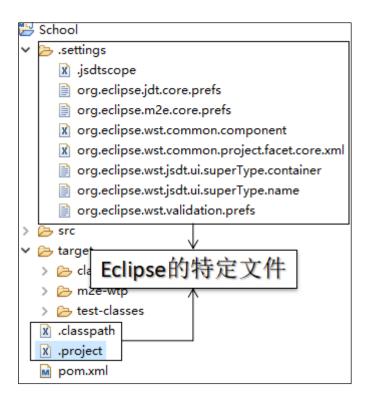
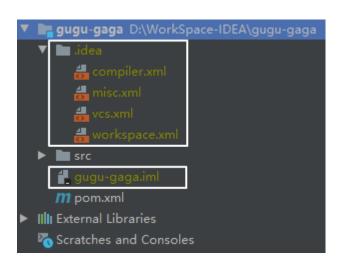
。第7章 IDEA 集成 Git

7.1 配置 Git 忽略文件

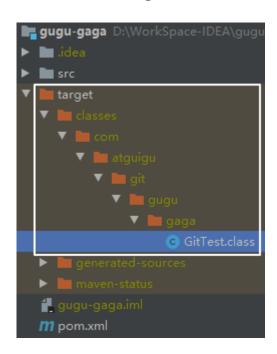
1) Eclipse 特定文件



2) IDEA 特定文件



3) Maven 工程的 target 目录



问题 1:为什么要忽略他们?

答:与项目的实际功能无关,不参与服务器上部署运行。把它们忽略掉能够屏蔽 IDE 工具之间的差异。

问题 2: 怎么忽略?

1) 创建忽略规则文件 xxxx.ignore (前缀名随便起,建议是 git.ignore)

这个文件的存放位置原则上在哪里都可以,为了便于让~/.gitconfig 文件引用,建议也放在用户家目录下

git.ignore 文件模版内容如下:

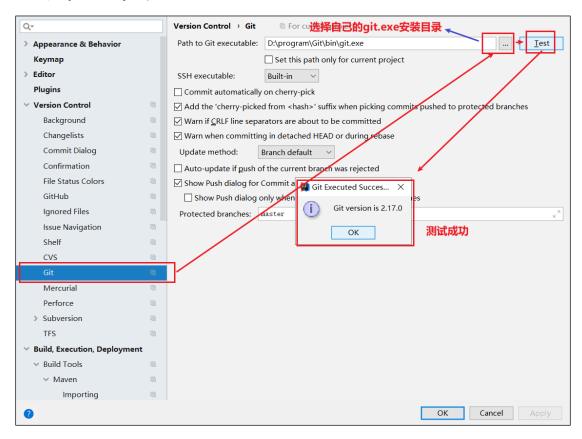
- # Compiled class file
- *.class

```
# Log file
*.log
# BlueJ files
*.ctxt
# Mobile Tools for Java (J2ME)
.mtj.tmp/
# Package Files #
*.jar
*.war
*.nar
*.ear
*.zip
*.tar.gz
*.rar
       virtual machine crash
                                                  logs,
                                                              see
http://www.java.com/en/download/help/error hotspot.xml
hs err pid*
.classpath
.project
.settings
target
.idea
*.iml
```

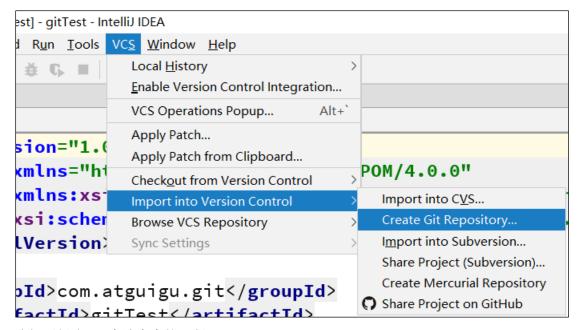
2) 在.gitconfig 文件中引用忽略配置文件(此文件在 Windows 的家目录中)

```
[user]
   name = Layne
   email = Layne@atguigu.com
[core]
   excludesfile = C:/Users/asus/git.ignore
注意: 这里要使用"正斜线(/)", 不要使用"反斜线(\)"
```

