Extract Delegate

The *Extract Delegate* refactoring enables extracting some of the fields and methods of a class into a separate, newly created class. This refactoring is useful, when a class has grown too large, too incoherent, or "does too many things". In those cases, good design suggests splitting the class into smaller, more cohesive classes. This refactoring will help automate the process.

> This refactoring is also available from UML Class diagram.

Example

Before	After
| ```
public class Foo {
    private String b;
    public String getInfo() {
        return ("(" + b + ")");
    }
    ...
}

public class Bar {
    Foo foo;
    String t2 = foo.getInfo();
    ...
}
``` | ```
public class Foo {
    private final Info info = new Info();
    public String getInfo() {
        return info.getInfo();
    }
    ...
}

public class Info {
    private String b;
    public Info() {}
    public String getInfo() {
        return ("(" + b + ")");
    }
}

public class Bar {
    Foo foo;
    String t2 = foo.getInfo();
    ...
}
``` |

To extract selected members to a class

1. Select the desired class. To do that, either open the class in question for editing, and position the caret anywhere in the class, or just click such class in the Project tool window.

2. Choose **Refactor | Extract | Delegate** on the main menu or on the context menu of the selection.

3. In the **Extract Delegate** dialog box:

   - Specify the name and package of the new class to be created.

   - Check the names of the members of the existing class to be included in the new class.

   - Click **Preview** to make IntelliJ IDEA search for usages of the selected fields or methods, and display the refactoring preview results in the Find tool window. This preview shows all found usages of the variable or method to be moved to a new class. In this preview, you can include usages into refactoring or skip them. Click **Do Refactor** to apply refactoring to the selected usages.
   If you do not want to view usages, click **Refactor**. In this case, the usages will be changed immediately.

   > The Refactoring preview window may appear anyway, if the files to be affected are read-only.

See Also

Concepts:

- Refactoring Source Code

Reference:

- Extract Delegate Dialog

Web Resources:

- Developer Community ⊡