SMA: Software Modeling and Analysis
A2020

Prof. Dr. Oscar Nierstrasz
Pascal Gadient, Pooja Rani

## Solution
## Assignment 11 — 25.11.2020 – v1.2
## Querying the Class Hierarchy

Please submit this exercise by mail to pascal.gadient@inf.unibe.ch before 02. December 2020, 10:15am.

**You must submit your code as editable text, i.e., use plain text file(s).**

In order to perform this assignment, you must import "Project Analyzer" and "Hierarchy Analyzer" from Github. You can do that in GT with just few steps. First, click on the "Git" button on the main screen. Next, click on the "+" sign at the top right, just next to the button "Fetch All". In the overlay view, click on "clone" and then enter the URL below into the text field.

`https://github.com/poojaruhal/sma2020.git`

Next, click on "Clone", then double-click on the value "sma2020" that should now appear in the list. In the upcoming view on the right hand side, please click on all three "Load" buttons. You successfully completed the clone of the repository when you see "Up to date" for every package. Once again, we recommend to save your image when you finished the clone process.

### Exercise 1: Exploring projects (3 pts)

In GT, a project does not have a clear structure, but instead it consists of multiple packages. In this exercise, we will work on an easy to use project explorer that can present all the packages or classes of a project.

*NB:*

- *GT follows the convention that all package names with the same base name* (i.e., *prefix) belong to the same project. For example, packages related to the project* `Collection` *start with the prefix "Collection",* e.g., `CollectionsAbstract`, `CollectionsStack`, *etc.*

- *We define a "package group" as a list of one or more packages that use the same base name.*

a) Describe the steps required to add an extension method. You can find an example in the Github repository you just cloned. **Answer:**

   *GT:*

   i) *Open a class whom behavior you want to extend in* Calypso.

   ii) *Add the extension method in the class.*

   iii) *In the category, input the character "*" and the name of your package.*

   iv) *You can also select your target package from the dropdown list.*

   *Pharo:*

   i) *Open the class whom behavior you want to extend in* Calypso.

   ii) *In the menu, choose "Move to package".*

   iii) *Select your target package.*

*You can also achieve that programmatically:*
```
ExistingClass compile:  'myMethod'
    ^ true classified:  '\*MyPackage'
```

b) How many unique projects exist in the GT image? The class `RPackageOrganizer` can assist you with this task. **Answer:**

```
allPackages :=
    RPackageOrganizer default packages
        collect:  #packageGroup.
    ^ allPackages asSet.
```

c) Collect all packages of the `Collections` project. The extension method `packageGroup` in `RPackage` can assist you with this task. **Answer:**

```
allPackages :=
    RPackageOrganizer default packages
        select:  [ :e | e packageGroup match:  'Collections'].
    ^ allPackages asSet.
```

## Exercise 2: Exploring hierarchies (7 pts)

A very common task in static code analysis is to fetch information about the inheritance relation between different classes. This requires the traversal of subclasses and superclasses, and the computation of hierarchy depth for each class. In this exercise, you have to map the GT class hierarchy to the project structure as discussed in the lecture. You are supposed to consider the `Object` class as root of all hierarchies, because all classes inherit from `Object`.

a) Calculate the class hierarchy depth of the `Collection` class.
   **Answer:**

   *The hierarchy depth is 2.*

   *You can retrieve that number with:* `Collection classDepth – 1.`

b) Does the superclass of `HashedCollection` reside in the same package? You can use the extension method `isSuperclassInAnotherPackage` in `Class` from the Github repository. **Answer:**

   *Yes, because the statement below returns* `false`*. You can easily validate that with the class view in GT.*
   `HashedCollection isSuperclassInAnotherPackage.`

c) Do the subclasses of `HashedCollection` reside in the same package? You can use the extension method `isSubclassesInAnotherPackage` in `Class` from the Github repository. **Answer:**

   *No, because the statement below returns* `true`*. You can easily validate that with the class view in*

SMA: Software Modeling and Analysis
A2020

Prof. Dr. Oscar Nierstrasz
Pascal Gadient, Pooja Rani

*GT.*
*HashedCollection isSubclassesInAnotherPackage.*

d) Calculate the depth of `Collection` in the package hierarchy. You can use the message implementation `HA_ClassExplorer calculatePackageLevel` from the Github repository. **Answer:**

*The package hierarchy depth is 1.*
*(HA_ClassExplorer asExplorerClass:  Collection) calculatePackageLevel.*

e) Add reasonable class comments to the classes `HA_ClassExplorer` and `PA_ProjectExplorer`. With the help of your comments, ensure the classes will be understood by other developers.

*NB: You can refer to the Pharo class comment template shown in the lecture, however you are not required to. You should write just about the information you believe is necessary to understand the class.*
**Answer:**

*Class:  HA_ClassExplorer*

*I represent a class explorer.*

*Responsibility:*
*I am responsible for exploring a class hierarchy in a package scope.*
*I know the package level.*

*Public API and key messages:*
*#calculatePackageLevel*
*I compute the hierarchy level of a class.*
*#asExplorerClass:  aClass*
*I return the explorer instance of the given class.*
*#isRootClassInPackage:  aClass*
*I can tell if a class is a root class in the package.*

*InstanceVariables*
*<packageLevel> package level of a class.*

*Example use:*
*HA_ClassExplorer asExplorerClass:  HashedCollection.*

*Class:  PA_ProjectExplorer*

*I represent a project explorer.*

SMA: Software Modeling and Analysis
A2020

Prof. Dr. Oscar Nierstrasz
Pascal Gadient, Pooja Rani

```
Responsibility:
I am responsible for exploring the class hierarchy in a project scope.
A project is composed of various packages starting with same prefix.
Each package has various classes in it.

Public API and key messages:
#classes
I return all the classes available in a project.

InstanceVariables
<packages> all packages in the project.
```

f) In this task, you are supposed to implement the method `calculateProjectLevel` in the class HA_ClassExplorer. **Answer:**

*Please refer to the repository (here) where you can find the detailed solution.*