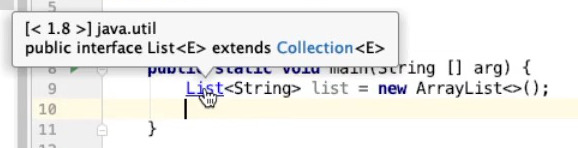
**Exercise**

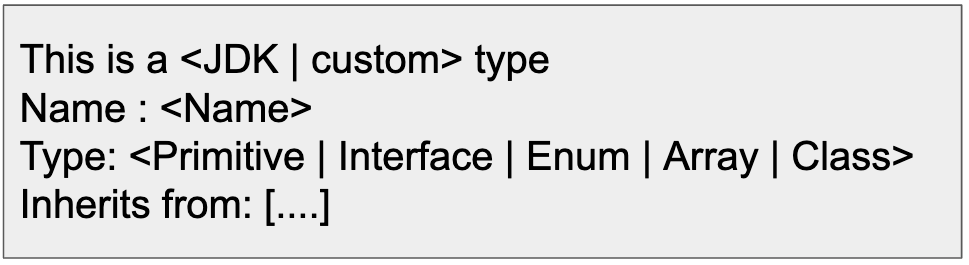
Some Java IDEs use Reflection to inspect the code we, the developers are typing, and provide us with additional information about this code.

For example if we hover over a JDK based class we would see a popup with information about this class:





In this exercise we want to help develop a different, smaller plugin which will provide a different popup window that looks like this:

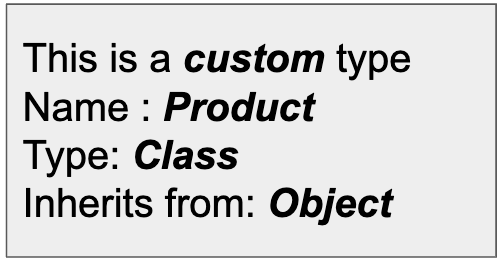


For example:

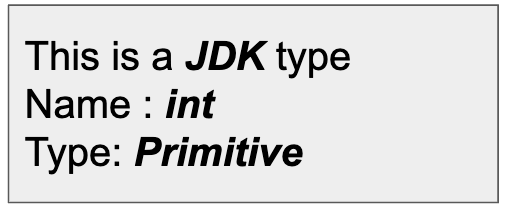
If we hover over the class List we will see this popup:



if we hover over our custom *Product* class we would see a popup window that looks like this:



And if we hover over a int type we would see a popup window that looks like this:



**Inherited Classes**

* If the input is an **interface** the inherited classes are all the names of the directly extended interfaces.
* If the input is NOT an interface, the inherited class is either the **name of the direct class it extends** or **null**

**JDK vs Custom Types**

To determine if a type is a JDK type or a custom type we need to inspect the package of the type.

Classes that belong to a package that starts with one of the following prefixes:

"com.sun.", "java", "javax", "jdk", "org.w3c", "org.xml"

is considered a JDK class

Primitive types do NOT belong to any package, and are also considered JDK types