Windows 配置

一、目录

二、Maven

1、下载

- 官网：

  Maven – Download Apache Maven

	Link	Checksums	Signature
Binary tar.gz archive	apache-maven-3.8.5-bin.tar.gz	apache-maven-3.8.5-bin.tar.gz.sha512	apache-maven-3.8.5-bin.tar.gz.asc
Binary zip archive	apache-maven-3.8.5-bin.zip	apache-maven-3.8.5-bin.zip.sha512	apache-maven-3.8.5-bin.zip.asc
Source tar.gz archive	apache-maven-3.8.5-src.tar.gz	apache-maven-3.8.5-src.tar.gz.sha512	apache-maven-3.8.5-src.tar.gz.asc
Source zip archive	apache-maven-3.8.5-src.zip	apache-maven-3.8.5-src.zip.sha512	apache-maven-3.8.5-src.zip.asc

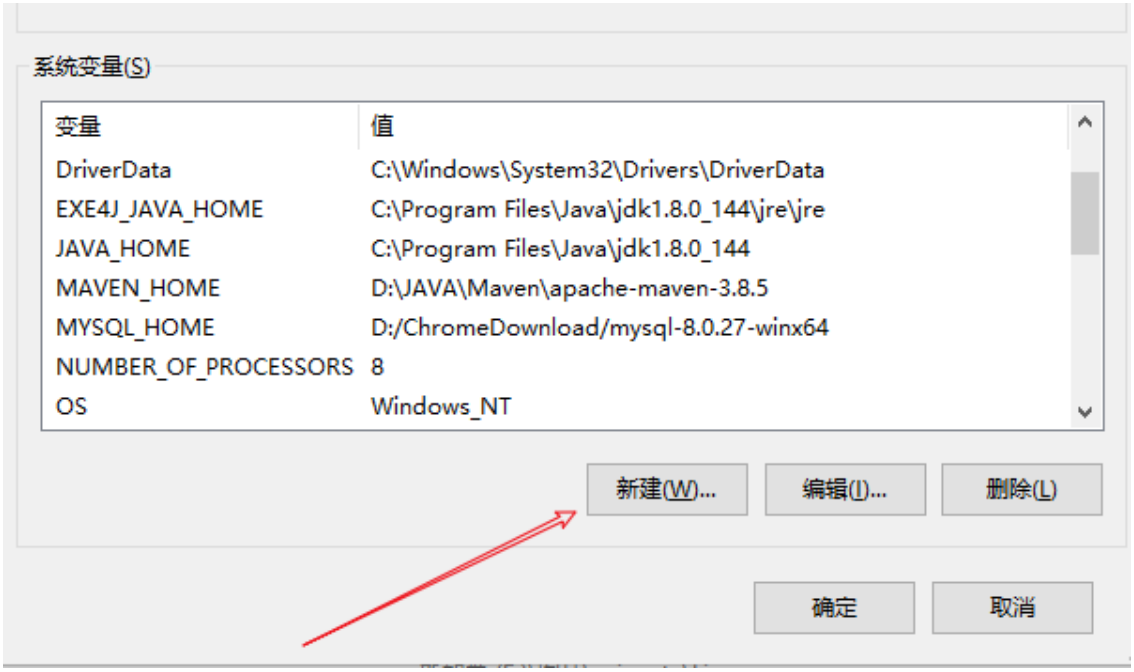
- Release Notes

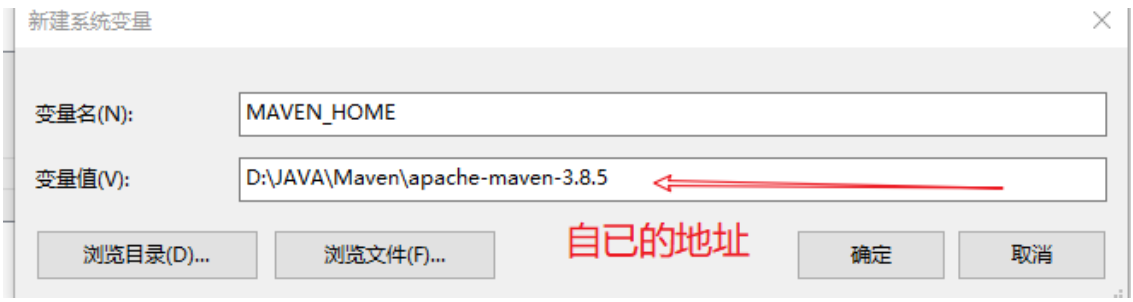
- 百度网盘：

  链接：https://pan.baidu.com/s/15glszTScDSDVVOfmFVZKrg
  提取码：ed6n
  --来自百度网盘超级会员V1的分享