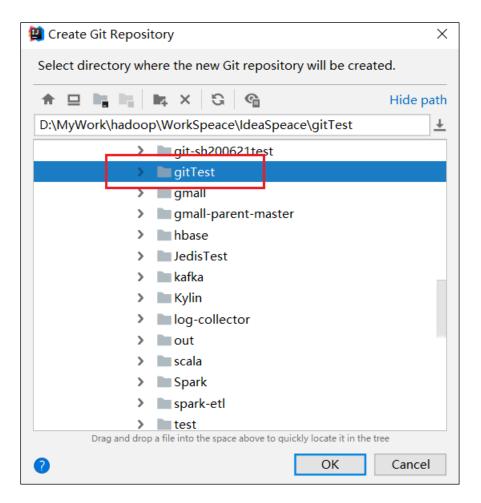
7.2 定位 Git 程序



7.3 初始化本地库

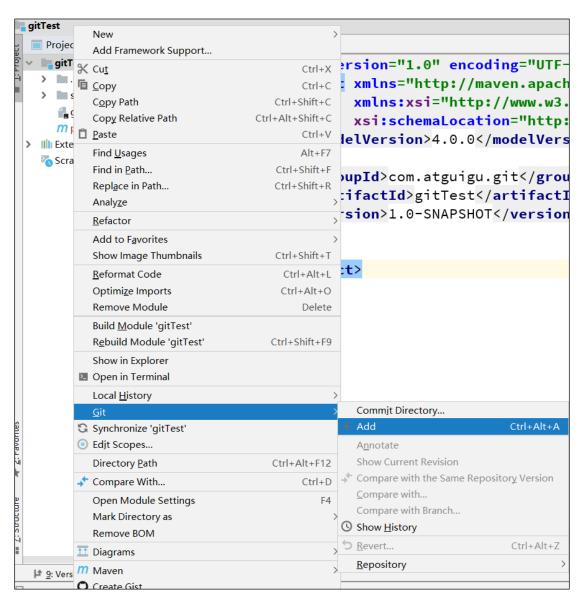


选择要创建 Git 本地仓库的工程。

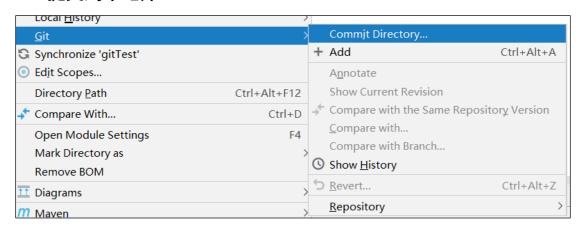


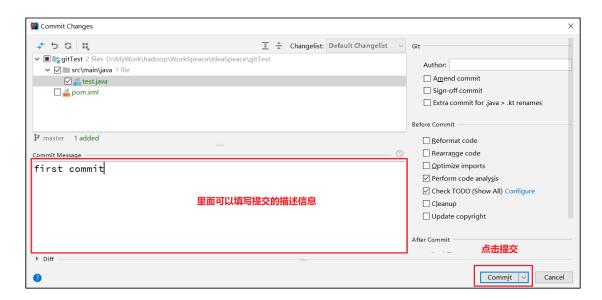
7.4 添加到暂存区

右键点击项目选择 Git -> Add 将项目添加到暂存区。



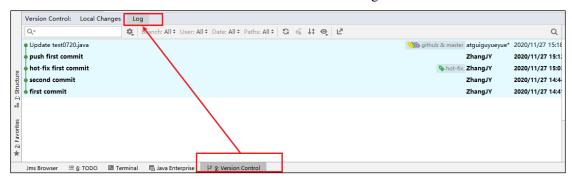
7.5 提交到本地库



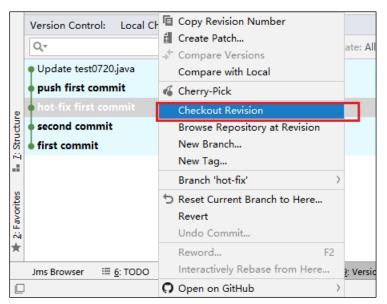


7.6 切换版本

在 IDEA 的左下角,点击 Version Control,然后点击 Log 查看版本

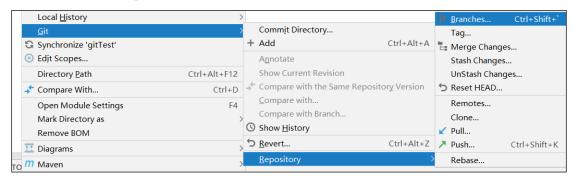


右键选择要切换的版本,然后在菜单里点击 Checkout Revision。

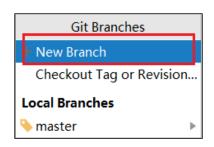


7.7 创建分支

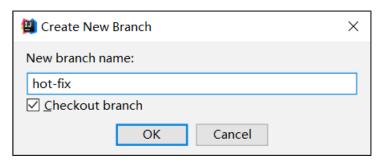
选择 Git, 在 Repository 里面, 点击 Branches 按钮。



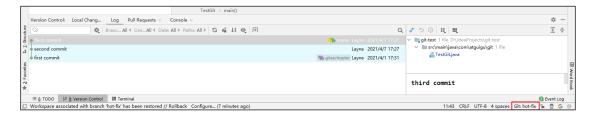
在弹出的 Git Branches 框里,点击 New Branch 按钮。



填写分支名称, 创建 hot-fix 分支。

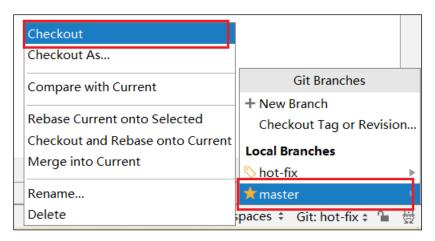


然后再 IDEA 的右下角看到 hot-fix, 说明分支创建成功, 并且当前已经切换成 hot-fix 分支



7.8 切换分支

在 IDEA 窗口的右下角,切换到 master 分支。

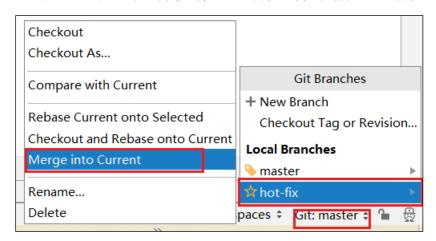


然后在 IDEA 窗口的右下角看到了 master,说明 master 分支切换成功。



7.9 合并分支

在 IDEA 窗口的右下角,将 hot-fix 分支合并到当前 master 分支。



如果代码没有冲突,分支直接合并成功,分支合并成功以后,代码自动提交,无需手动提交本地库。

```
■ Project ▼
                              🗸 📭 git-test D:
                                                  package com.atguigu.git;
 > 🗎 .idea
 ∨ 🗎 src
   🗸 🖿 main
                                                  * @author Layne
                                           4
     🗸 🖿 java
                                           5
                                                   * @create 2021-04-01 17:27
       v 🖿 com.atguigu.git
                                           6
            c TestGit
                                           7
                                                  public class TestGit {
                                           8
                                                      public static void main(String[] args) {
                                                          System.out.println("hello git!");
                                           9
       iava java
                                                          System.out.println("hello git2!");
                                          10
   #git-test.iml
                                          11
                                                          System.out.println("hello git3!");
   m pom.xml
                                                          System.out.println("hello git4!");
                                          12
 III External Libraries
                                          13
Scratches and Consoles
                                          14
                                          15
```

