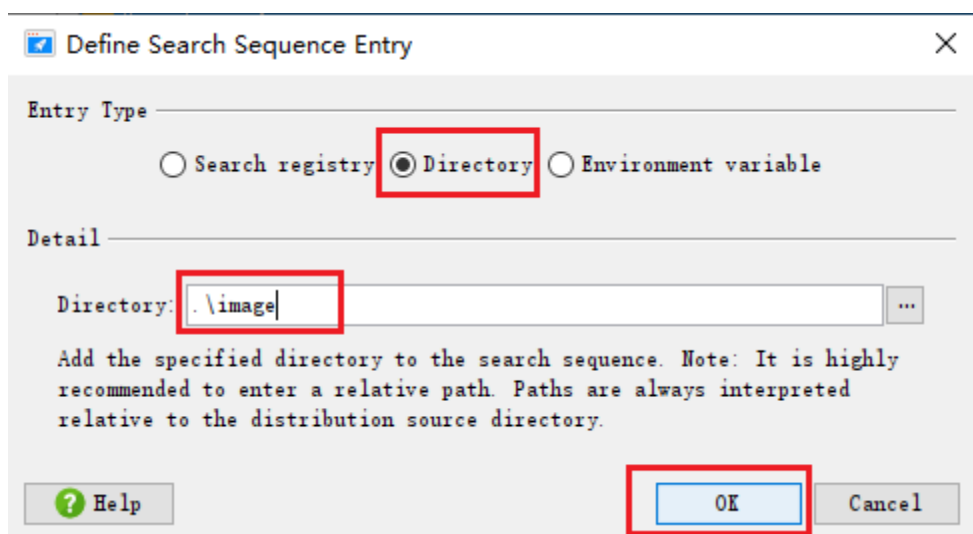


21, 选择 Directory

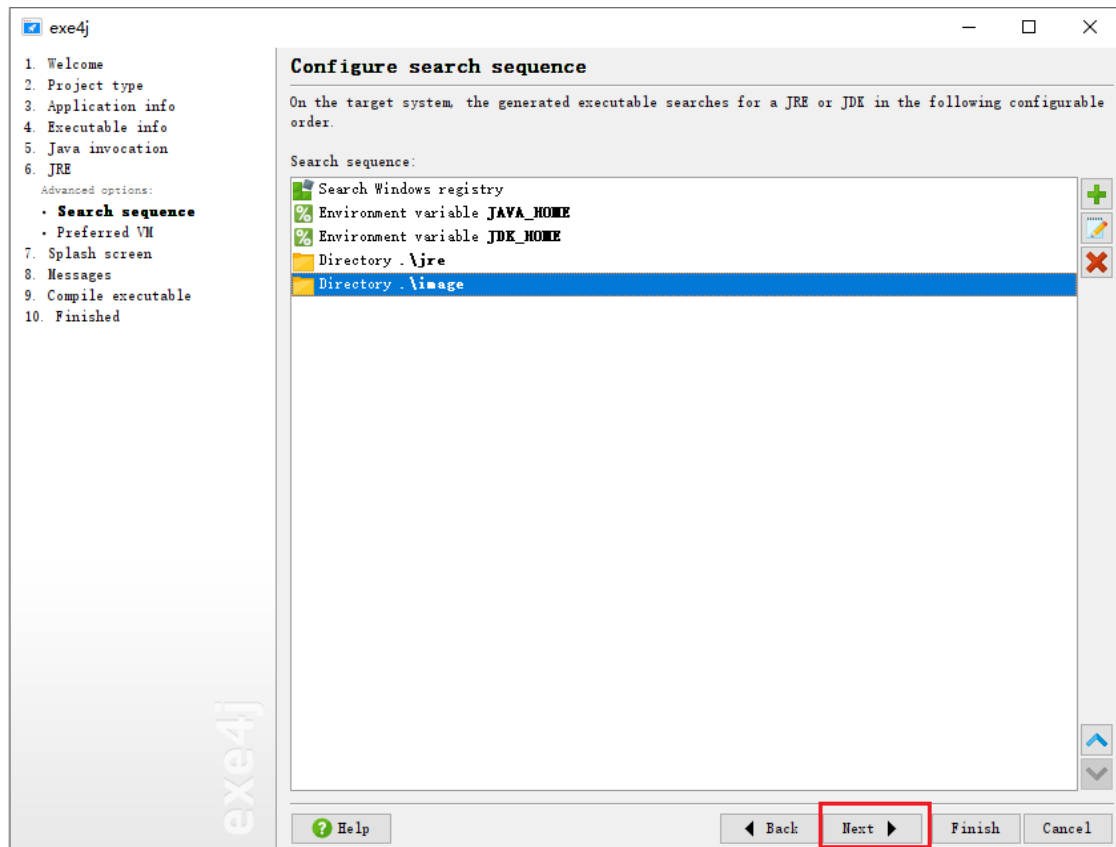
下面输入.\image

(输入点杠 image, 都要是英文状态下输入)