2、配置环境

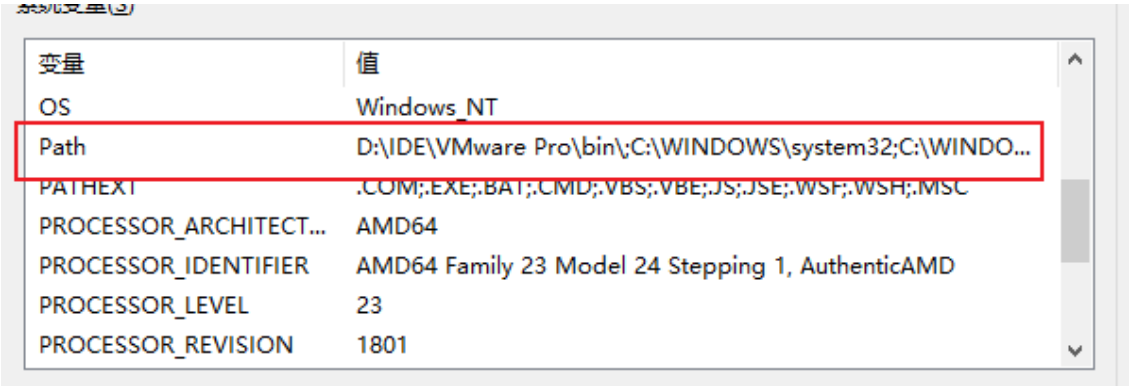
1. 打开编辑环境变量（控制面板搜索环境变量）
2. 系统变量新建





```
MAVEN_HOME
```

3. path变量



点击编辑（/双击），点击增加

写入

```
%MAVEN_HOME%\bin
```

4. 环境变量成功，检查

管理员身份运行cmd，输入 **mvn -v**

```
C:\Users\MateBook13>mvn -v
Apache Maven 3.8.5 (3599d3414f046de2324203b78ddcf9b5e4388aa0)
Maven home: D:\JAVA\Maven\apache-maven-3.8.5
Java version: 1.8.0_144, vendor: Oracle Corporation, runtime: C:\Program Files\Java\jdk1.8.0_144\jre
Default locale: zh_CN, platform encoding: GBK
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"

C:\Users\MateBook13>
```

出现以上内容，创建成功

3、修改配置文件

1. 打开maven解压目录下的conf中的**settings.xml**（记事本或者编辑器打开）

```
<mirrors>
  <!-- mirror
   | Specifies a repository mirror site to use instead of a given repository. The repository that
   | this mirror serves has an ID that matches the mirrorOf element of this mirror. IDs are used
   | for inheritance and direct lookup purposes, and must be unique across the set of mirrors.
   |
  <mirror>
    <id>mirrorId</id>
    <mirrorOf>repositoryId</mirrorOf>
    <name>Human Readable Name for this Mirror.</name>
    <url>http://my.repository.com/repo/path</url>
  </mirror>
   -->
  <mirror>
    <id>maven-default-http-blocker</id>
    <mirrorOf>external:http:*</mirrorOf>
    <name>Pseudo repository to mirror external repositories initially using HTTP.</name>
    <url>http://0.0.0.0/</url>
    <blocked>true</blocked>
  </mirror>
  <!--阿里云仓库-->
  <mirror>
   <id>nexus-aliyun</id>
   <mirrorOf>*</mirrorOf>
   <name>Nexus aliyun</name>
   <url>http://maven.aliyun.com/nexus/content/groups/public</url>
</mirror>

</mirrors>
```

找到

配置阿里云镜像仓库，国外仓库下载速度慢，体验不好。

```
<!--阿里云仓库-->
<mirror>
      <id>nexus-aliyun</id>
      <mirrorOf>*</mirrorOf>
      <name>Nexus aliyun</name>
      <url>http://maven.aliyun.com/nexus/content/groups/public</url>
</mirror>
```

