Exercise 3:

Summary: Zoophobia is the fear of animals and some of them are ailurophobia (fear of cats) and cynophobia (fear of dogs). In what follows you have to write two classes that will serve as a dictionary for such phobias. Each instance of **Zoophobia** corresponds to one phobia. **Dictionary** contains the list of the phobias. **Exercise** has the method *main*. You load in the information stored in the text file *old.txt* and create with its content instances of **Zoophobia**. You update, add and delete some phobias and save the database to disk.

Task 1: Implement Zoophobia (2 points):

The two instance variables store the names of the phobia and the animal.

The constructor initializes both instance variables.

setPhobia(String) and *getPhobia()* set and return the value of *phobia*.

setAnimal(String) and *getAnimal()* set and return the value of *animal*.

Zoophobia
−phobia : String −animal : String
+Zoophobia(String, String) +setPhobia(String) : void +setAnimal(String) : void +getPhobia() : String +getAnimal() : String +compareTo(Zoophobia) : int + toString() : String

compareTo(arg) compares the values of *phobia* alphabetically and we will use this method to sort the phobias.

toString() returns the value of *phobia* followed by " - fear of " and the value of *animal*.

Task 2: Implement Dictionary (4 points):

The instance variable *dataBase* contains the list of all zoophobias.

The constructor initializes *dataBase*.

load(arg) loads the content of the file that is attached to *arg*.

sort() sorts the list of zoophobias alphabetically by the value of *phobia*.

Dictionary
−dataBase : ArrayList<Zoophobia>
+Dictionary() +load(File) : void +sort() : void +insert(Zoophobia) : void +delete(String) : void +lookup(String) : String +save(File) : void +toString() : String

insert(arg) goes through the list of all zoophobias and compares their value of *phobia* with the *phobia* of *arg*. It does nothing if the phobias and animals are the same. If *dataBase* contains a zoophobia with the same value of *phobia* as *arg* but with different values of *animal* then it updates the one in *dataBase*. If the *phobia* of *arg* is not stored in *dataBase* then it adds *arg* to *dataBase*.

delete(arg) deletes the zoophobia with *phobia* = *arg*.

lookup(arg) searches *dataBase* for a zoophobial with the value *phobia* = *arg* and returns the associated value of *animal*.

save(arg) saves the return value of *toString()* to the file *arg*. Catch IO errors.

toString() concatenates the return values of the methods *toString()* of all zoophobias (one per line) to one string and returns this string.

Task 3: Implement Exercise (4 points):

Read in the name of the file to open from the console and call the method *load(arg)* of **Dictionary** with it. Ask on the console how many new phobias you want to insert into the data base. Check if the phobias are already known and update the animal if necessary. If the phobias are not yet stored in the database then you should add them to it. List the content of the database after that. Implement a loop that allows you to delete phobias until you type "end". Sort the phobias alphabetically by calling the method *sort()* of **Dictionary**. Write the content of the database to the file *arg* using the method *save(arg)* of **Dictionary**. The filename should be set to a string that was read in from the console.