点击 OK

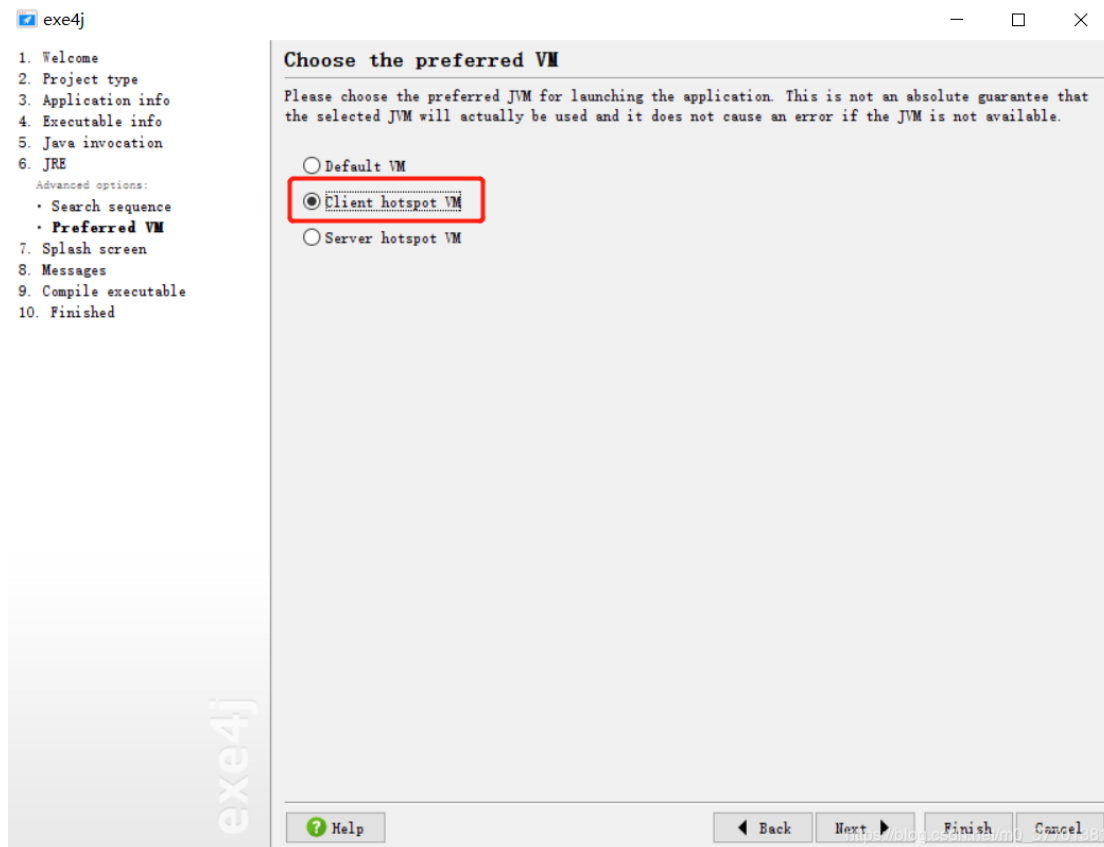


22, 点击右下角的 next

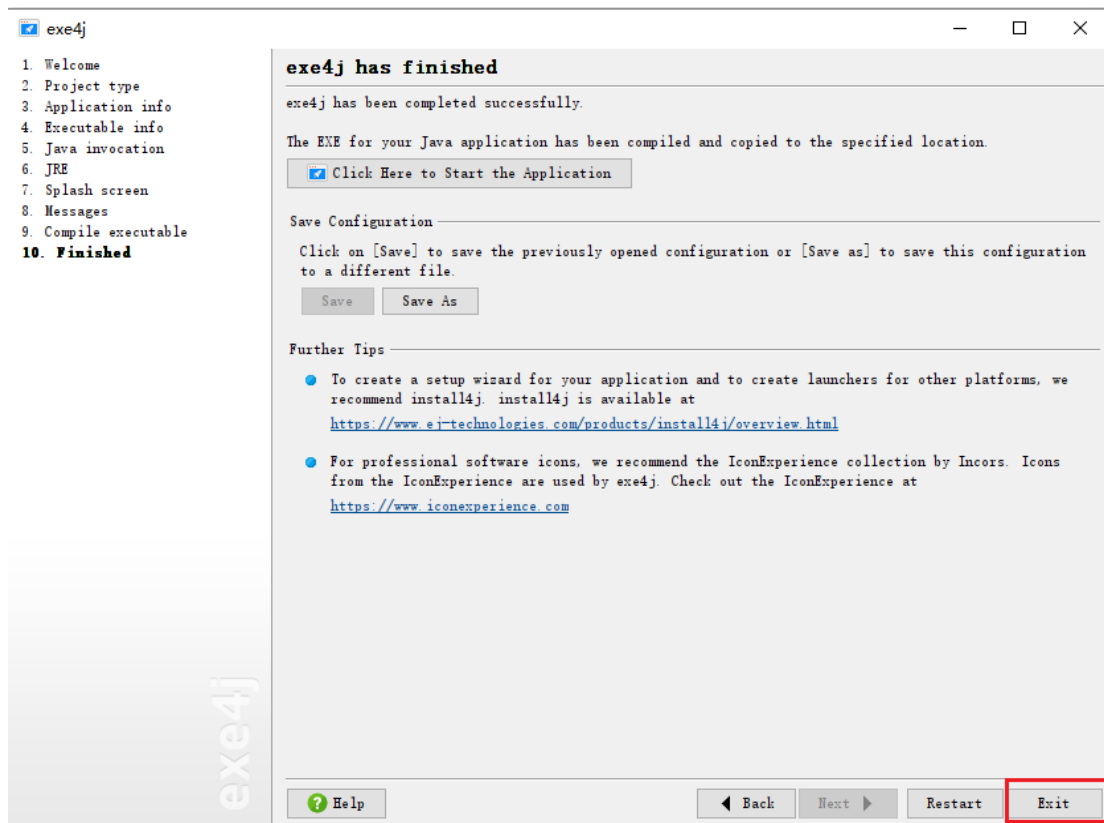


23 , 选择 Client VM

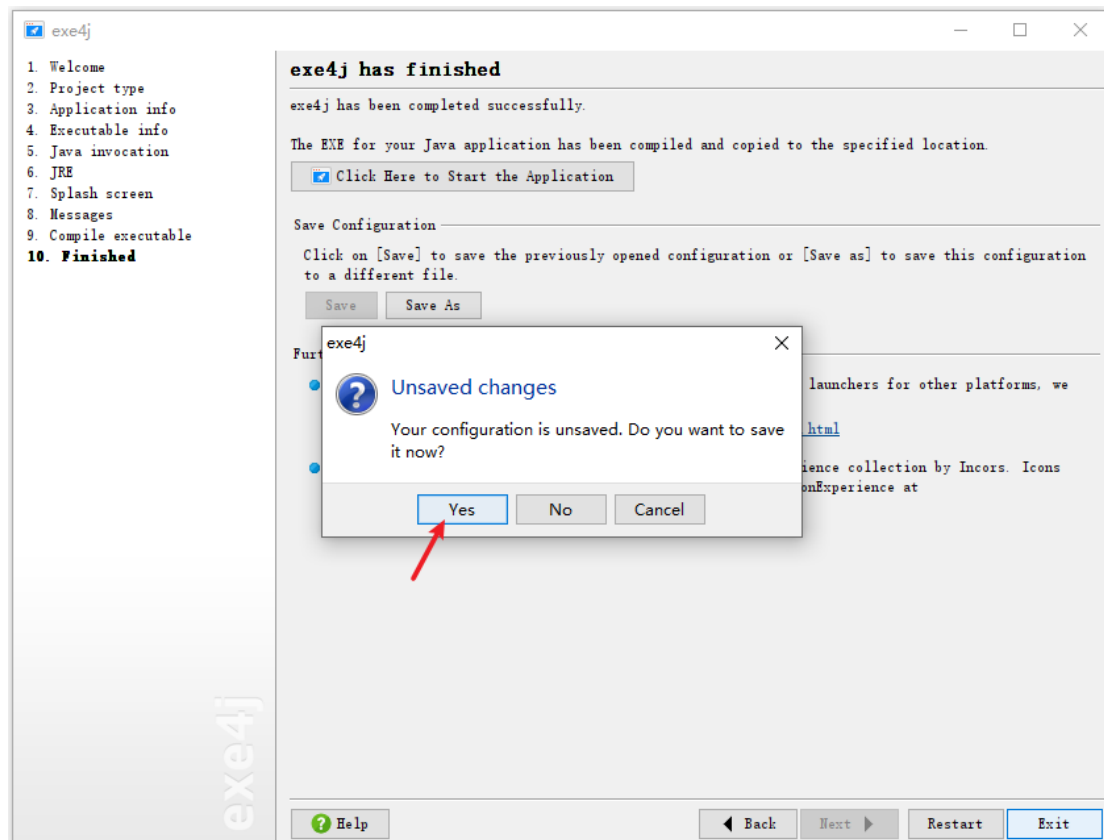
点击右下角的 next



24，然后一直下一步，最终出现如下界面
点击右下角的 Exit 退出。



25，点击 Exit 后，会提示是否需要保存刚刚的配置信息，可以点击 Yes，并选择一个路径进行保存。



26，如果第三步选择的 exe 保存的路径是桌面，那么在桌面上就能看到生成的 puzzlegame.exe 文件了。

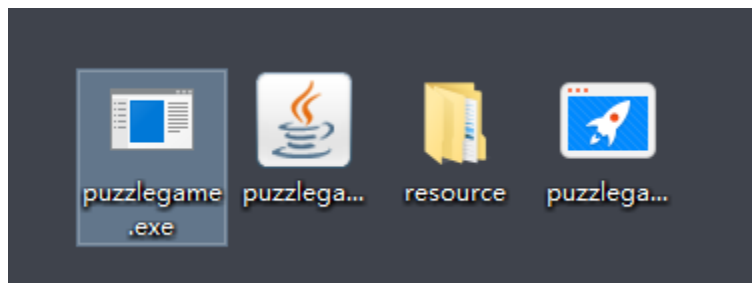
四个文件分别为：

左一：刚刚用 jar 生成的 exe 文件。

左二：idea 生成的 jar 包

左三：游戏用到的资源图片

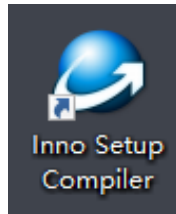
左四：刚刚用 exe4J 设置完毕之后保存的信息。



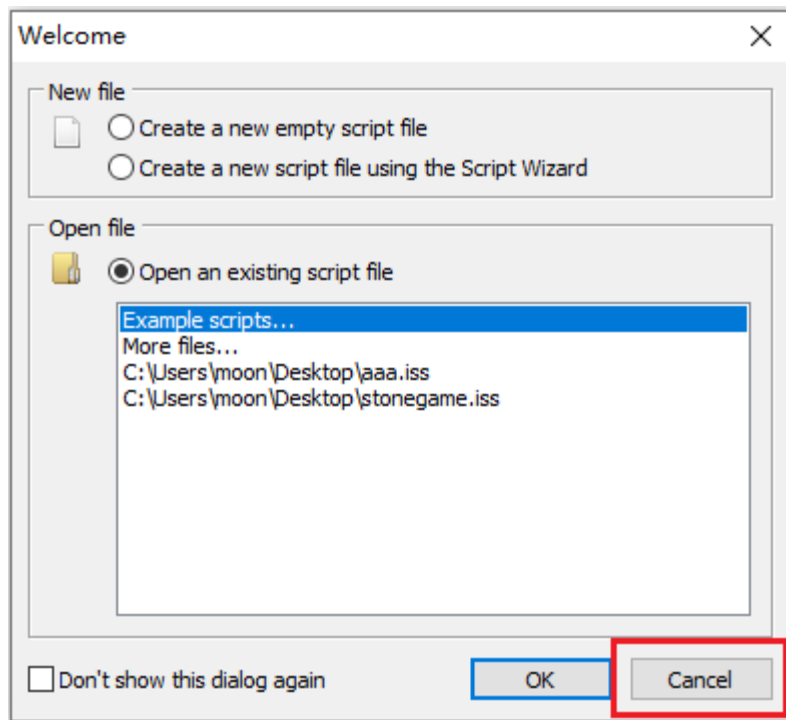
第四步：将jdk、资源文件、jar 包转换后的 exe 三者再次打包成最终的 exe。