！！！！注意格式和位置不要写进注释里。

2. 配置本地仓库的地址（所有的包都下载在这里）

> maven默认地址为C盘的 用户目录下/.m2/repository
>
> 建议修改，保护C盘

```
  | Default: ${user.home}/.m2/repository
<localRepository>/path/to/local/repo</localRepository>
-->
 <localRepository>D:\JAVA\Maven\.m2\repository</localRepository>
```

再次打开**settings.xml**文件

```
  | The sections in this sample file are intended to give you a running start at
  | getting the most out of your Maven installation. Where appropriate, the default
  | values (values used when the setting is not specified) are provided.
  |
  |-->
<settings xmlns="http://maven.apache.org/SETTINGS/1.2.0"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.2.0 https://maven.apache.or
  <!-- localRepository
   | The path to the local repository maven will use to store artifacts.
   |
   | Default: ${user.home}/.m2/repository
  <localRepository>/path/to/local/repo</localRepository>
  -->
   <localRepository>D:\JAVA\Maven\.m2\repository</localRepository>
```

找到如图所示的位置,写入下面的内容，注意（**内容里的文件路径是你的本地仓库的路径，自行创建修改**）

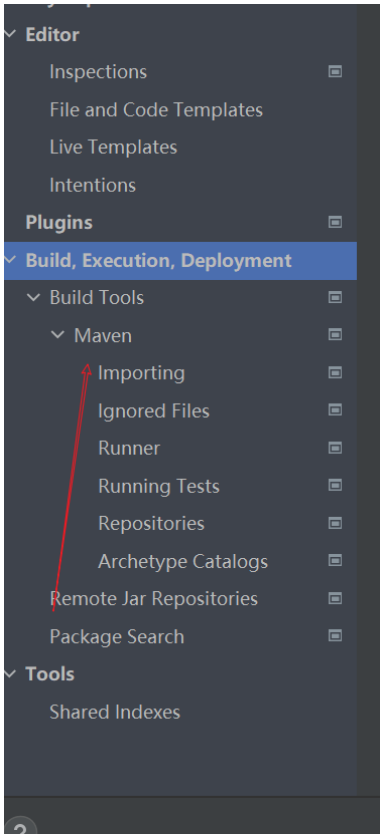
！！！！注意格式和位置不要写进注释里。

```
<localRepository>D:\JAVA\Maven\.m2\repository</localRepository>
```

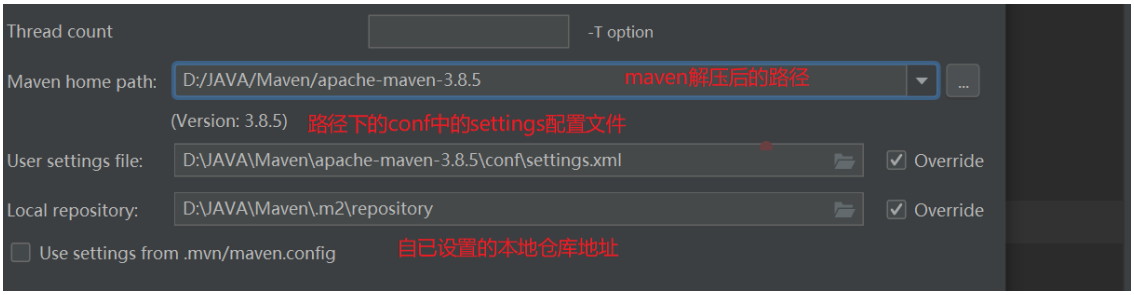
4、idea配置Maven信息

1. File->Settings->Build……->Build Tools->Maven

   (File->Settings->搜索Maven)



2.



按照图片所示配置路径

三个路径都要修改为自己设置的路径

！！！成功

三、Gradle

四、Node.js

> 官网：
>
> http://nodejs.cn/download/

下载

长期支持版本: **16.15.0**

Windows 安装包 (.msi)	32 位	64 位
Windows 二进制文件 (.zip)	32 位	64 位
macOS 安装包 (.pkg)	64 位 / ARM64	
macOS 二进制文件 (.tar.gz)	64 位	ARM64
Linux 二进制文件 (x64)	64 位	
Linux 二进制文件 (ARM)	ARMv7	ARMv8
Docker 镜像	官方镜像	
全部安装包	全部安装包	

