

HO GENT

OOSDII

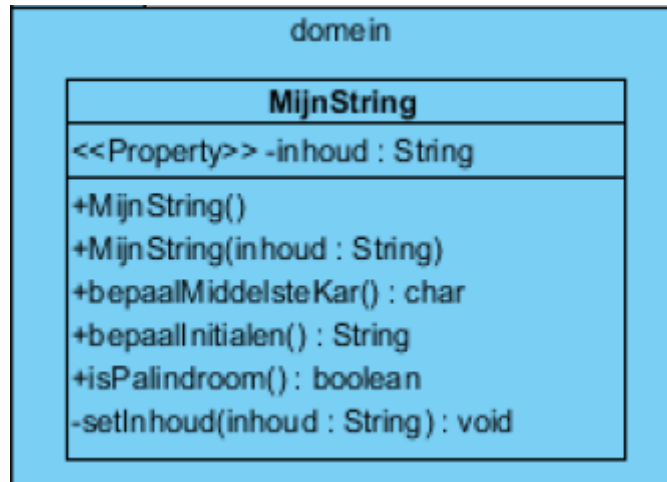
Strings en reguliere expressies

Table of Contents

1. Oefening 1	1
1.1. bepaalMiddelsteKar	1
1.2. isPalindroom	1
2. Oefening 2	1
2.1. keerOm	1
2.2. isPalindroom	1
2.3. Werk onderstaande gui uit	1
3. Oefening 3	2
3.1. maakWartaal	2
4. Oefening 4: validatie	4
5. Oefening 5: splitsen van teksten	4
6. Oefening 6	4
6.1. maakWartaal	5
6.2. splitsOp	5
7. Oefening 7	6
7.1. PatternMatcher1	6
7.2. PatterMatcher2	6

1. Oefening 1

Werk de domeinklasse MijnString uit (zie chamilo voor het startproject met testklasse)



1.1. bepaalMiddelsteKar

Geeft het middelste karakter van inhoud terug.

1.2. isPalindroom

Bepaalt of inhoud een palindroom is.