刚刚，我们仅仅是把 java 代码变成了 exe。下面我们要把游戏中依赖的资源文件，也就是使用到的所有图片，还有 JDK 三者再次打包成最终的 exe，这样在没有jdk 电脑环境下也能运行。

1，打开 inno setup

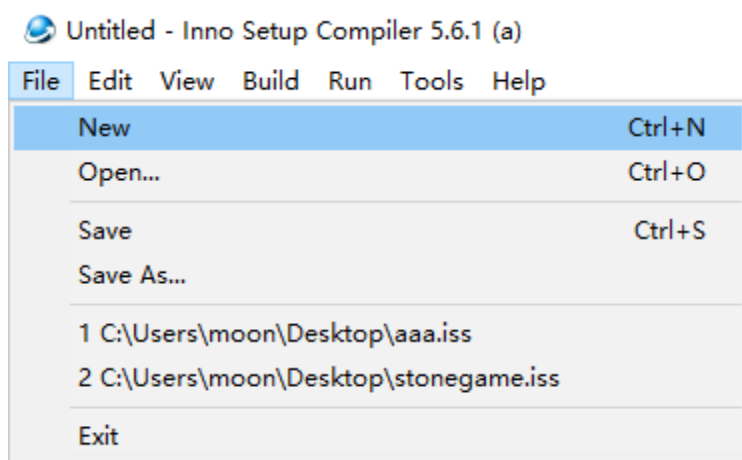


2, 在欢迎页面点击右下角的关闭

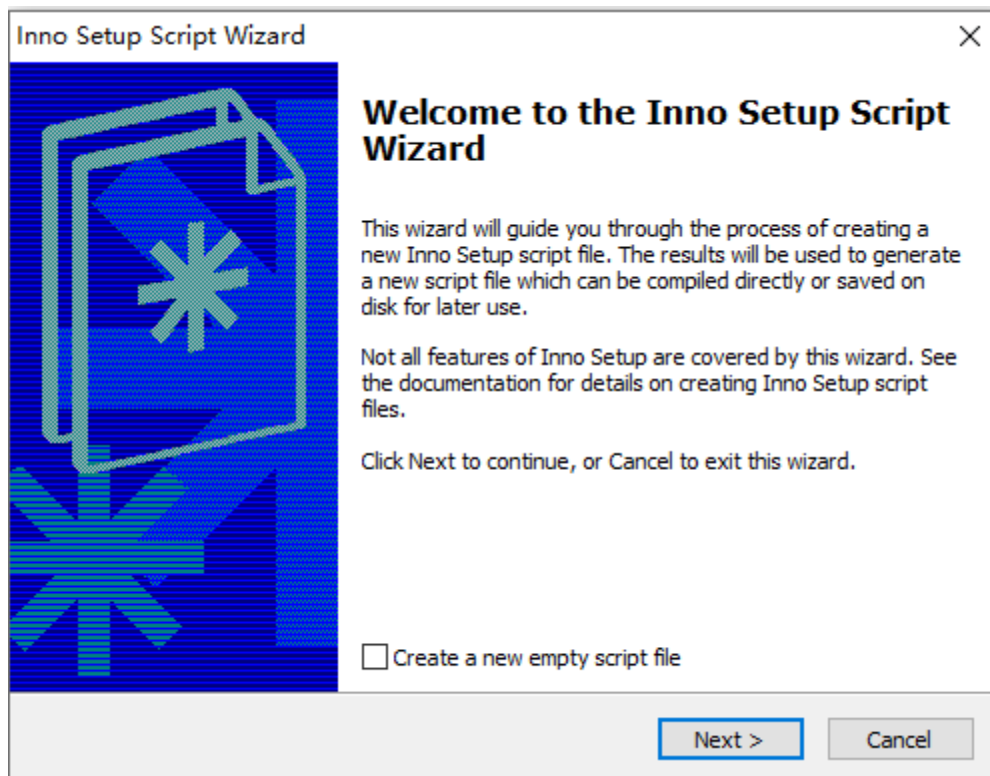


4, 点击左上角的 File

再点击 NEW

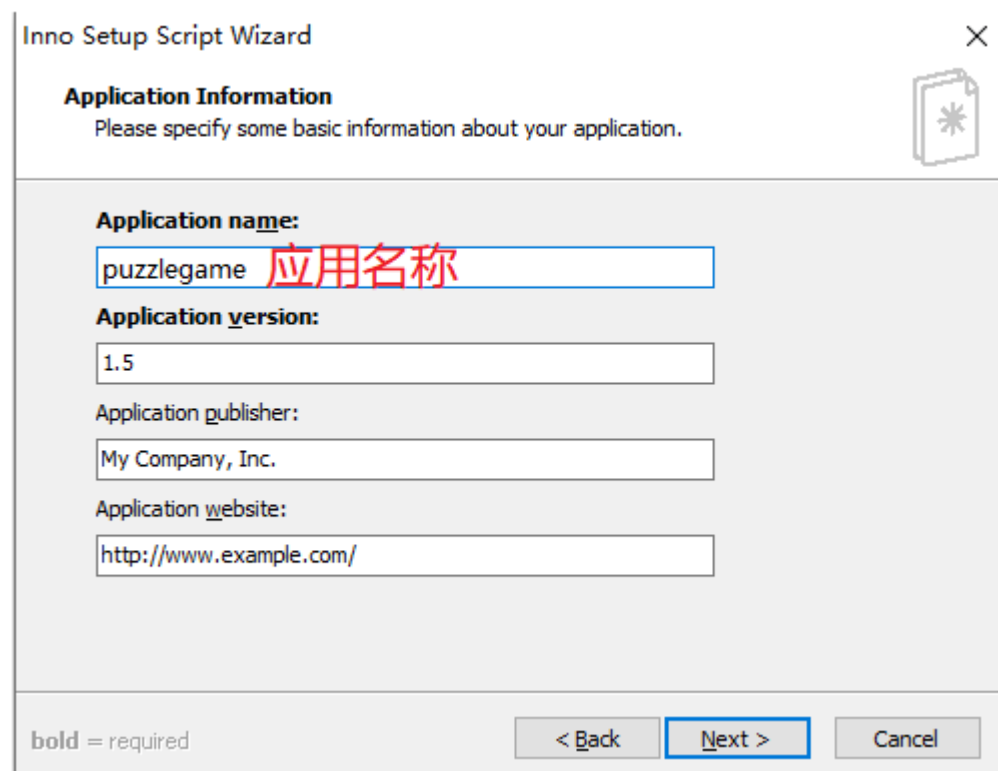


5, 点击 next



6 , 输入应用名称 puzzlegame

点击 next



7 , 不修改任何东西 , 直接点击 next

Inno Setup Script Wizard

Application Folder
Please specify folder information about your application.

Application destination base folder:
Program Files folder

Application folder name:
puzzlegame

☒ Allow user to change the application folder
Other:
☐ The application doesn't need a folder

bold = required

< Back Next > Cancel

8, 点击这里, 选择桌面上已经生成好的 puzzlegame.exe

Inno Setup Script Wizard

Application Files
Please specify the files that are part of your application.