检查

```
C:\Users\MateBook13>node -v
v16.13.2

C:\Users\MateBook13>npm version
{
  npm: '8.1.2',
  node: '16.13.2',
  v8: '9.4.146.24-node.14',
  uv: '1.42.0',
  zlib: '1.2.11',
  brotli: '1.0.9',
  ares: '1.18.1',
  modules: '93',
  nghttp2: '1.45.1',
  napi: '8',
  llhttp: '6.0.4',
  openssl: '1.1.1l+quic',
  cldr: '39.0',
  icu: '69.1',
  tz: '2021a',
  unicode: '13.0',
  ngtcp2: '0.1.0-DEV',
  nghttp3: '0.1.0-DEV'
}

C:\Users\MateBook13>npx -v
8.1.2
```

五、MinGW

1、下载

> 官网：
>
> https://sourceforge.net/projects/mingw-w64/files/

- MinGW-W64-install.exe

MinGW-W64 GCC-8.1.0

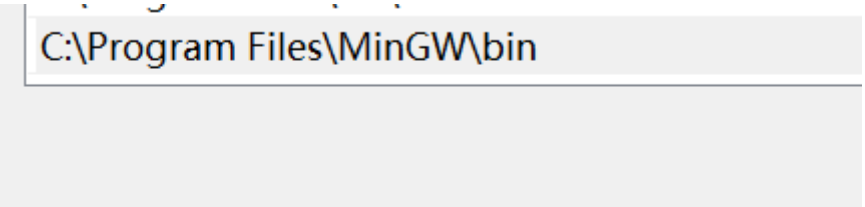
- x86_64-posix-sjlj
- x86_64-posix-seh
- x86_64-win32-sjlj
- x86_64-win32-seh
- i686-posix-sjlj
- i686-posix-dwarf
- i686-win32-sjlj
- i686-win32-dwarf

2、安装

解压

bin	2022/6/4 11:32	文件夹	
etc	2022/6/4 11:31	文件夹	
include	2022/6/4 11:31	文件夹	
lib	2022/6/4 11:31	文件夹	
libexec	2022/6/4 11:31	文件夹	
licenses	2022/6/4 11:31	文件夹	
opt	2022/6/4 11:31	文件夹	
share	2022/6/4 11:31	文件夹	
x86_64-w64-mingw32	2022/6/4 11:31	文件夹	
build-info.txt	2018/5/12 0:00	文本文档	52 KB

3、环境变量



系统变量-->path-->新建-->解压后的bin目录

4、验证

```
gcc --version
g++ --version
```

```
C:\Program Files\MinGW\bin>gcc -v
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=C:/Program\ Files/MinGW/bin/../libexec/gcc/x86_64-w64-mingw32/8.1.0/lto-wrapper.exe
Target: x86_64-w64-mingw32
Configured with: ../../../src/gcc-8.1.0/configure --host=x86_64-w64-mingw32 --build=x86_64-w64-mingw32 --target=x86_64-w
64-mingw32 --prefix=/mingw64 --with-sysroot=/c/mingw810/x86_64-810-posix-sjlj-rt_v6-rev0/mingw64 --enable-shared --enabl
e-static --enable-targets=all --enable-multilib --enable-languages=c,c++,fortran,lto --enable-libstdcxx-time=yes --enabl
e-threads=posix --enable-libgomp --enable-libatomic --enable-lto --enable-graphite --enable-checking=release --enable-fu
lly-dynamic-string --enable-version-specific-runtime-libs --enable-sjlj-exceptions --disable-libstdcxx-pch --disable-lib
stdcxx-debug --enable-bootstrap --disable-rpath --disable-win32-registry --disable-nls --disable-werror --disable-symver
s --with-gnu-as --with-gnu-ld --with-arch-32=i686 --with-arch-64=nocona --with-tune-32=generic --with-tune-64=core2 --wi
th-libiconv --with-system-zlib --with-gmp=/c/mingw810/prerequisites/x86_64-w64-mingw32-static --with-mpfr=/c/mingw810/pr
erequisites/x86_64-w64-mingw32-static --with-mpc=/c/mingw810/prerequisites/x86_64-w64-mingw32-static --with-isl=/c/mingw
810/prerequisites/x86_64-w64-mingw32-static --with-pkgversion='x86_64-posix-sjlj-rev0, Built by MinGW-W64 project' --wit
h-bugurl=https://sourceforge.net/projects/mingw-w64 CFLAGS='-O2 -pipe -fno-ident -I/c/mingw810/x86_64-810-posix-sjlj-rt_
v6-rev0/mingw64/opt/include -I/c/mingw810/prerequisites/x86_64-zlib-static/include -I/c/mingw810/prerequisites/x86_64-w6
4-mingw32-static/include' CXXFLAGS='-O2 -pipe -fno-ident -I/c/mingw810/x86_64-810-posix-sjlj-rt_v6-rev0/mingw64/opt/incl
ude -I/c/mingw810/prerequisites/x86_64-zlib-static/include -I/c/mingw810/prerequisites/x86_64-w64-mingw32-static/include
 CPPFLAGS=' -I/c/mingw810/x86_64-810-posix-sjlj-rt_v6-rev0/mingw64/opt/include -I/c/mingw810/prerequisites/x86_64-zlib-
static/include -I/c/mingw810/prerequisites/x86_64-w64-mingw32-static/include' LDFLAGS='-pipe -fno-ident -L/c/mingw810/x8
6_64-810-posix-sjlj-rt_v6-rev0/mingw64/opt/lib -L/c/mingw810/prerequisites/x86_64-zlib-static/lib -L/c/mingw810/prerequi
sites/x86_64-w64-mingw32-static/lib '
Thread model: posix
gcc version 8.1.0 (x86_64-posix-sjlj-rev0, Built by MinGW-W64 project)
```

```
C:\Windows\system32>g++ --version
g++ (x86_64-posix-sjlj-rev0, Built by MinGW-W64 project) 8.1.0
Copyright (C) 2018 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

六、rust安装

1、下载

> 官网：https://www.rust-lang.org/zh-CN/tools/install

2、安装

> 提示：需要C/C++编译器 `MSVC` 或者 `MinGW`

1、打开安装包

