

在 IntelliJ IDEA 中，没有类似于 Eclipse 工作空间（Workspace）的概念，而是提出了 Project 和 Module 这两个概念。接下来，就让咱们一起看看 IntelliJ IDEA 官方是如何描述两者的吧！

对于 Project，IntelliJ IDEA 官方是这样介绍的：

A project is a top-level organizational unit for your development work in IntelliJ IDEA. In its finished form, a project may represent a complete software solution. A project is a collection of:

- Your work results: source code, build scripts, configuration files, documentation, artifacts, etc.
- SDKs and libraries that you use to develop, compile, run and test your code.
- Project settings that represent your working preferences in the context of a project.

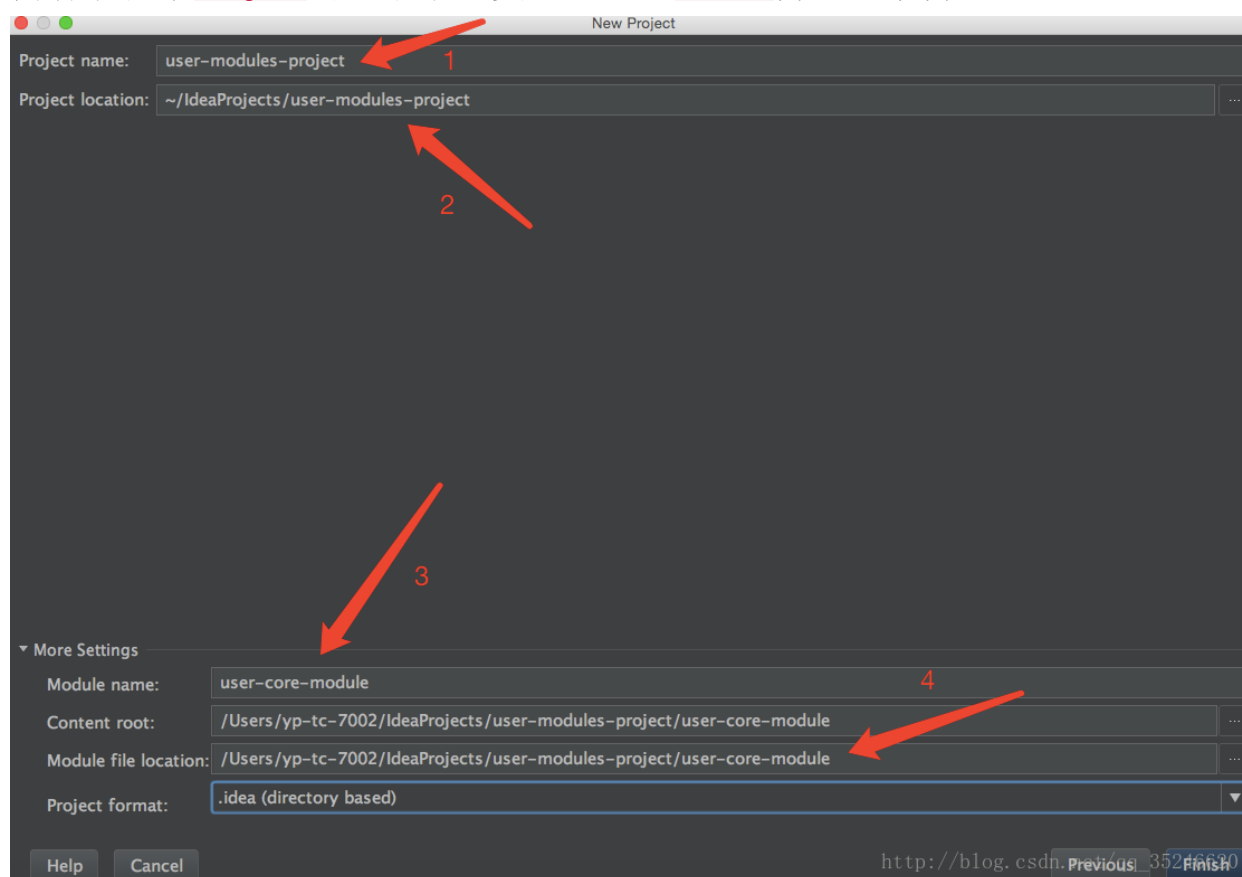
A project has one or more modules as its parts.

对于 Module，IntelliJ IDEA 官方是这样介绍的：

- A module is a part of a project that you can compile, run, test and debug independently.
- Modules are a way to reduce complexity of large projects while maintaining a common (project) configuration.
- Modules are reusable: if necessary, a module can be included in more than one project.

