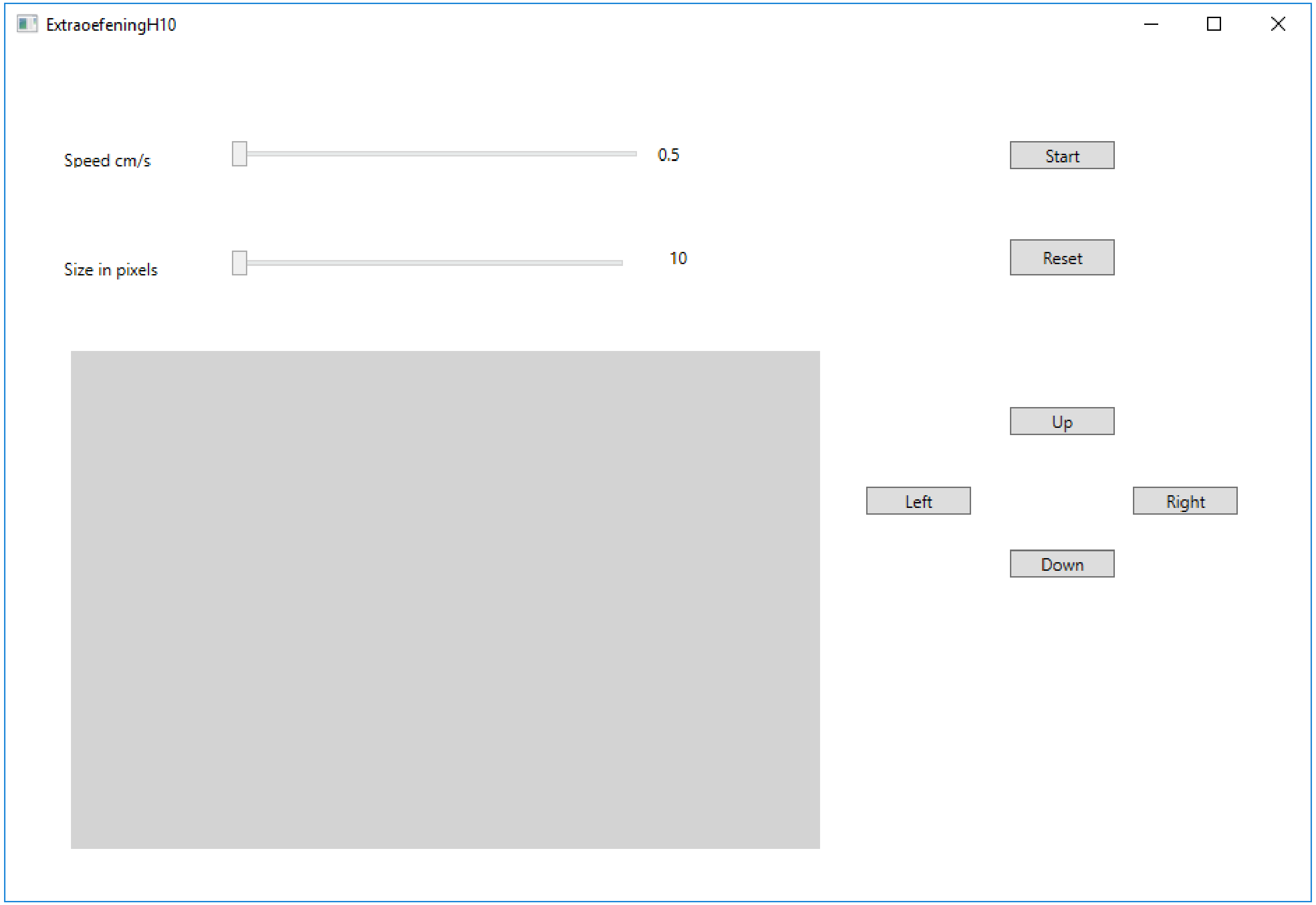
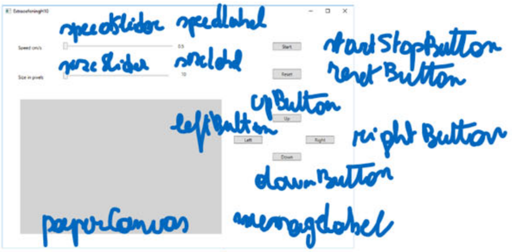
Extra oefening Beetle