```
The Cargo home directory is located at:

  C:\Users\MateBook13\.cargo

This can be modified with the CARGO_HOME environment variable.

The cargo, rustc, rustup and other commands will be added to
Cargo's bin directory, located at:

  C:\Users\MateBook13\.cargo\bin

This path will then be added to your PATH environment variable by
modifying the HKEY_CURRENT_USER/Environment/PATH registry key.

You can uninstall at any time with rustup self uninstall and
these changes will be reverted.

Current installation options:


   default host triple: x86_64-pc-windows-msvc
     default toolchain: stable (default)
               profile: default
  modify PATH variable: yes

1) Proceed with installation (default)
2) Customize installation
3) Cancel installation
>1
```

默认安装

2、检查

```
C:\Users\MateBook13\.cargo\bin>rustc --version
rustc 1.62.1 (e092d0b6b 2022-07-16)

C:\Users\MateBook13\.cargo\bin>
```

3、新建项目

1、创建

创建

```
cargo new hello
```

目录结构

- hello
  - src
    - main.rs
  - Cargo.toml

2、运行

运行

```
cargo run
```

```
E:\Data\Rust>cd hello

E:\Data\Rust\hello>cargo run
   Compiling hello v0.1.0 (E:\Data\Rust\hello)
    Finished dev [unoptimized + debuginfo] target(s) in 0.72s
     Running `target\debug\hello.exe`
Hello, world!
```

3、添加依赖

在Cargo.toml中添加

```
[dependencies]
ferris-says = "0.1"
```

```
[package]
name = "hello"
version = "0.1.0"
edition = "2021"

# See more keys and their definitions at https://doc.rust-lang.org/cargo/reference/manifest

[dependencies]
ferris-says = "0.1"
```

安装依赖

```
cargo build
```

测试

```rust
//main.rs

use ferris_says::say; // from the previous step
use std::io::{stdout, BufWriter};

fn main() {
    let stdout = stdout();
    let message = String::from("Hello fellow Rustaceans!");
    let width = message.chars().count();

    let mut writer = BufWriter::new(stdout.lock());
    say(message.as_bytes(), width, &mut writer).unwrap();
}
```

运行： cargo run

```
PS E:\Data\Rust\hello> cargo run
    Finished dev [unoptimized + debuginfo] target(s) in 0.10s
     Running `target\debug\hello.exe`
 ---------------------------
| Hello fellow Rustaceans! |
 ---------------------------
              \
               \
                  ~^~^~
                 _     _
               \) /  o o  \ (/
                   '  _  '
                  / '-----' \
PS E:\Data\Rust\hello>
```

3、命令

1、更新

```
rustup update
```

2、卸载

```
rustup self uninstall
```

3、本地文档

```
rustup doc
```