

Grundlegende Einführung in die Sprache und ihre Konzepte bzw. Rust is awesome

Philipp Schmid Andrea Matecsa

April 11, 2017

Contents

1	Introduction	1
1.1	Was ist Rust	1
1.2	Besonderheiten von Rust	1
2	Pattern Matching	2
2.1	was ist PM?	2
2.2	Beispiel	2
3	Mutability	2
4	Higher Order Functions	2
5	Comparison with other Languages	2
5.1	Ocaml	2
5.2	andere	2
6	Discussion/Conclusion	2

1 Introduction

1.1 Was ist Rust

Warum Rust? Was spricht für Rust? erstellt von mozilla

1.2 Besonderheiten von Rust

-Focus auf Sicherheit -ähnelt c und ocaml

2 Pattern Matching

2.1 was ist PM?

Pattern Matching ist matchen auf ein bestimmtes Pattern

2.2 Beispiel

```
1 let x=4;
2 match x{
3 1=>!println("x is 1")
4 2=>!println("x is 2")
5 3=>!println("x is 3")
6 4=>!println("x is 4")
7 _=>!println("x is anything else")
8 }
9 wie ocaml match a with x::xs->xs
```

3 Mutability

Was ist mutability? wie wird es umgesetzt? vorteile nachteile?

4 Higher Order Functions

Was sind HOF? Warum HOF? Beispiel, Gründe für HOF breitere Wiederverwendung möglich bekannte higher order functions bspw. fold Beispiel

5 Comparison with other Languages

Rust ist besser als alles andere

5.1 Ocaml

5.2 andere

6 Discussion/Conclusion

Rust ist supergeil

References

- [1] Official Rust Webpage, *L^AT_EX*: Documentation about Rust, <https://www.rust-lang.org/en-US/documentation.html> accessed on the 10.4.2017