2. Oefening 2

Voeg de volgende methode toe in de klasse MijnString

2.1. keerOm

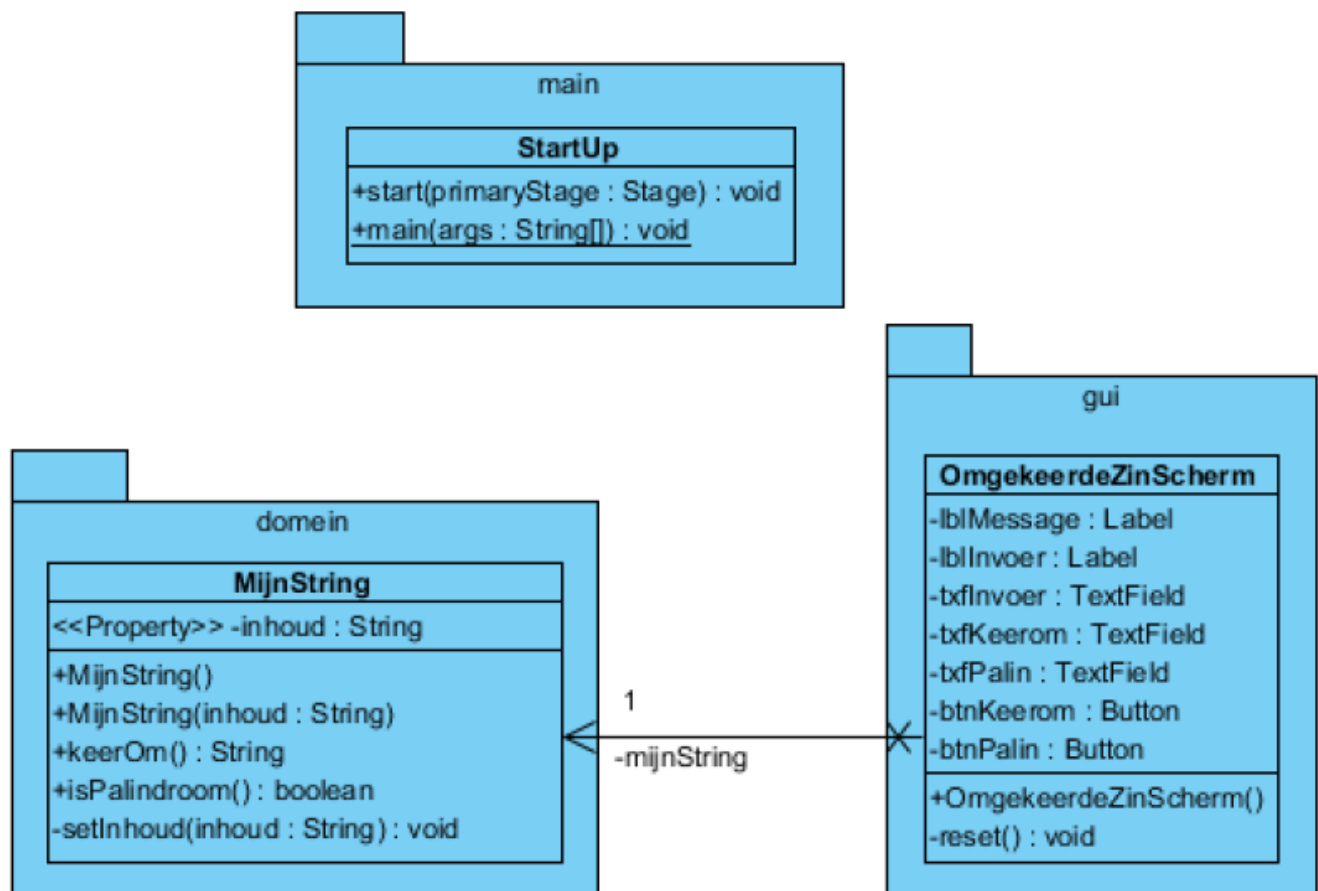
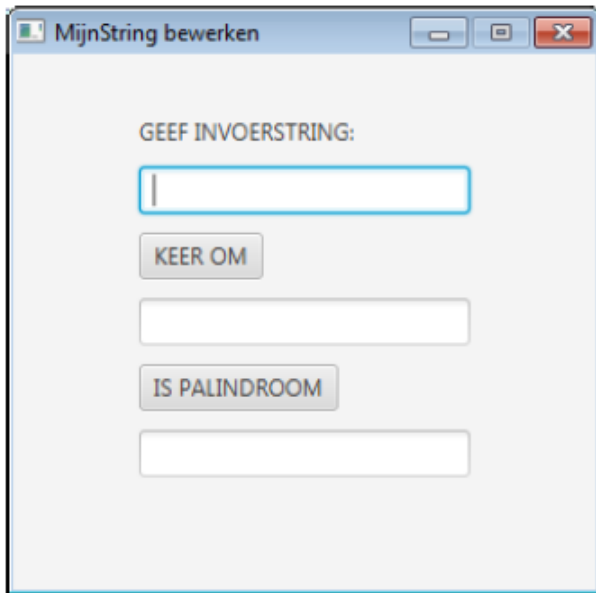
Geeft de inhoud in omgekeerde volgorde terug.

2.2. isPalindroom

Herschrijf de implementatie; gebruik nu een StringBuilder-methode.

2.3. Werk onderstaande gui uit

en gebruik MijnString.



3. Oefening 3

Breid de klasse `MijnString` uit met de methode

3.1. maakWartaal

die 2 karakters (oud en nieuw) als parameter heeft en die in het attribuut elk voorkomen van het eerste karakter vervangt door het tweede karakter.

Het resultaat wordt door de methode teruggegeven, met andere woorden het attribuut zelf wordt

niet gewijzigd!