Application main executable file:
C:\Users\moon\Desktop\puzzlegame.exe

☒ Allow user to start the application after Setup has finished
☐ The application doesn't have a main executable file

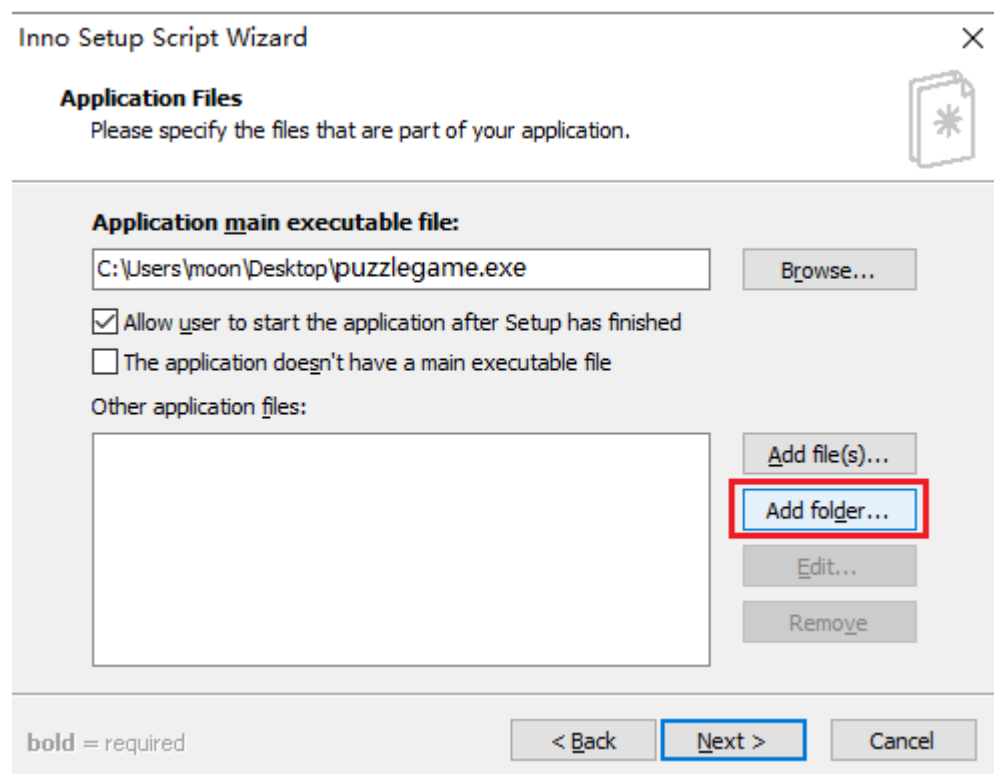
Other application files:

