**Question 1 - STEPS**

1. Define and initialize variable; str, minStr=” ”, minLen, count=0
2. Define constructor, initialize str and Set minLen to length of str
3. Loop i=0 to str.length
   1. Loop j=0; to str.length

Check if each substring is unique and length is less than minLen

Set minLen to length of substring

Set minStr to subStr

1. Check if count of substring

public class Q2ReverseOrder {

    // set the default value of the variable

    protected String sentence = "A quick brown fox jumps over the lazy dog";

    protected String reversedSentence = "";

    // constructor

    public Q2ReverseOrder() {

    }

    // getter

    public String getSentence() {

        return sentence;

    }

    // setter

    public void setSentence(String sentence) {

        this.sentence = sentence;

    }

    // getter

    public String getReversedSentence() {

        return reversedSentence;

    }

    // setter

    public void setReversedSentence(String reversedSentence) {

        this.reversedSentence = reversedSentence;

    }

    // print reversed string

    public void printReversedStr() {

        System.out.println("From Based class: " + getReversedSentence());

    }

}

class Main extends Q2ReverseOrder {

    // constructor

    public Main(String sentence) {

        this.sentence = sentence;

    }

    // method to reverse the string

    public final void reverseStr() {

        String[] strArray = getSentence().split(" ");

        for (int i = 0; i < strArray.length; i++) {

            reversedSentence = strArray[i] + " " + reversedSentence;

        }

    }

    // print reversed string

    public void printReversedStr() {

        System.out.println("From derived class: " + getReversedSentence());

    }

    public static void main(String[] args) {

        // create an object of the class name q2

        Main q2 = new Main("A chicken is a bird");

        // create an object of the class name q3

        Main q3 = new Main("I am a student");

        // call the method to reverse the string

        q2.reverseStr();

        q3.reverseStr();

        // print the reversed string

        q2.printReversedStr();

        q3.printReversedStr();

    }

}

**Question 2 – STEPS**

1. Define

public class Q2ReverseOrder {

    // set the default value of the variable

    protected String sentence = "A quick brown fox jumps over the lazy dog";

    protected String reversedSentence = "";

    // constructor

    public Q2ReverseOrder() {

    }

    // getter

    public String getSentence() {

        return sentence;

    }

    // setter

    public void setSentence(String sentence) {

        this.sentence = sentence;

    }

    // getter

    public String getReversedSentence() {

        return reversedSentence;

    }

    // setter

    public void setReversedSentence(String reversedSentence) {

        this.reversedSentence = reversedSentence;

    }

    // print reversed string

    public void printReversedStr() {

        System.out.println("From Based class: " + getReversedSentence());

    }

}

class Main extends Q2ReverseOrder {

    // constructor

    public Main(String sentence) {

        this.sentence = sentence;

    }

    // method to reverse the string

    public final void reverseStr() {

        String[] strArray = getSentence().split(" ");

        for (int i = 0; i < strArray.length; i++) {

            reversedSentence = strArray[i] + " " + reversedSentence;

        }

    }

    // print reversed string

    public void printReversedStr() {

        System.out.println("From derived class: " + getReversedSentence());

    }

    public static void main(String[] args) {

        // create an object of the class name q2

        Main q2 = new Main("A chicken is a bird");

        // create an object of the class name q3

        Main q3 = new Main("I am a student");

        // call the method to reverse the string

        q2.reverseStr();

        q3.reverseStr();

        // print the reversed string

        q2.printReversedStr();

        q3.printReversedStr();

    }

}

**Question 3 – STEPS**

1. Define

package positive;

public class Q3Postive2Digit {

    protected int checkAgainst;

    protected int num1, num2;

    protected int sum;

    public Q3Postive2Digit(int checkAgainst) {

        this.checkAgainst = checkAgainst;

    }

    public int getSum(int num1, int num2) {

        return num1 + num2;

    }

    public void findNumbers() {

        for (int i = 0; i < this.checkAgainst; i++) {

            for (int j = 0; j < this.checkAgainst; j++) {

                sum = getSum(i, j);

                if (sum == this.checkAgainst) {

                    System.out.println(i + " & " + j + " can be added to get " + this.checkAgainst);

                }

            }

        }

    }

}

package positive;

public class MyException extends Exception {

    // constructor

    public MyException(String str) {

        // super(str);

        System.out.println("MyException: " + str);

    }

}

package positive;

public class Main {

    public static void main(String[] args) throws MyException {

        // Q3Postive2Digit q3 = new Q3Postive2Digit(15);

        // Uncomment the following line to see the error message

        Q3Postive2Digit q3 = new Q3Postive2Digit(-15);

        try {

            if (q3.checkAgainst < 0) {

                throw new MyException("Cannot find numbers to be added to get " + q3.checkAgainst);

            }

            q3.findNumbers();

        } catch (Exception e) {

        }

    }

}



