

Um bei dem Racket Interpreter neue Funktionen hinzuzufügen, kann man am einfachsten dieser Anleitung folgen.

Schritt 0:

Führt alle „RacketEvaluationTests.java“ Test aus, um zu kontrollieren, dass alles funktioniert, bevor ihr Änderungen durchführt.

Schritt 1:

Raussuchen der Funktion in Racket, die man hinzufügen möchte.

Schritt 2:

Öffnen von „RacketEvaluationTests.java“

Schritt 3:

Nun schreibt man einen Test für die Funktion, die man hinzufügen möchte. Dabei sollte man darauf achten, dass alles Wichtige getestet wird. Am besten checkt man die Antworten auch noch einmal in Dr. Racket, weil das tatsächliche Ergebnis nicht immer das Intuitive ist.

Schritt 4:

Man geht in das Paket „eu.qped.racket.functions“ und sucht sich den Passenden Typen für seine Funktion aus. (Bsp. eu.qped.racket.functions.numbers)

Schritt 5:

Man erstellt in dem Passenden Paket eine Neue Klasse, die man Sinnvoll benennt

Schritt 6:

Man extends in der neuen Klasse die Klasse Expression (aus dem Paket eu.qped.racket.test)

Schritt 7:

Man fügt die Folgenden Methoden ein:

```
@Override
public String evaluate(Expression e) {
    return evaluate(e.getRest(super.getId()));
}

@Override
public String evaluate(List<Expression> list) {
    return null; // TODO The Evaluation;
}

@Override
public String toString() {
    return "(TODO Add Function name here) " + " (" + super.getId() + ")";
}
```

Schritt 8:

Man passt die 2 mit TODO beschriebenen Teile (in toString() und evaluate(List<Expression> list)) an.

Schritt 9:

Man öffnet die Klasse DrRacketInterpreter.java

Schritt 10:

Man geht in die Klasse „private static boolean typeName(String valueString, Expression expression)“

Nun scrollt man zu dem passenden Abschnitt, also On Numbers oder On Boolean usw. und fügt am Ende folgendes ein:

```
if (valueString.compareTo("value=\"TODO the Function name from Racket\"")
== 0) {    //TODO The function Name as a Comment
    expression.addPart(new //TODO function you created);
    return true;
}
```

Schritt 11:

Passt die drei TODOS an.

Schritt 12:

Führt euren Test durch und kontrolliert damit, dass alles richtig funktioniert

Schritt 13:

Führt alle „RacketEvaluationTests.java“ Test aus um zu kontrollieren, dass ihr nicht kaputt gemacht habt.