Assignment 1: Program FSIS

1. Requirement

Program FSIS of a flower shop captures and processes data about an entity named `Customer` and a specific type of `Customer` named `HighEarner`. The former represents customers that are of interest to the shop, while the latter represents wealthy customers whose income are above a given threshold. Figure 1 shows the concept class diagram of FSIS. Table 1 lists the attribute design details of `Customer` and `HighEarner`. Note that attribute `income` is specific to `HighEarner`.

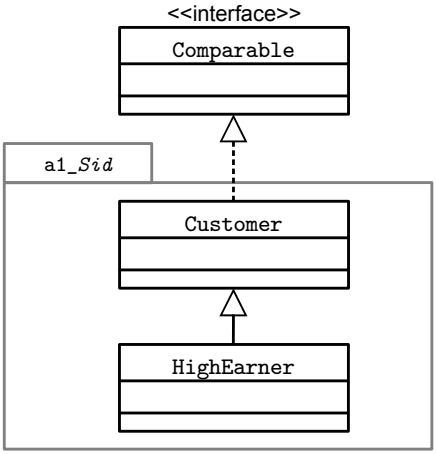

Figure 1: FSIS class diagram.

To ease listing of customer objects, program FSIS requires that `Customer` realises an interface named `Comparable`. This interface is provided by Java. The implementation of the method `Comparable.compareTo` must compare two `Customers` by name.

Please note that your program will be marked automatically by a program, which expects you to strictly follow all the relevant the design rules. An error in one part may affect other parts.

Table 1: The attribute design details of `Customer` and `HighEarner`.

Attribute(s)	formal type	mutable	optional	min	max	length
`id`	`Integer`	F	F	for `Customer`		
				1	10^9	-
				`HighEarner`		
				10^7	-	-
`name`	`String`	T	F	-	-	50
`phoneNumber`	`String`	T	F	-	-	10
`address`	`String`	T	F	-	-	100
Specific to **`HighEarner`**						
`income`	`Float`	T	F	10^7	-	-

2. Tasks

1. Create a package named `a1_`*`Sid`* as shown in Figure 1, where *`Sid`* is your student id. For example, if your student id is 123456789 then the package name is `a1_123456789`.

   You will need to use this package to hold all the Java classes that you create for the program.

   IMPORTANT: Failure to name the package as described above will result in an invalid program.

2. Specify and implement class `Customer` with all the ***essential*** attribute(s) and operations.

   Note: this class must use exceptions where suitable to increase code robustness.

3. Specify and implement class `HighEarner` with all the ***essential*** attribute(s) and operations.

   Note: this class

   - must use exceptions where suitable to increase code robustness.
   - should invoke the shared data validation methods inherited from `Customer` when necessary

IMPORTANT: The essential mutator and observer operations of the same attribute affect one another. An error in one operation affects the validity of the other operation.

3. Submission

Create a zip-compressed file containing **just the folder of the package** specified in Task 1. You must name the file as follows: **a1_*Sid*.zip**, where ***Sid*** is your student id.

Submit your file to the designated submission box for this assignment.

IMPORTANT: Failure to name the file as described above will result in an invalid program. In particular, ONLY the **ZIP** format is accepted. Other formats (e.g. RAR) are NOT accepted.