Add file(s)...
Add folder...
Edit...
Remove

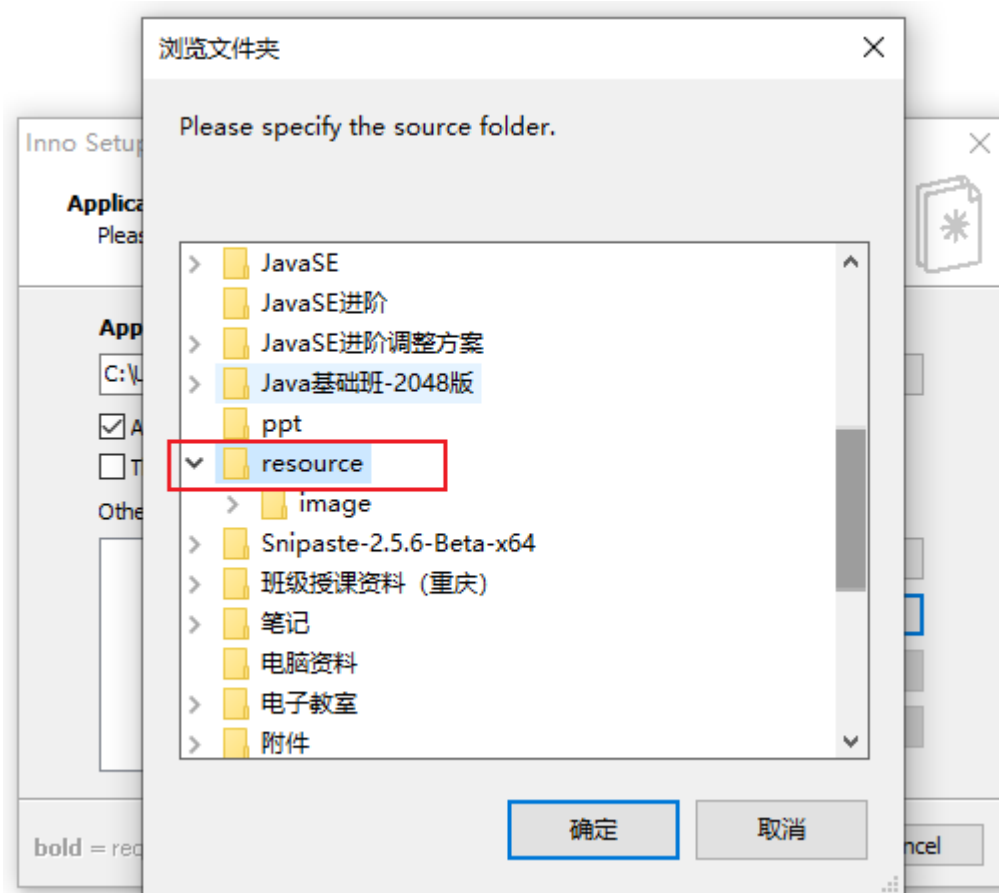
bold = required

< Back Next > Cancel

9, 点击 Add folder

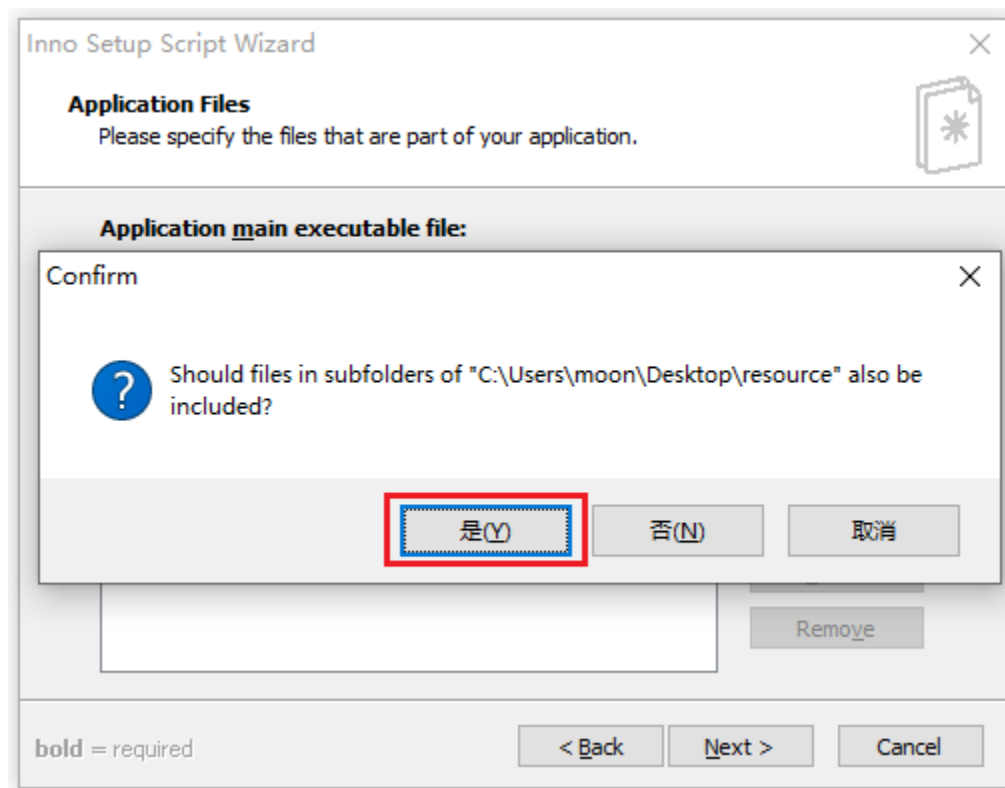


10 , 选择桌面的 resource , 再点击确定。

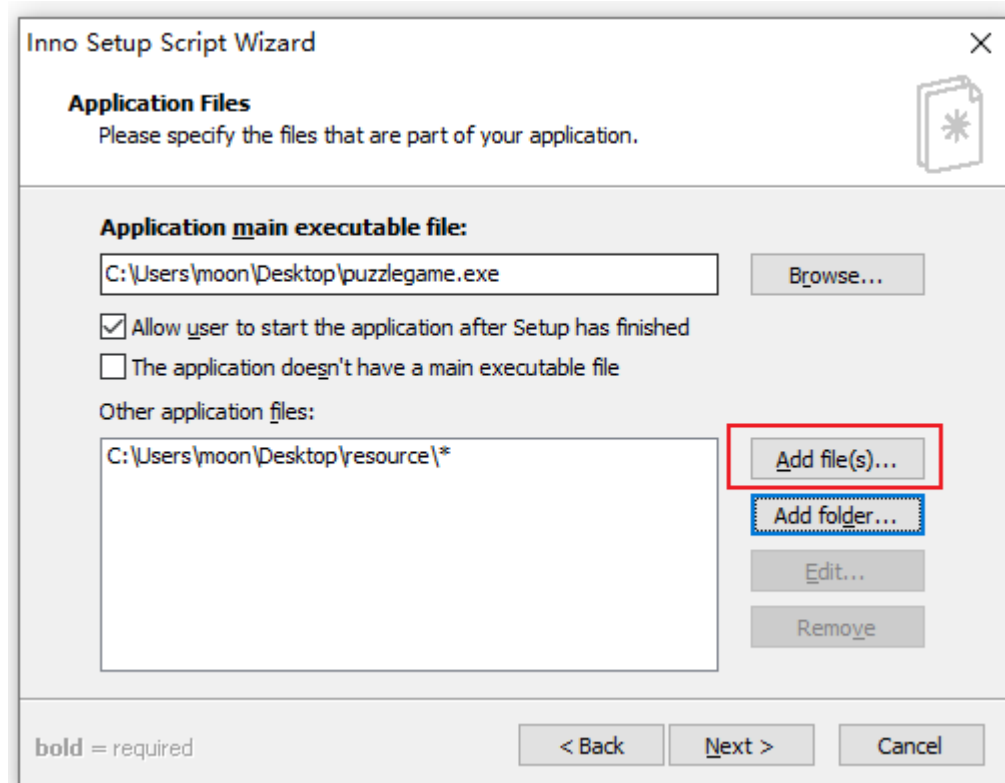


11 , 如果出现下面弹框 , 则点击是。

如果没有出现也没有任何关系。

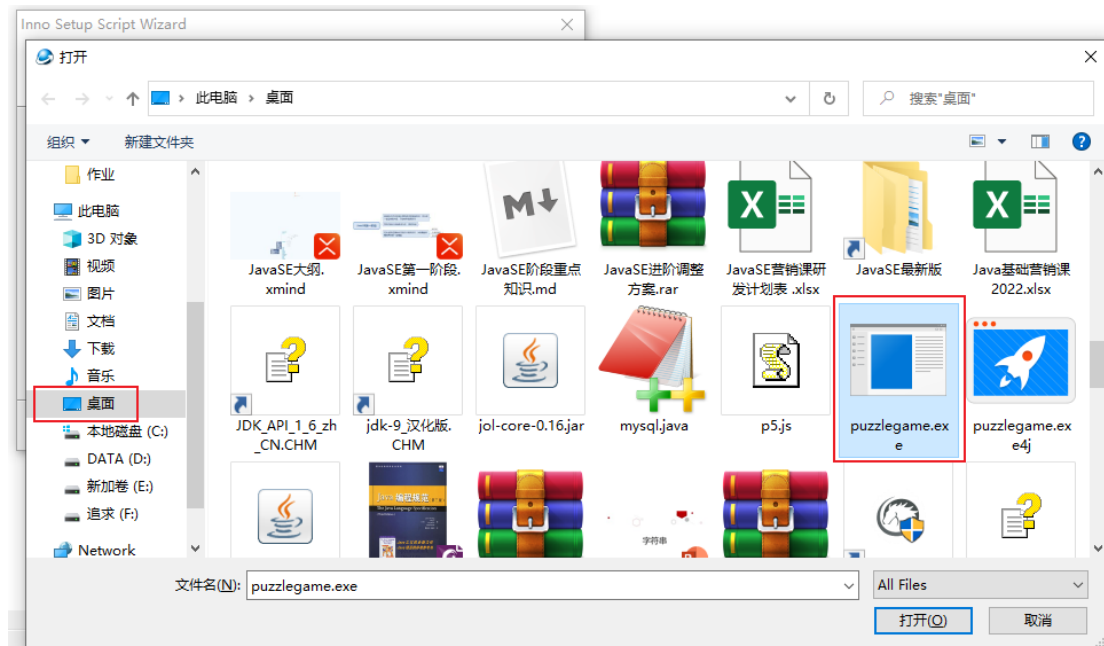


12 , 再点击 Add file(s)...