通过上面的介绍，咱们知道，在 IntelliJ IDEA 中 Project 是最顶级的结构单元，然后就是 Module，一个 Project 可以有多个 Module。目前，主流的大型项目结构基本都是多 Module 的结构，这类项目一般是按功能划分的，比如：user-core-module、user-facade-module 和 user-hessian-module 等等，模块之间彼此可以相互依赖。通过这些 Module 的命名可以看出，她们都是处于同一个项目中的模块，彼此之间是有着不可分割的业务关系的。

因此，咱们可以大致总结出：一个Project是由一个或多个Module组成，尽量让各模块处于同一项目之中，此时彼此之间具有互相依赖的关联关系。在这里，之所以说“尽量”，是因为 IntelliJ IDEA 的Project是一个没有具备任何编码设置、构建等开发功能的，主要起到一个项目定义、范围约束、规范类型的效果，或许，咱们也可以简单地理解Project就是一个单纯的目录，只是这个目录在命名上必须有其代表性的意义。在一般情况下，IntelliJ IDEA 是默认单Project单Module的，这时Project和Module合二为一，在没有修改存储路径的时候，显然Project对Module具有强约束作用啦！不过说实话，这里就是将Module的内容放在了Project的目录下，实际上还是Module自己约束自己。

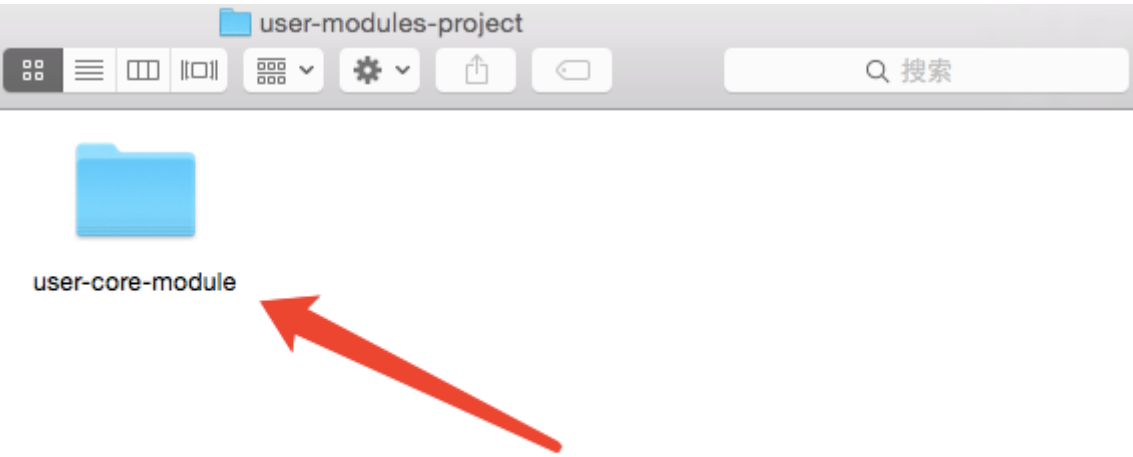


000

- 标注1：Project name，项目名称；
- 标注2：Project location，项目存储地址；
- 标注3：Module name，模块名称；
- 标注4：Module file location，模块存储地址。

如上图所示，通过观察Project和Module的存储地址，咱们可以发现，在此处建立了一个名为user-core-module的目录，并将其放在了名为user-modules-project

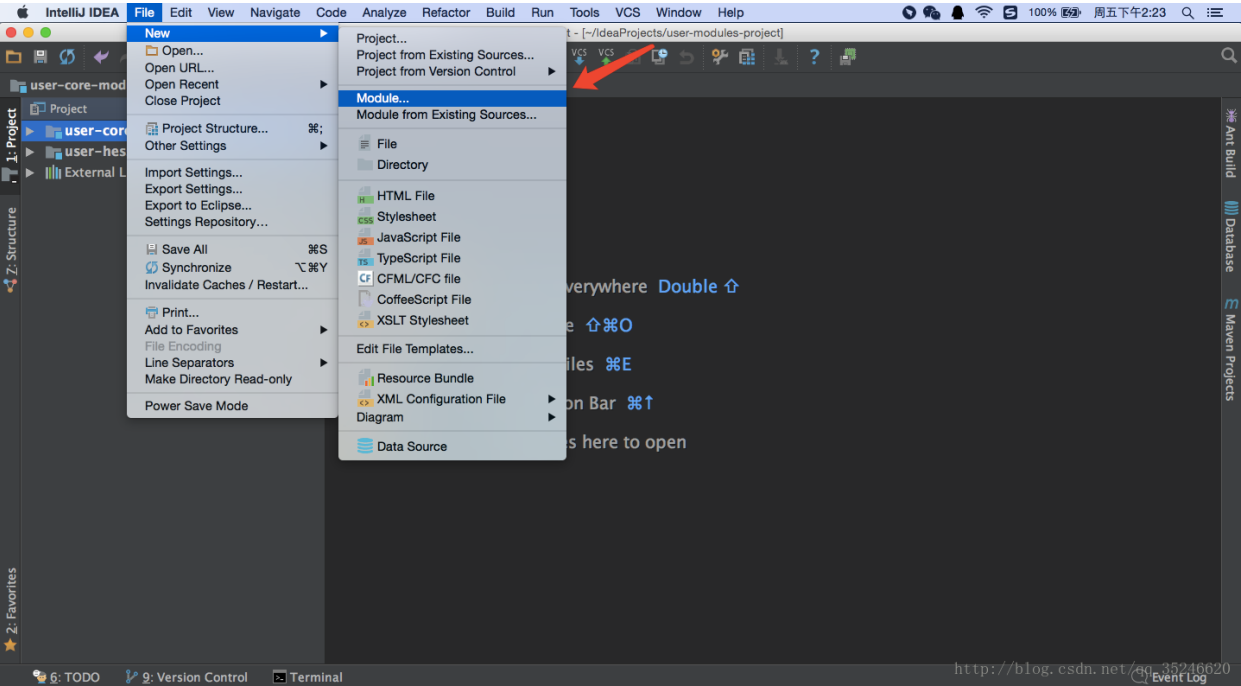
的目录下，而没有将两个目录合二为一，也就为咱们建立多Module的Project作了准备。