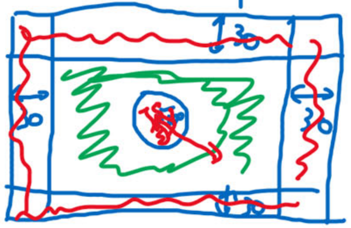
MainWindow

eig
- xstart en ystart
- timer (DispatcherTimer)
- beetle
- startTime

constructor
- aanmaken van beetle zeker meegeven xstart, ystart, canvas
- aanmaken van timer + interval instellen

methoden

- genereren van startpositie : xstart, ystart



bv 5sec 10cm/s → 50 cm

Beetle

eig ~~speed~~ size x, y   body (Ellipse) ~~up~~ ~~right~~   ~~invisible~~ canvas

properties X (update Ellipse) Up
Y (" ") Right
Size (" ") Invisible
Speed

constructor parameters x, y (int), Canvas
- aanmaken van ellipse CreateBeetle
- omhoog en naar rechts
- onzichtbaar maken

methoden
CreateBeetle (aanmaken ellipse, x, y instellen... plaatsen op canvas)

Mainwindow

- eventhandler startStopButton
  - bij start
    - * beetle zichtbaar
    - * timer starten
    - * tekst → stop
    - * sliders disabelen
    - * startTime instellen
  - bij stop
    - * sliders enabelen
    - * timer stoppen
    - * tekst → start
    - * messageLabel aanpassen
      - ↳ stoptime bepalen
        afgelegde afstand
- eventhandler sizeSlider
  - property size waarde geven
  - timerInterval instellen
- eventhandler speedSlider
  - property Speed waarde geven
    timerInterval instellen

Beetle

----> ComputeDistance (startTime, stopTime)

vb grootte 12 → per tik 0,12 cm afgelegd

speed 5cm/s

0,12 cm → 1 tik

1cm → $\frac{1}{0{,}12}$ tikken

5cm → $\boxed{\frac{5}{0{,}12}}$ tikken / seconde

10

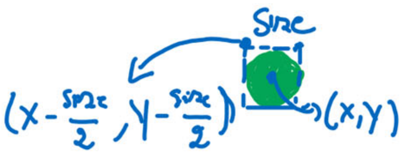
tijd tussen 2 tikken in milliseconden

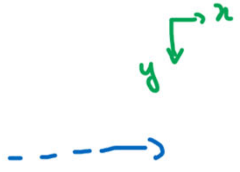
$$1000 / (\overset{speed}{\boxed{5}} / \overset{grootte/100}{\boxed{0{,}12}})$$

$$= 1000 / (speed / grootte * 100)$$

Mainwindow

- eventhandler timer
  kever verplaatsen





Beetle

opm: Speed=0
geen beweging

- ChangePosition( )

