YOLT! Programming Language

Geschreven door Mathieu te Stroete.

Legenda

[Legenda: 2](#_Toc101702921)

[Kenmerken van de taal: 3](#_Toc101702922)

[Hello world in mijn taal: 3](#_Toc101702923)

[Werken met Jolt. 3](#_Toc101702924)

[Voorbeeldprogramma’s 5](#_Toc101702925)

[Fibonacci 5](#_Toc101702926)

[Guess the number 6](#_Toc101702927)

[OOP system. 6](#_Toc101702928)

# YOLT!

## Samenvatting van de taal

De officiële naam van de taal is YOLT! (Yeah, orcs love typing!). In dit verslag haal ik het uitroepteken aan het eind weg voor leesbaarheid. YOLT is een taal die het beste van python combineert met het beste van Java!

## Duidelijke kenmerken van YOLT

* YOLT is een statische taal. Dit betekent dat je al je variabelen van tevoren aan moet geven.
* YOLT ondersteund variabelen.
* YOLT ondersteund methodes met parameters.
* YOLT ondersteund geen classes
* YOLT gebruikt brackets voor scoping.
* YOLT vereist een ; aan het eind van een regel code.
* YOLT ondersteund comments. //\* dit is een comment
* YOLT ondersteund user output met gebruik van SPEAK();
* YOLT ondersteund user input met gebruik van TALK();
* YOLT ondersteund het afsluiten van de applicatie op elk moment met DIE();

## Wat wordt ondersteund door YOLT

### Datatypes

* Boolean (BOOL)
* Int (NUMBER)
* String (WORDS)
* GC (Gold coins)

### Operators met priority van laag naar hoog

* + , -
* /
* \*
* % (Modulo)
* ^ (3^3 = 3\*3\*3)
* ()

### Logische operators

* ==
* =!=
* >
* <
* &&
* ||

# Code voorbeelden YOLT

## Hello world

**SPEAK(**"**Hello World**"**);**

# Werken met Jolt.

Een variabele aanmaken in Jolt gaat door eerst aan te geven local, hierna kan je deze een naam geven, en daarna een waarde, zoals in het voorbeeld hieronder:

**BOOL value = FALSE;**

Operations op variabelen kunnen als volgt:

**Value = value + 8; (16).**

**Value = value \* 8; (64).**

**Value = value / 8; (1).**

**Value = value – 8; (0).**

**Value = value ^ 3; (512), (8\*8\*8)**

Logical operations kunnen als volgt:

**(Value == 8); (returned true als ze hetzelfde zijn).**

**(Value =!= 8); (returned false als ze hetzelfde zijn).**

**(Value == 8 && value2 =!= 4); (returned true als ze hetzelfde zijn).**

**(Value > 8);**

**(Value < 8);**

**(Value >= 8);**

**(Value <= 8);**

De taal ondersteund een if() constructie, else if(), en else

**IF(value < 8):{**

**}**

**ELSE IF (value < 6 ): {**

**}**

**ELSE: {**

**}**

De taal ondersteund 2 verschillende loops, Repeat en Repeat until. Daarnaast ondersteund het break;

**REPEAT(condition):**

**{**

**If ()**

**{**

**}**

**}**

**REPEAT:**

**{**

**If ()**

**{**

**BREAK;**

**}**

**}**

**WHILE(condition);**

De taal ondersteund methods:

**FUNCTION Start(Param){**

**}**

De taal ondersteunt input van de user via een scanner.

**WORDS message = TALK();**

De taal ondersteunt output van de user via print.

**SPEAK(“Tekst voor de gebruiker!”);**

# Voorbeeldprogramma’s

## Basic Functionality

FUNCTION Fibonacci()

{

NUMBER num1 = 0;

BOOL lies = FALSE;

SPEAK(num1);

SPEAK(num2);

REPEAT(num3 < 100):

{

num3 = num1 + num2;

num1 = num2;

num2 = num3;

SPEAK(num3);

}

SPEAK("DONE");

}

## Fibonacci

FUNCTION Fibonacci()

{

NUMBER num1 = 0;

NUMBER num2 = 1;

NUMBER num3 = 0;

SPEAK(num1);

SPEAK(num2);

REPEAT(num3 < 100):

{

num3 = num1 + num2;

num1 = num2;

num2 = num3;

SPEAK(num3);

}

SPEAK("DONE");

}

## FizzBuzz

FUNCTION FizzBuzz()

{

NUMBER i = 0;

REPEAT(i < 100):

{

IF(I % 5 == 0 && I % 7 == 0)

{

SPEAK("FizzBuzz")

}

ELSE IF (I % 5 == 0)

{

SPEAK("Fizz")

}

ELSE IF (I % 7 == 0)

{

SPEAK("Buzz")

}

ELSE

{

SPEAK(I);

}

I += 1;

}

}

## 

## Guess The Number!

FUNCTION GuessTheNumber()

{

BOOL found = FALSE;

SPEAK("HELLO, WELCOME TO GUESS THE NUMBER!");

NUMBER amountOfGuesses = 0;

NUMBER toGuess = RANDOM(100);

REPEAT(amountOfGuesses < 7 && FOUND == FALSE):

{

NUMBER guess = TALK();

IF(guess > toGuess):

{

SPEAK("Your guess is too high!");

}

ELSE IF(Guess < toGuess):

{

SPEAK("YOUR GUESS IS TOO LOW!");

}

ELSE:

{

SPEAK("You guessed the number!");

FOUND = TRUE;

}

}

IF(FOUND =!= TRUE):

{

SPEAK("You have failed guess the number, better luck next time!");

}

}

## Calculator

FUNCTION GuessTheNumber()

{

SPEAK(“HELLO, WELCOME TO THE CALCULATOR!”);

}

FUNCTION add(NUMBER number, NUMBER number2)

{

Number + number2;

RETURN

}

FUNCTION subtract(NUMBER number, NUMBER number2)

{

}

FUNCTION POWER(NUMBER number, NUMBER number2)

{

}