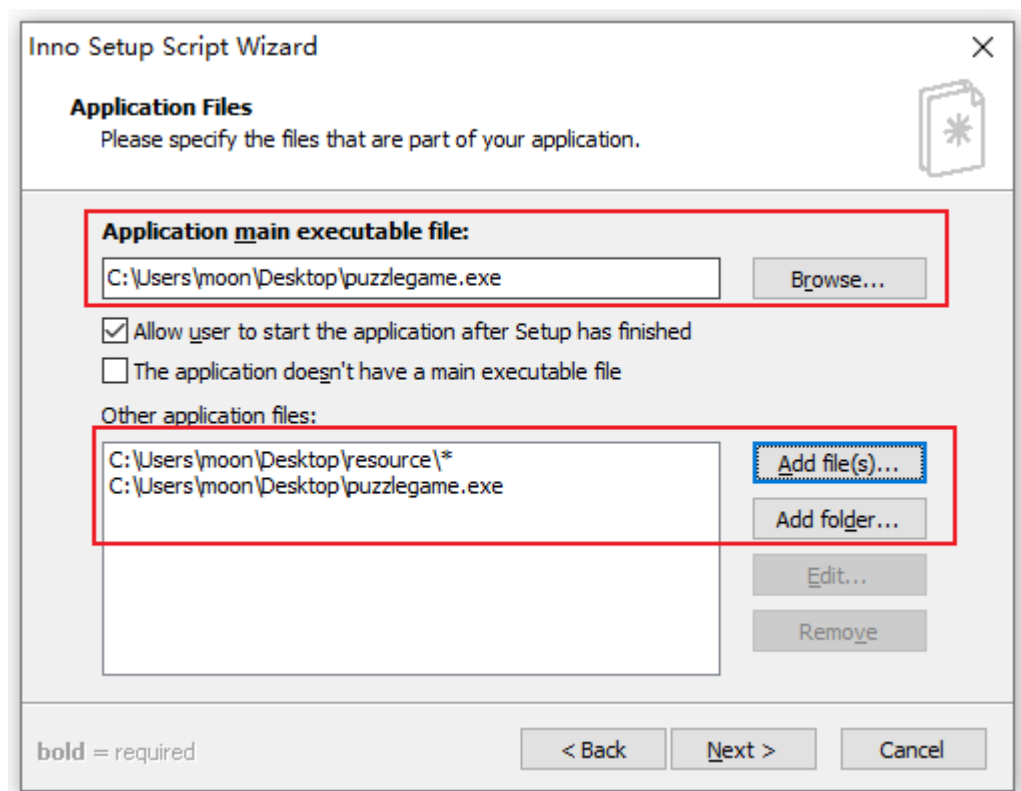
13，选中桌面的 puzzlegame.exe，再次添加一次。

点击下面的打开。

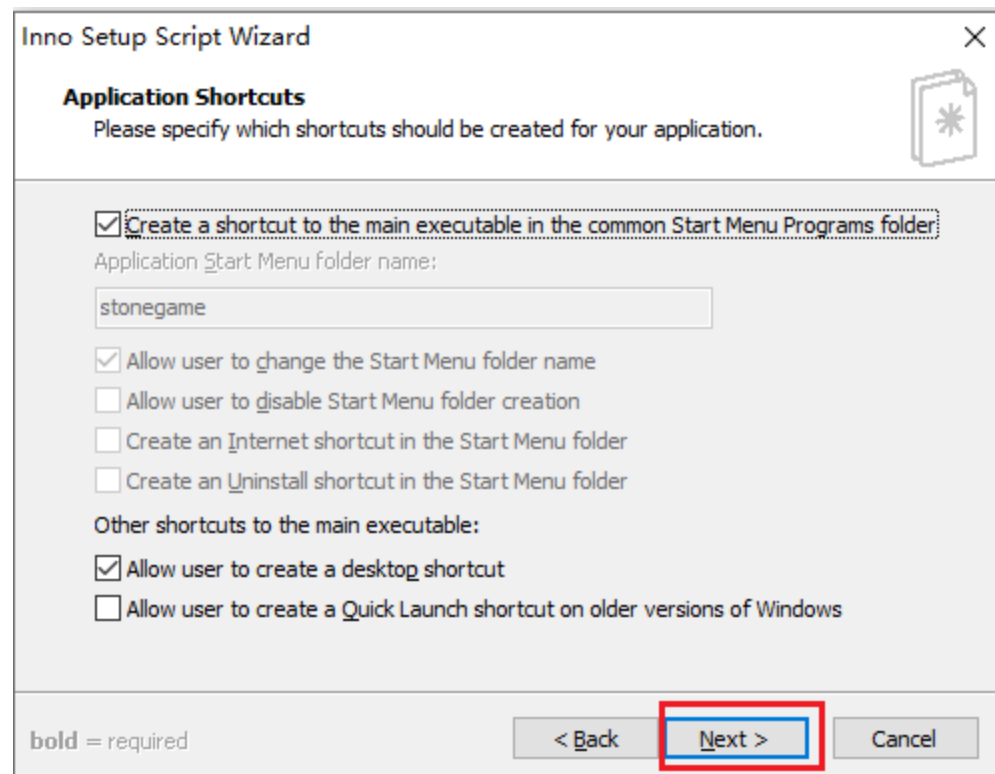


14，在本页面中一共设置了三处地方。

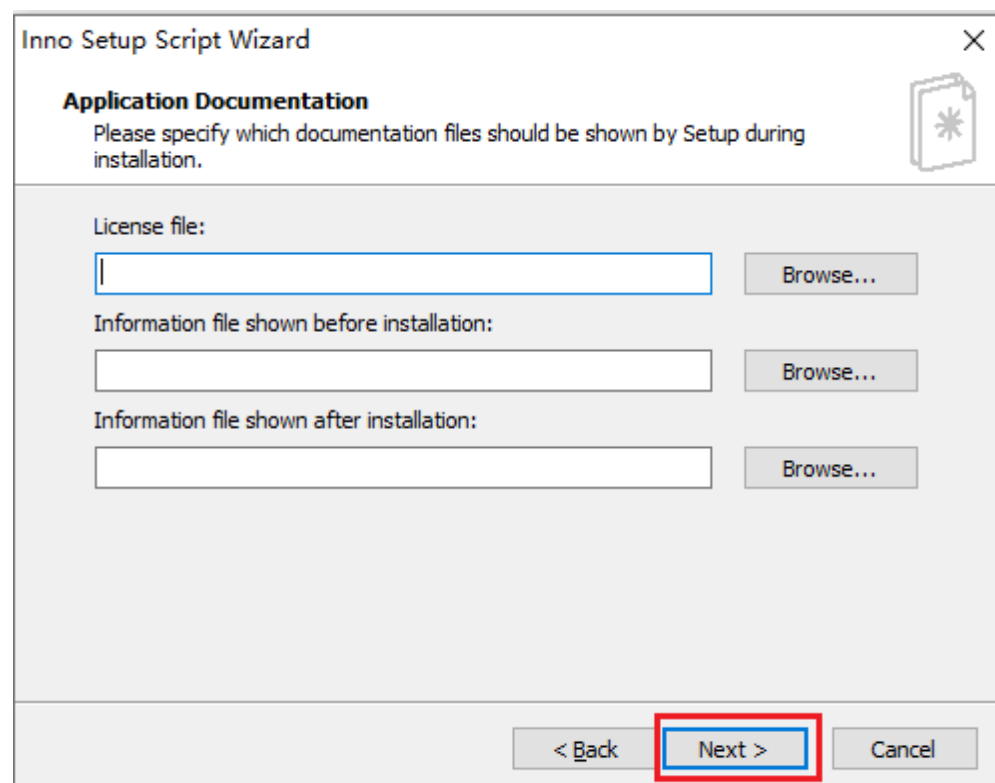
全部设置完毕，点击 next。



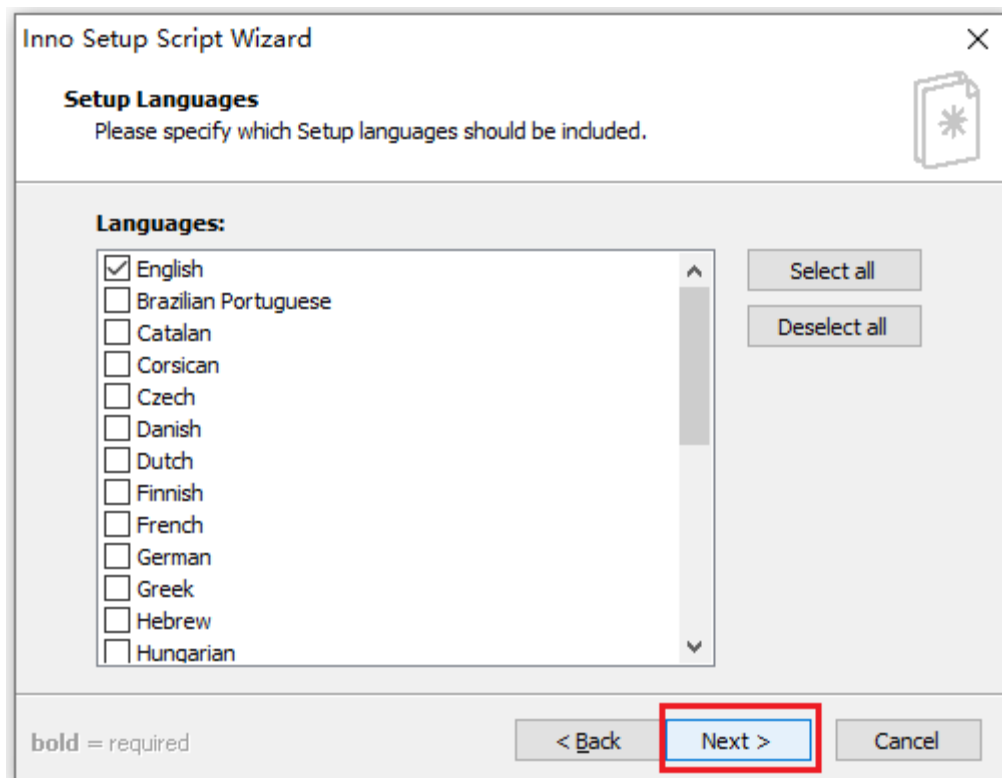
15，默认不用选择，点击 next



16，默认不用选择，点击 next



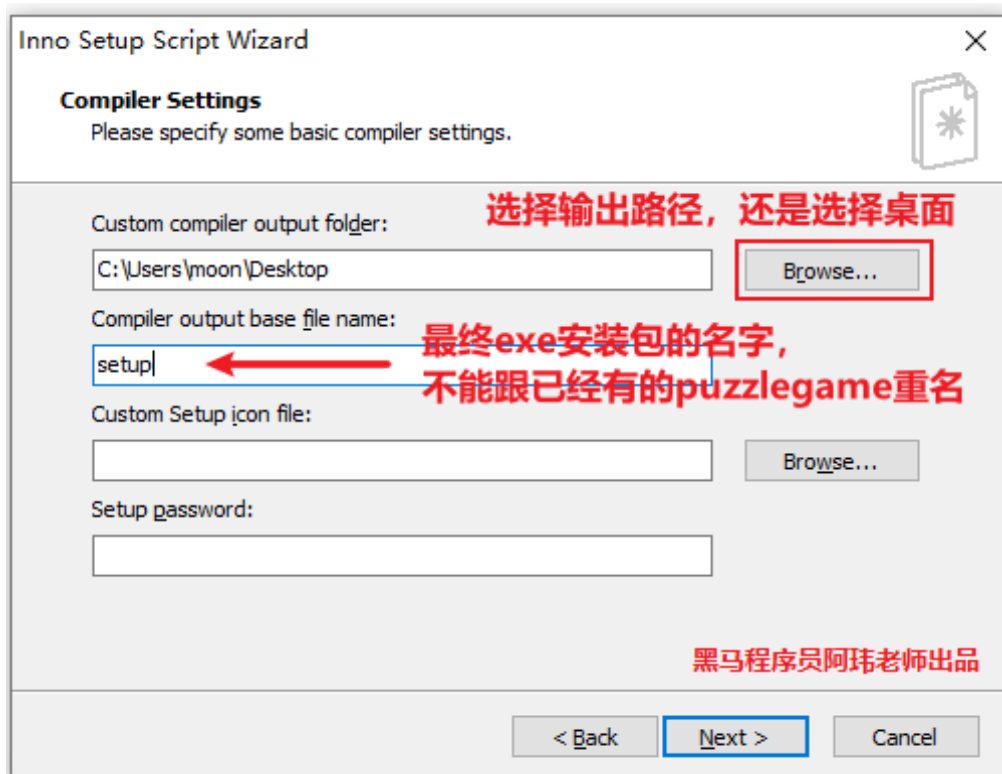
17，选择语言，还是默认，点击 next



18，选择输出路径，还是选择桌面。

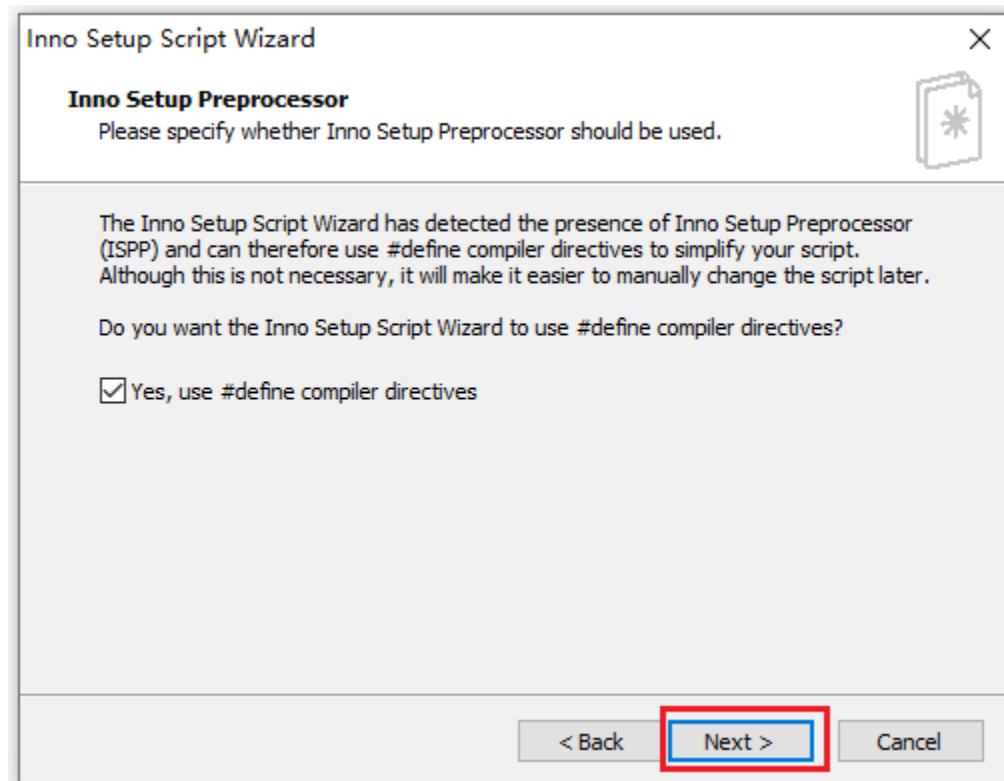
输入最终安装包的名字，不能跟已有的 puzzlegame 重名。

所以我写 setup，再点击右下角 next

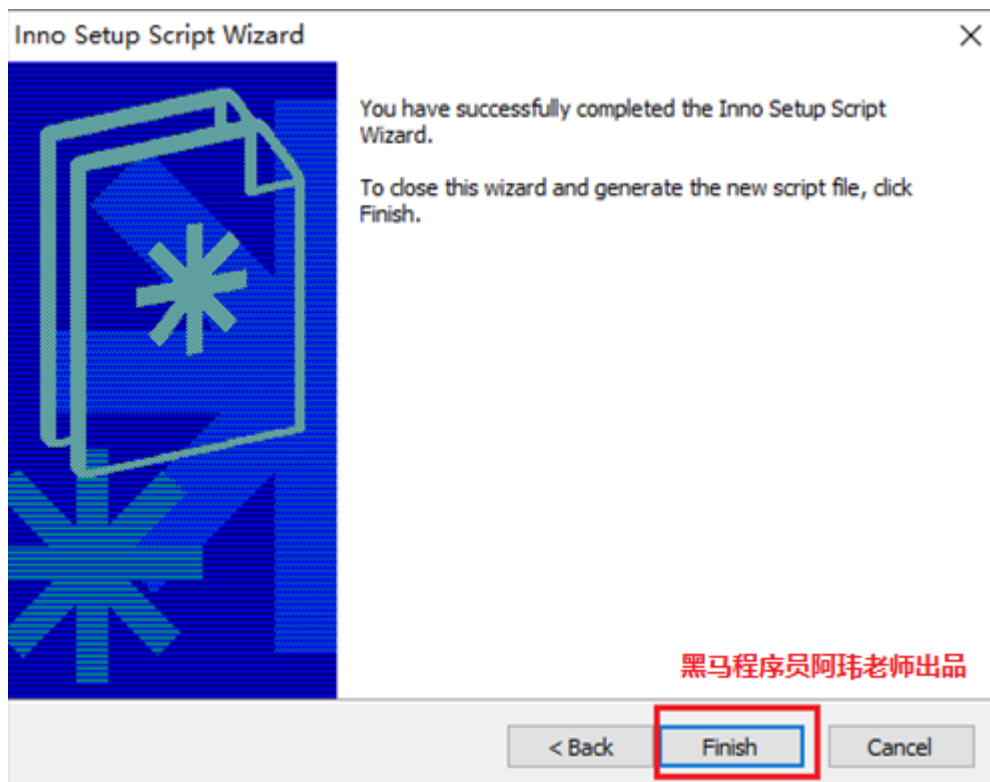


19，默认点击 next

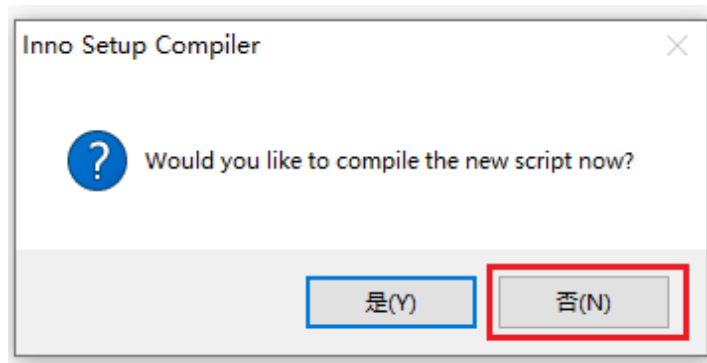
有部分同学电脑不显示这一步，也没有关系。



20，完成，点击 finish



21，配置到最后一步了，脚本文件，到这里会弹出问你是否马上编译，选择否，先把脚本写好再自己编译。



22，上面红色箭头处添加一行脚本。

#define MyJdkName "jdk"

添加前：

```
; Script generated by the Inno Setup Script Wizard.
; SEE THE DOCUMENTATION FOR DETAILS ON CREATING INNO SETUP SCRIPT FILES

#define MyAppName "puzzlegame"
#define MyAppVersion "1.5"
#define MyAppPublisher "My Company, Inc."
#define MyAppURL "http://www.example.com/"
#define MyAppExeName "puzzlegame.exe"

← 该位置添加一行脚本

[Setup]
; NOTE: The value of AppId uniquely identifies this application. Do not use the same AppId value in installers for other applications. (To generate a new GUID, click Tools | Generate GUID inside the IDE.)
AppId={{DCF07A86-622E-4035-AF65-E224E3A7A223}}
AppName={#MyAppName}
AppVersion={#MyAppVersion}
; AppVerName={#MyAppName} {#MyAppVersion}
AppPublisher={#MyAppPublisher}
```

23，添加完之后，如下图所示

```
; Script generated by the Inno Setup Script Wizard.