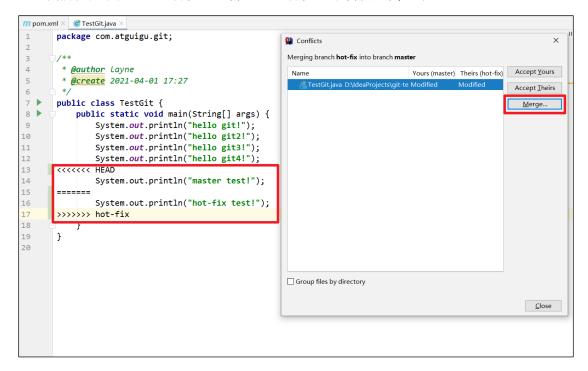
7.10 解决冲突

如图所示,如果 master 分支和 hot-fix 分支都修改了代码,在合并分支的时候就会发生冲突。

```
package com.atguigu.git;
3
4
       * @author Layne
        * <u>@create</u> 2021-04-01 17:27
5
6
7
       public class TestGit {
          public static void main(String[] args) {
8
9
              System.out.println("hello git!");
              System.out.println("hello git2!");
10
              System.out.println("hello git3!");
11
              System.out.println("hello git4!");
              System.out.println("master test!");
13
                     master分支修改代码
      }
15
16
```

```
package com.atguigu.git;
 1
 2
 3
 4
         @author Layne
 5
        * @create 2021-04-01 17:27
 6
 7
       public class TestGit {
           public static void main(String[] args) {
 8
              System.out.println("hello git!");
9
              System.out.println("hello git2!");
10
              System.out.println("hello git3!");
11
              System.out.println("hello git4!");
              System.out.println("hot-fix test!");
13
                    hot-fix分支修改代码
       }
15
16
```

我们现在站在 master 分支上合并 hot-fix 分支,就会发生代码冲突。



点击 Conflicts 框里的 Merge 按钮,进行手动合并代码。

```
Apply non-conflicting changes:
Changes from master
package com.atguigu.git;

    Do not ignore ∨ Highlight words ∨ □‡ ‡ ?

                                                                                                                                                package com.atguigu.git;
                                                                                                                                                                                                                                                                                           package com.atguigu.git;
                                                                                                                                                                                                                                                                                              * @author Layne
* @create 2021-04-01 17:27
    * @author Layne
                                                                                                                                                   * @author Layne
       @create 2021-04-01 17:27
                                                                                                                                                       @create 2021-04-01 17:27
*/
public class TestGit {
    public static void main(String[] arg |
        System.out.println("hello git!")
        System.out.println("hello git2!"
        System.out.println("hello git3!"
        System.out.println("hello git4!"
        System.out.println("master test!"
        System.out.println("master test!"
                                                                                                                                              public class TestGit {
   public static void main(String[] args) {
      System.out.println("hello git1!");
      System.out.println("hello git2!");
      System.out.println("hello git3!");
      System.out.println("hello git4!");
}
                                                                                                                                                                                                                                                                                          public class TestGit {
    public static void main(String[] args
        System.out.println("hello git!");
        System.out.println("hello git2!")
        System.out.println("hello git3!")
        System.out.println("hello git4!")
        System.out.println("hello git4!")
                                                                                                                                                                                                                                                                                                              System.out.println("hot-fix test
}
                                                                                                                                                                                                                                                                       15X: 删掉代码
16 《: 合并冲突代码
                        master分支代码
                                                                                                                                                                                     合并后的代码
                                                                                                                                                                                                                                                                                                                               hot-fix分支代码
 Apply Cancel
```

手动合并完代码以后,点击右下角的 Apply 按钮。

代码冲突解决,自动提交本地库。

```
■ git-test  

■ src  

■ main  

■ java  

□ com  

■ atguigu  
□ git  

© TestGit
  ■ Project ▼
                                   ⊕ 🛨 🜣 –
                                                            C TestGit.java ×
                                                 m pom.xml ×
 🗸 📭 git-test D
                                                         package com.atguigu.git;
   > idea
      ∨ ■ main
                                                          * @author Layne
                                                  4

✓ iava

                                                  5
                                                          * @create 2021-04-01 17:27
           com.atquiqu.git
                                                  6
               TestGit
                                                  7
                                                         public class TestGit {
          resources
                                                              public static void main(String[] args) {
                                                    8
       ✓ 🖿 test
                                                                  System.out.println("hello git!");
          iava java
                                                  9
                                                                  System.out.println("hello git2!");
                                                 10
   > target
      🖺 git-test.iml
                                                                  System.out.println("hello git3!");
                                                 11
      m pom.xml
                                                                  System.out.println("hello git4!");
                                                 12
 > || External Libraries
                                                                  System.out.println("master test!");
                                                 13
   Scratches and Consoles
                                                                  System.out.println("hot-fix test!");
                                                 14
                                                 15
                                                 16
                                                         }
                                                 17
```