Programmeren II

**ZORG ER VOOR DAT HET PROGRAMMA IN GUI EN CUI NIET CRASHT!**

**Combobox & listview**

# H9-10

super.toString()

getClass().getSimpleName()

if (item instanceof Kluis)

((Rekening)item).getHouder()

KAT( new String[]{"geboortejaar"}),

HOND( new String[]{"naam", "geboortejaar"});

switch (DierSoort.values()[keuze-1]){

case KAT:

# package **exceptions**;

public class **BuitenBereikException extends IllegalArgumentException**{

public BuitenBereikException() {

this("respecteer interval");

}

public BuitenBereikException(String msg) {

super(msg);

}

public BuitenBereikException(Throwable cause) {

super(cause);

}

public BuitenBereikException(String msg, Throwable cause) {

super(msg, cause);

}

}

public NegativeNumberException(String msg, Throwable cause, boolean enableSupression, boolean writeableStackTrace) {

super(msg, cause, enableSupression, writeableStackTrace);

}

**FINALLY**

# Strings

StringBuilder(inhoud).reverse().toString()

geefAllePermutatiesVanDrieMetStringBuilder -> H14oef5

matches éà -> H14oef8

matches 🡪 e-mail, postcode, huisnummer

String[] woorden = zin.split(" \\d+ ");

woorden = zin.split("[^a-zA-Z]+");

Pattern expression =

Pattern.compile("J.\*\\d[0-35-9]-\\d\\d-\\d\\d");

Matcher matcher = expression.matcher(string1);

while (matcher.find())

Character.isLetter(c)

Character.isLowerCase(c)

Character.forDigit(digit, radix));

.append(System.getProperty("line.separator"))

# JavaFX

txfInvoer.setOnAction(new EventHandler<ActionEvent>() {

@Override

public void handle(ActionEvent evt) {

# H7 deel 2

Arrays.sort(bierenArray, new Comparator<Bier>() {

@Override

public int compare(Bier o1, Bier o2) {

Arrays.toString(containers)

Arrays.fill(getallen, 0, 4, 6);

Arrays.fill(getallen, 5, 6, 4);

Arrays.fill(getallen, 7, 9, 8);

//getallen weergeven op het scherm

toonArray("oplossing : 6 6 6 6 6 4 4 8 8 8", getallen);

Arrays.asList(namen).indexOf(naam)

# JavaFX

ObservableList<String> options = FXCollections.observableArrayList(

"Rechthoek",

"Driehoek"

);

keuzeVeelhoek = new ComboBox<>(options);

this.getChildren().addAll(lblTitel, keuzeVeelhoek);

keuzeVeelhoek.getSelectionModel().selectedItemProperty().addListener(new ChangeListener<String>() {

@Override

public void changed(ObservableValue<? extends String> observable, String oldValue, String newValue) {

if (oldValue != null && oldValue.equals("Rechthoek")) {

verwijderRechthoekVelden();

} else if (oldValue != null && oldValue.equals("Driehoek")) {

verwijderDriehoekVelden();

}

if (newValue.equals("Rechthoek")) {

voegRechthoekToe();

} else if (newValue.equals("Driehoek")){

voegDriehoekToe();

}

}

});

ObservableList<String> items = FXCollections.observableArrayList(aannemers);

lvAannemers.setItems(items); // items -> String[]

lvAannemers.getSelectionModel().selectedItemProperty().addListener

(

new ChangeListener<String>()

{

@Override

public void changed(ObservableValue<? extends String> arg0, String arg1, String arg2)

{

int index = lvAannemers.getSelectionModel().getSelectedIndex();

op.update(type, index);

}

}

);