; SEE THE DOCUMENTATION FOR DETAILS ON CREATING INNO SETUP SCRIPT FILES!

#define MyAppName "puzzlegame"
#define MyAppVersion "1.5"
#define MyAppPublisher "My Company, Inc."
#define MyAppURL "http://www.example.com/"
#define MyAppExeName "puzzlegame.exe"
#define MyJdkName "jdk"

[Setup]
; NOTE: The value of AppId uniquely identifies this application.
; Do not use the same AppId value in installers for other applications.
; (To generate a new GUID, click Tools | Generate GUID inside the IDE.)
AppId={{DCF07A86-622E-4035-AF65-E224E3A7A223}
AppName={#MyAppName}
AppVersion={#MyAppVersion}
```

24，往下拉，把有红色框起来的这一行删掉

```
OutputBaseFilename=setup
Compression=lzma
SolidCompression=yes

[Languages]
Name: "english"; MessagesFile: "compiler:Default.isl"

[Tasks]
Name: "desktopicon"; Description: "{cm:CreateDesktopIcon}"; GroupDescription: "{cm:AdditionalIcons}"; Flags: unchecked

[Files]
Source: "C:\Users\moon\Desktop\puzzlegame.exe"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\Users\moon\Desktop\resource\*"; DestDir: "{app}"; Flags: ignoreversion recursesubdirs createallsubdirs
Source: "C:\Users\moon\Desktop\puzzlegame.exe"; DestDir: "{app}"; Flags: ignoreversion
; NOTE: Don't use "Flags: ignoreversion" on any shared system files

[Icons]
Name: "{commonprograms}\{#MyAppName}"; Filename: "{app}\{#MyAppExeName}"
Name: "{commondesktop}\{#MyAppName}"; Filename: "{app}\{#MyAppExeName}"; Tasks: desktopicon

[Run]
Filename: "{app}\{#MyAppExeName}"; Description: "{cm:LaunchProgram,{StringChange(MyAppName, '&', '&&')}}"; Flags: nowait
```