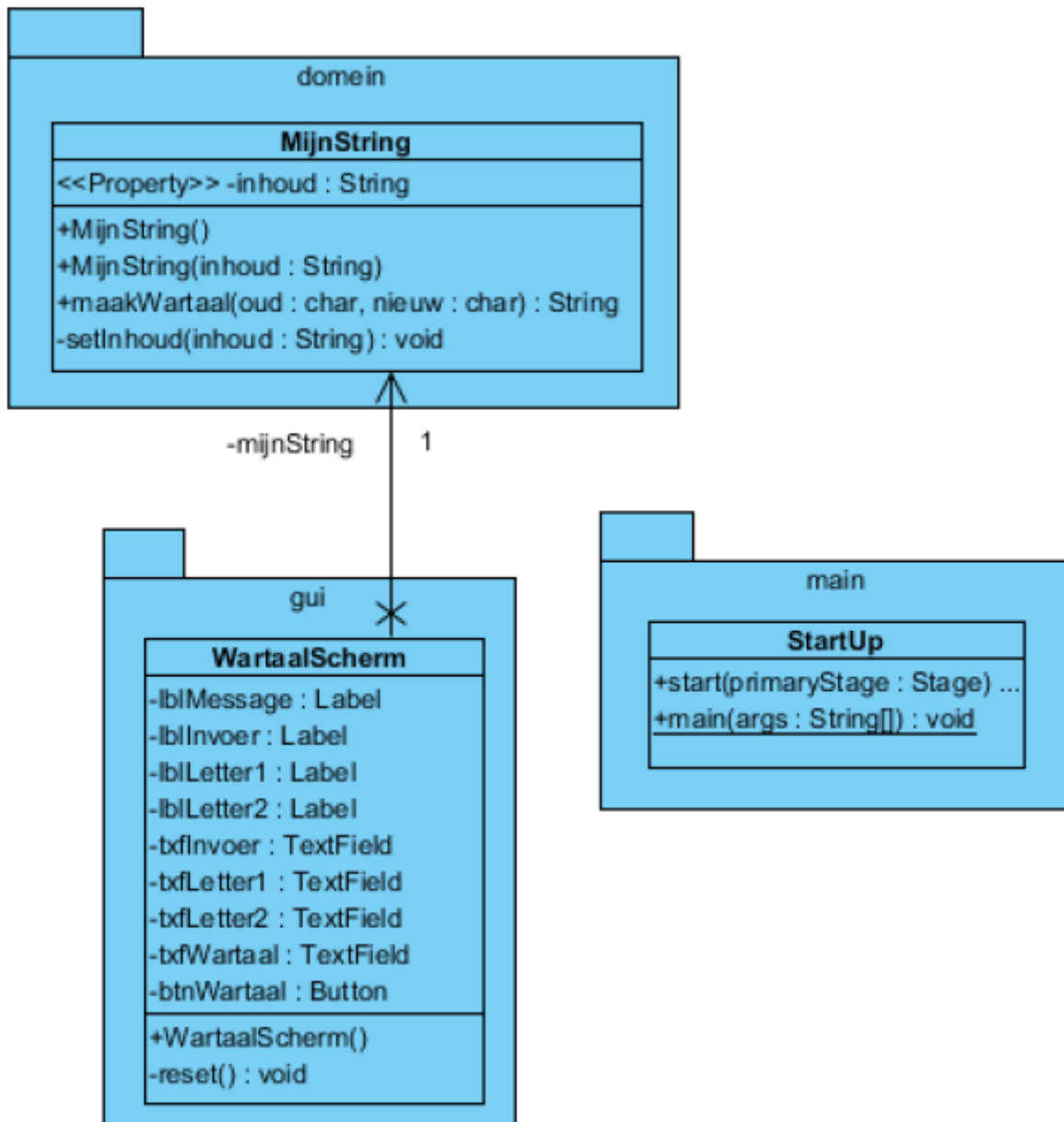
Vb. "lepel". maakWartaal('e','a') geeft lapal

"LEPEL".maakWartaal('e','a') geeft "LAPAL"

The image shows two side-by-side screenshots of a Java Swing window titled "MijnString bewerken".

Left Screenshot: The window contains a label "GEEF INVOERSTRING:" above a text input field that is empty. Below this are two labels, "1ste LETTER:" and "2de LETTER:", each followed by an empty text input field. A button labeled "MAAK WARTAAL" is positioned below these fields. At the bottom of the window is another empty text input field.

Right Screenshot: This screenshot shows the same window after the "MAAK WARTAAL" button has been clicked. The top text input field now contains the string "een eenvoudige test". The "1ste LETTER:" input field contains the character "e", and the "2de LETTER:" input field contains the character "a". The "MAAK WARTAAL" button is now highlighted with a blue border. The bottom text input field now contains the string "aan aanvoudiga tast".



4. Oefening 4: validatie

Valideer een e-mail, huisnummer, postcode en naam.

Gebruik `Validate.java` (Chamilo), specificaties validaties in opgave.

5. Oefening 5: splitsen van teksten

Gebruik `StartUpSplitKleineOef.java` (Chamilo), specificaties in opgave.