Up $\xrightarrow{j}$ Y -= 1
Up $\xrightarrow{N}$ Y += 1

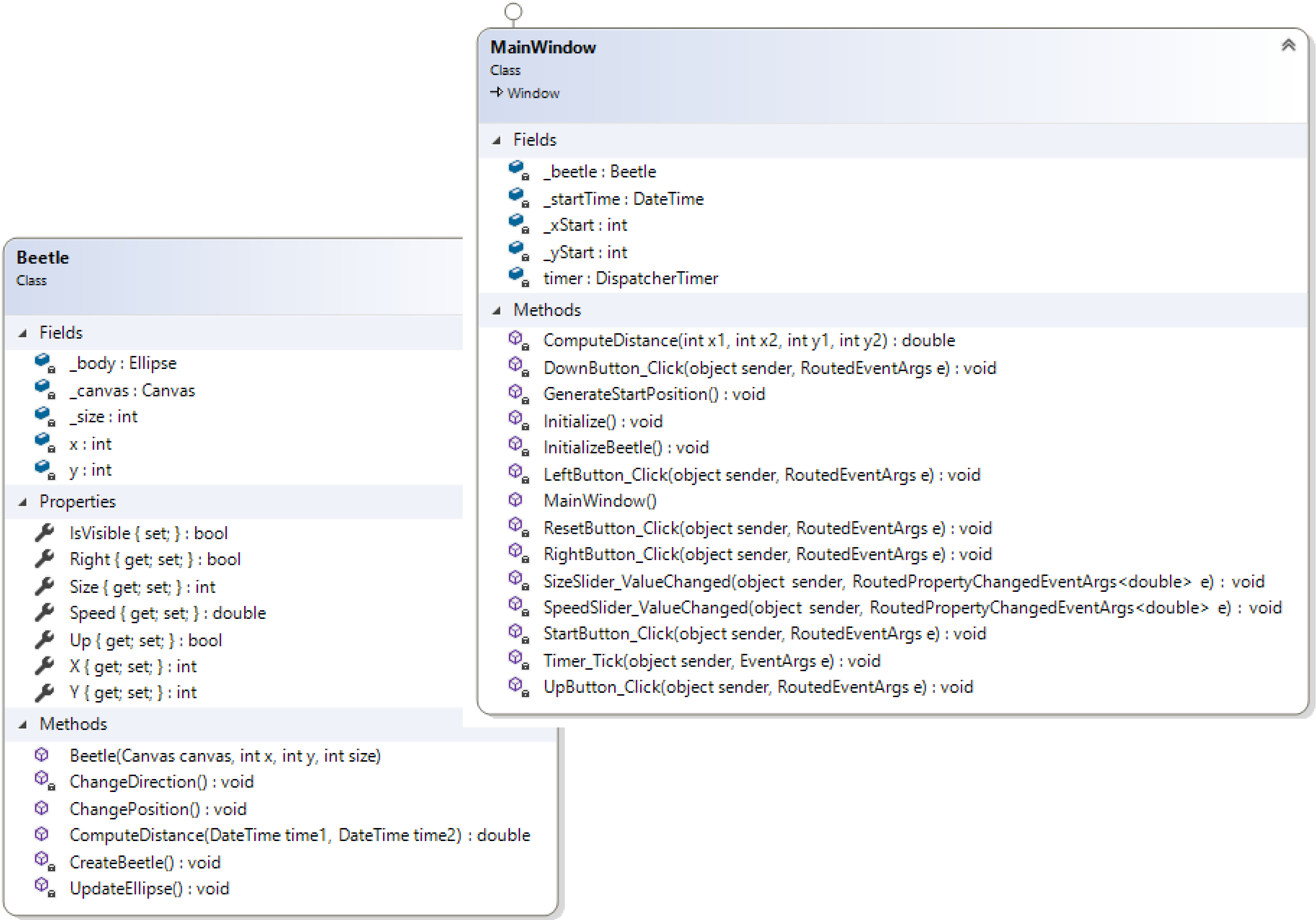
Right $\xrightarrow{j}$ X += 1
Right $\xrightarrow{N}$ X -= 1

controle uitvoeren of je aan de rand bent?
↳ ja ChangeDirection( )

- UpDate Ellipse

- eventhandler upButton
- downButton
- right Button
- left Button

- event handler reset Button
  * beetle onzichtbaar maken
  * sliders minimale waarde
  * sliders enabelen
  * messageLabel leegmaken
  * nieuwe positie genereren
  * timer stoppen



MainWindow
Class
→ Window

Fields
- _beetle : Beetle
- _startTime : DateTime
- _xStart : int
- _yStart : int
- timer : DispatcherTimer

Methods
- ComputeDistance(int x1, int x2, int y1, int y2) : double
- DownButton_Click(object sender, RoutedEventArgs e) : void
- GenerateStartPosition() : void
- Initialize() : void
- InitializeBeetle() : void
- LeftButton_Click(object sender, RoutedEventArgs e) : void
- MainWindow()
- ResetButton_Click(object sender, RoutedEventArgs e) : void
- RightButton_Click(object sender, RoutedEventArgs e) : void
- SizeSlider_ValueChanged(object sender, RoutedPropertyChangedEventArgs<double> e) : void
- SpeedSlider_ValueChanged(object sender, RoutedPropertyChangedEventArgs<double> e) : void
- StartButton_Click(object sender, RoutedEventArgs e) : void
- Timer_Tick(object sender, EventArgs e) : void
- UpButton_Click(object sender, RoutedEventArgs e) : void

Beetle
Class

Fields
- _body : Ellipse
- _canvas : Canvas
- _size : int
- x : int
- y : int

Properties
- IsVisible { set; } : bool
- Right { get; set; } : bool
- Size { get; set; } : int
- Speed { get; set; } : double
- Up { get; set; } : bool
- X { get; set; } : int
- Y { get; set; } : int

Methods
- Beetle(Canvas canvas, int x, int y, int size)
- ChangeDirection() : void
- ChangePosition() : void
- ComputeDistance(DateTime time1, DateTime time2) : double
- CreateBeetle() : void
- UpdateEllipse() : void