25，在上一步删除位置添加一段行的文字

Source: "自己本地 JDK 路径*"; DestDir:
 "{app}\{#MyJdkName}"; Flags: ignoreversion
 recursesubdirs createallsubdirs

千万注意：这段文字不要直接复制。根据自己电脑中 JDK 的安装位置进行修改，添加完毕之后如下图

```
OutputBaseFilename=setup
Compression=lzma
SolidCompression=yes

[Languages]
Name: "english"; MessagesFile: "compiler:Default.isl"

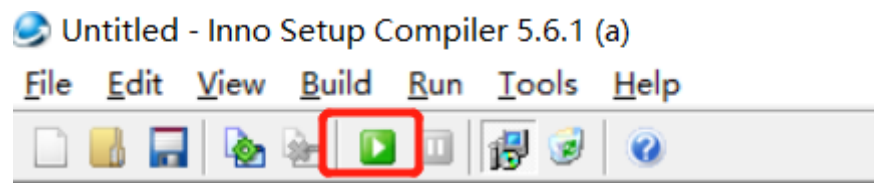
[Tasks]
Name: "desktopicon"; Description: "{cm:CreateDesktopIcon}"; GroupDescription: "{cm:AdditionalIcons}"; Flags: unchecked

[Files]
Source: "C:\Users\moon\Desktop\puzzlegame.exe"; DestDir: "{app}"; Flags: ignoreversion
Source: "C:\Users\moon\Desktop\resource\*"; DestDir: "{app}"; Flags: ignoreversion recursesubdirs createallsubdirs
Source: "F:\develop\jdk\*"; DestDir: "{app}\{#MyJdkName}"; Flags: ignoreversion recursesubdirs createallsubdirs
; NOTE: Don't use "Flags: ignoreversion" on any shared system files

[Icons]
Name: "{commonprograms}\{#MyAppName}"; Filename: "{app}\{#MyAppExeName}"
Name: "{commondesktop}\{#MyAppName}"; Filename: "{app}\{#MyAppExeName}"; Tasks: desktopicon

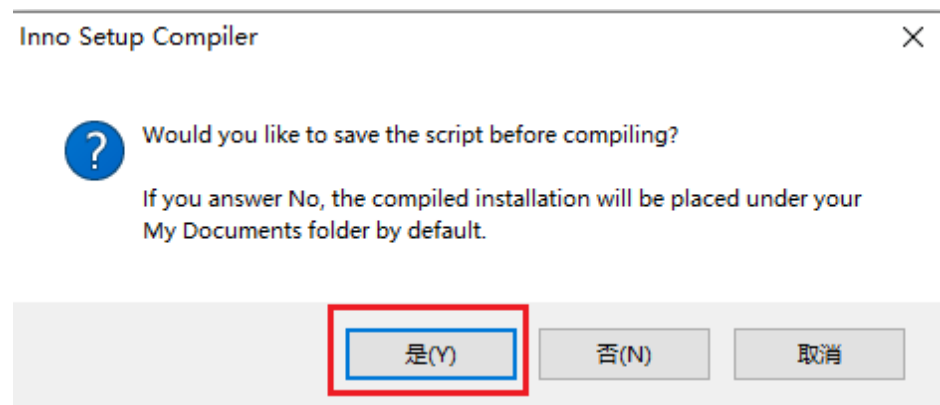
[Run]
Filename: "{app}\{#MyAppExeName}"; Description: "{cm:LaunchProgram,{StringChange(MyAppName, '&', '&&')}}"; Flags: nowait postinstall skipifsilent
```

26，点击上方的绿色按钮开始编译。

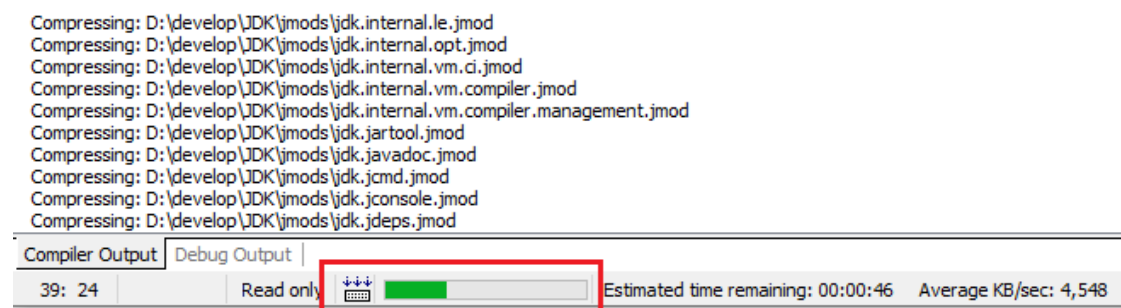


27，此时会询问，是否需要保存。

可以点击是，选择一个位置保存一下刚刚修改之后的结果。



28，然后等待绿色滚动条结束



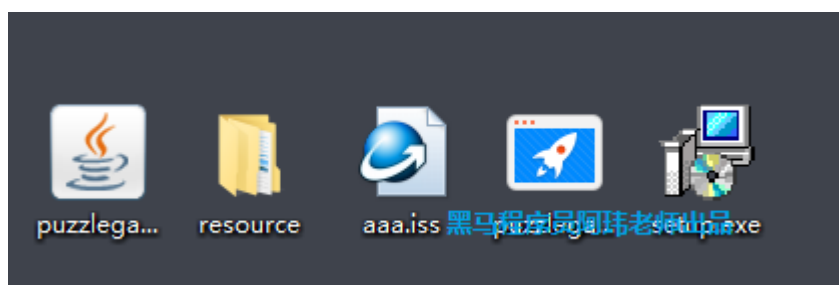
29，当绿色滚动条结束后，会自动安装 setup.exe 文件。

此时可以点击否，先不安装。

在桌面上，会多了一个 setup.exe 文件和一个后缀名为 iss 的文件

setup.exe：打包成功的游戏安装包。

iss 文件：就是刚刚设置的脚本文件。



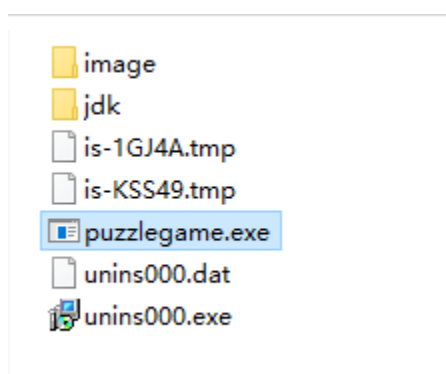
30，现在就可以把这个 exe 文件发给你的好基友了，他的电脑上不需要安装 JDK，直接双击这个安装包就可以玩游戏了。

在安装的时候可以选择安装路径。

还可以在桌面生成快捷方式。

注意点：

1， 安装完毕之后，可以到安装目录去找 puzzlegame.exe 双击就可以玩游戏了。刚刚桌面上的 5 个文件可以全部删除。以后双击 puzzlegame.exe 就可以玩游戏了。



2， 如果游戏运行之后，有效果但是无图片，或者部分图片无法加载。如下图所示：此时背景图片无法加载。



3, 修改步骤一：

检查安装目录下 image 文件夹中是否有该图片。如果没有将缺少的图片粘贴进去重新运行游戏。(黑马程序员阿玮老师出品)

修改步骤二：

检查代码，图片的路径需要把模块名删掉。

```
public void initView() {  
    //1. 添加用户名文字  
    JLabel usernameText = new JLabel(new ImageIcon( filename: "image\\login\\用户名.png"));  
    usernameText.setBounds( x: 116, y: 135, width: 47, height: 17);  
    this.getContentPane().add(usernameText);  
  
    //2. 添加用户名输入框  
  
    username.setBounds( x: 195, y: 134, width: 200, height: 30);  
    this.getContentPane().add(username);  
  
    //3. 添加密码文字  
    JLabel passwordText = new JLabel(new ImageIcon( filename: "image\\login\\密码.png"));  
    passwordText.setBounds( x: 130, y: 195, width: 32, height: 16);  
    this.getContentPane().add(passwordText);  
}
```

4, 代码修改后, 需要重新生成 exe 安装包, 重新安装游戏。