[http://blog.csdn.net/qq\\_35246620](http://blog.csdn.net/qq_35246620)

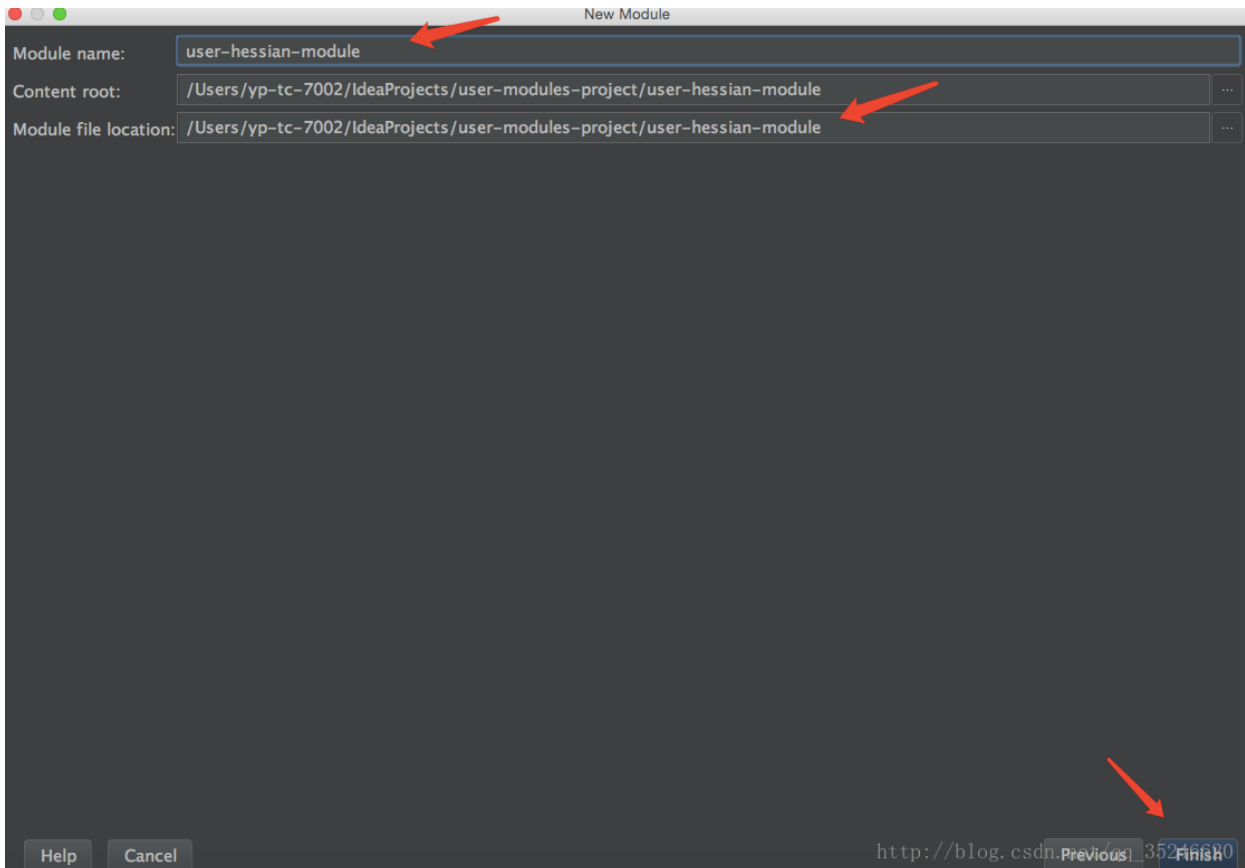
111

如上图所示，显然user-modules-project仅表现为一个目录而已。



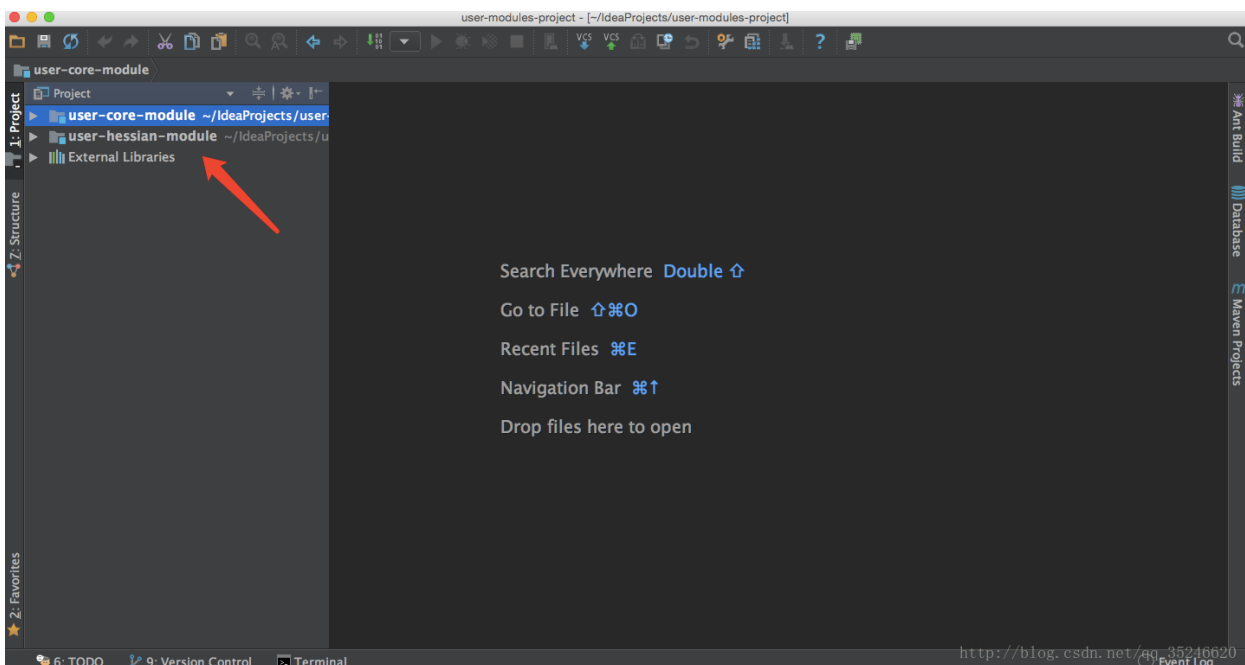
112

如上图所示，依次点击File -> New -> Module，进入如下界面：



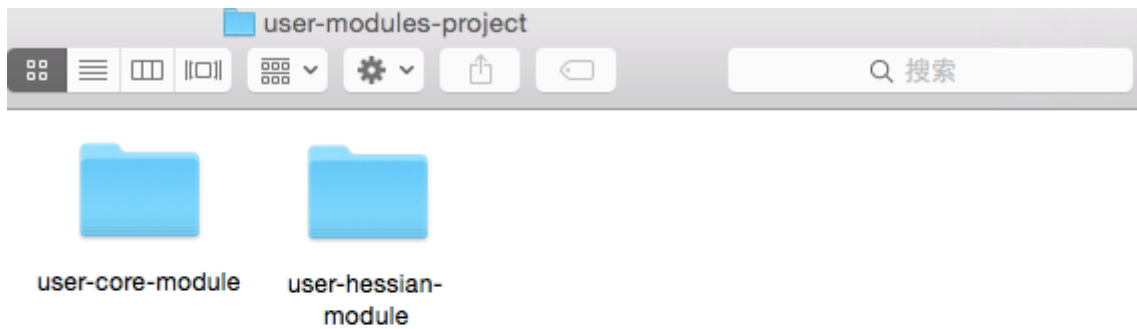
113

如上图所示，输入Module name之后，Content root和Module file location自动发生改变，然后点击finish，完成：



114

如上图所示，咱们在项目user-modules-project中，建立了两个Module，分别为user-core-module和user-hessian-module，接下来，咱们再看看存储目录：



[http://blog.csdn.net/qq\\_35246620](http://blog.csdn.net/qq_35246620)

115

如上图所示，显然在user-modules-project目录中，又多了一个名为user-hessian-module的目录。至此，多Module的Project构建完成！