6. Oefening 6

Gebruik opnieuw de domeinklasse `MijnString` en

6.1. maakWartaal

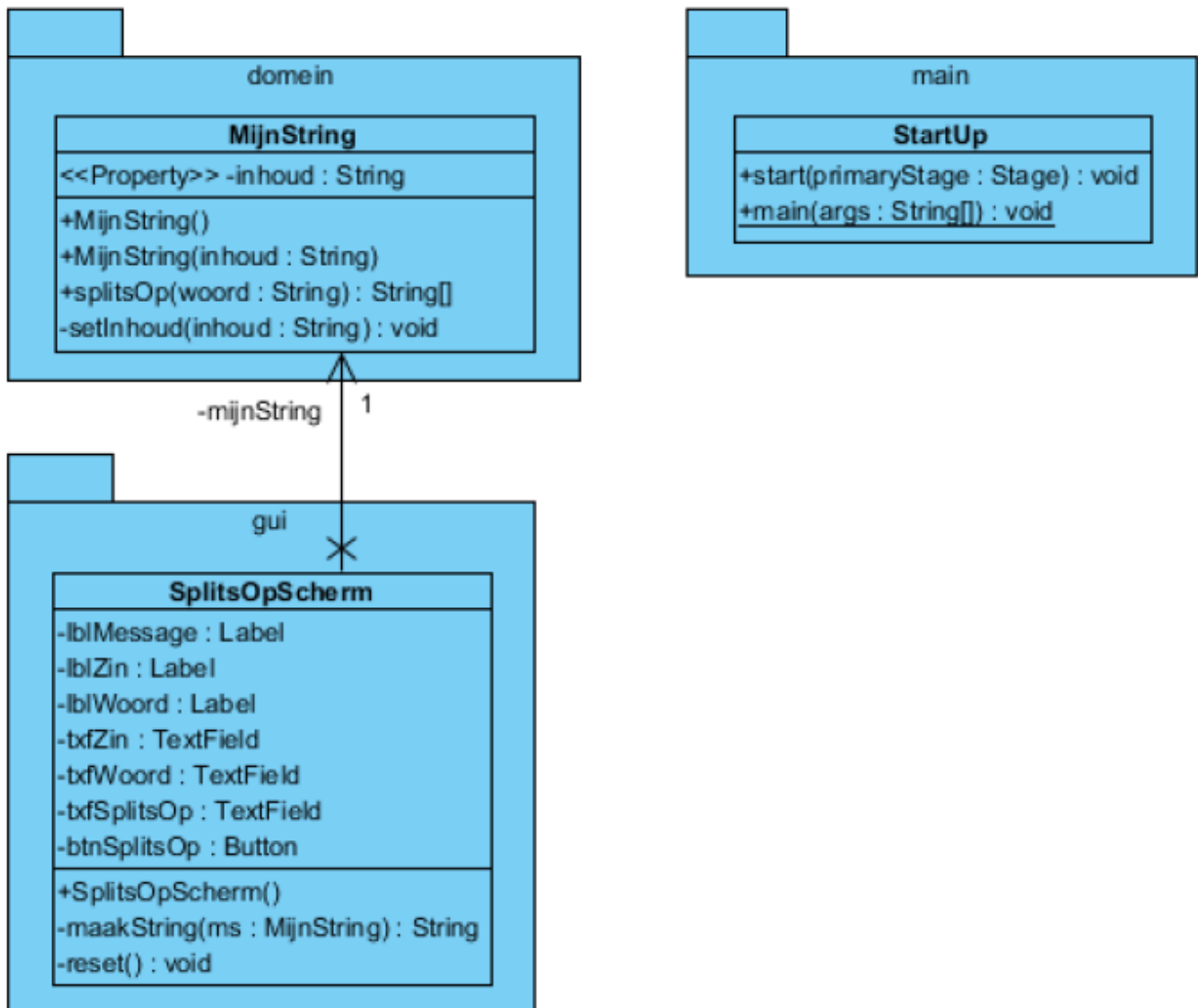
Herschrijf de implementatie; gebruik nu een reguliere expressie

6.2. splitsOp

Deze methode splitst inhoud op volgens het meegegeven woord(=scheidingsteken).



The screenshot shows a Java Swing window titled "MijnString bewerken". Inside the window, there are two text input fields. The first field is labeled "GEEF EEN ZIN:" and contains the text "Oefening dit is een voorbeeld dit is een :". The second field is labeled "GEEF EEN WOORD:" and contains the text "dit". Below these fields is a button labeled "OPSPLITSEN". At the bottom of the window, there is a text area displaying the result of the split operation: "[Oefening, is een voorbeeld, is een zin]".



7. Oefening 7

7.1. PatternMatcher1

specificaties in opgave

7.2. PatterMatcher2

specificaties in opgave