Contents

1 Basic Test Results

```
======================
===== EXO TESTER =====
======================


===== EXTRACTING =====


===== CHECKING FILES =====


===== ANALYZE README =====
Whats up shiragelbstein?


==== CHECKING LINE LENGTHS ====


===== COMPILE CODE =====


===== EXECUTE TESTS =====
************************* Test 1: Bot Class *************************
Description: BASIC behavior of the bot class
BOT: Passed 7 out of 7
PASSED, Superb!
*********************************************************************
************************* Test 2: Chat Class *************************
Description: BASIC behavior of the chat class
CHAT: Passed 1 out of 1
PASSED, Spectacular!
**********************************************************************

You passed all the tests, GGWP
```

2 README

```
shiragelbstein
212926737
```

3 Chat.java

```java
import java.lang.reflect.Array;
import java.util.Scanner;

public class Chat {

        public static void main(String[] args) {
                String[] stringArray1 = new String[]{"what ", "should i say "};
                String[] stringArray3 = new String[]{"say <phrase> ok ill say <phrase>: <phrase>! "};
                ChatterBot chatter1= new ChatterBot("Shira", stringArray3,stringArray1);
                String[] stringArray2 = new String[]{"whaaat ", "say say "};
                String[] stringArray4 = new String[]{"haha halourios, but ill say it: <phrase>. "};
                ChatterBot chatter2= new ChatterBot("Tali",stringArray4,stringArray2);
                ChatterBot[] arrofchatter = {chatter1,chatter2};
                String statement = "say boo";
                //int x=0;
                while(true){
                     for(ChatterBot bot : arrofchatter)
                     {
                             statement = bot.replyTo(statement);
                             System.out.print(bot.getName()+": ");
                             System.out.print(statement);
                             System.out.print("\n");
                             //x++;
                     }
                }

        }
}
```

4 ChatterBot.java

```java
import java.util.*;

/**
 * Base file for the ChatterBot exercise.
 * The bot's replyTo method receives a statement.
 * If it starts with the constant REQUEST_PREFIX, the bot returns
 * whatever is after this prefix. Otherwise, it returns one of
 * a few possible replies as supplied to it via its constructor.
 * In this case, it may also include the statement after
 * the selected reply (coin toss).
 * @author Dan Nirel
 */
class ChatterBot {
    static final String REQUEST_PREFIX = "say ";
    static final String PLACEHOLDER_FOR_REQUESTED_PHRASE = "<phrase>";
    static final String PLACEHOLDER_FOR_ILLEGAL_REQUEST = "<request>";
    Random rand = new Random();
    String[] repliesToIllegalRequest;
    String[] legalRequestsReplies;
    String name;
    ChatterBot(String name, String[] repliesToLegalRequest, String[] repliesToIllegalRequest) {
        this.repliesToIllegalRequest = new String[repliesToIllegalRequest.length];
        for(int i = 0 ; i < repliesToIllegalRequest.length ; i = i+1) {
            this.repliesToIllegalRequest[i] = repliesToIllegalRequest[i];
        }
        this.legalRequestsReplies = new String[repliesToLegalRequest.length];
        for(int i = 0 ; i < repliesToLegalRequest.length ; i = i+1) {
            this.legalRequestsReplies[i] = repliesToLegalRequest[i];
        }
        this.name = name;
    }

    String replyTo(String statement) {
        if(statement.startsWith(REQUEST_PREFIX)) {
            //we don’t repeat the request prefix, so delete it from the reply
            String newstatement = statement.replaceFirst(REQUEST_PREFIX, "");
            return replyToLegalRequest(newstatement);
        }
        return replyToIllegalRequest(statement);
    }

    String replyToIllegalRequest(String statement) {
        return replacePlaceholderInARandomPattern
                (repliesToIllegalRequest,PLACEHOLDER_FOR_ILLEGAL_REQUEST,statement);
    }

    String getName() {
        return this.name;
    }

    String replyToLegalRequest(String statement) {
        return replacePlaceholderInARandomPattern
                (legalRequestsReplies,PLACEHOLDER_FOR_REQUESTED_PHRASE,statement);
    }

    String replacePlaceholderInARandomPattern(String[] patterns, String placeholder, String replacement)
    {
        int randomIndex = rand.nextInt(patterns.length);
        String responsePattern = patterns[randomIndex];
        String reply;
        reply = responsePattern.replaceAll(placeholder, replacement);
        return reply;
    }

}
```