

Übungen zur Vorlesung  
**Spezielle Gebiete zum Software Engineering**  
Sommersemester 2015

**Aufgabenblatt 1: EcmaScript6, Typescript, Dart**

**Aufgabe 1.1: EcmaScript 6 (Javascript 2015) Features mit Traceur oder Babel (5 Punkte)**

Um Software, die in EcmaScript 6 programmiert wurde, in aktuellen Umgebungen lauffähig zu machen, muss diese derzeit zunächst noch nach EcmaScript 5 „statisch transpiliert“ werden. Dies geschieht sinnvollerweise vor dem Laden der Software. Zur Bequemlichkeit kann die Software auch „dynamisch transpiliert“ werden. Auf der Webseite

<https://github.com/google/traceur-compiler/wiki/Getting-Started>

bzw.

<https://babeljs.io/docs/usage/browser/>

wird beschrieben, wie EcmaScript 6-Code dynamisch nach EcmaScript 5-Code konvertiert werden kann. Erstellen Sie mit Hilfe von *Traceur* ODER *Babel* für neue Sprachfeatures von EcmaScript 6, die auf der Webseite

<https://github.com/google/traceur-compiler/wiki/LanguageFeatures>

dargestellt werden, jeweils ein Beispiel und zeigen Sie, dass dieses in aktuellen Browsern abläuft. Sie sollten Beispiele für alle „default“-Sprachfeatures, Symbole sowie für asynchrone Funktionen zeigen können.

**Aufgabe 1.2: Refactoring einer Javascript-Anwendung nach EcmaScript 6 (5 Punkte)**

Konvertieren Sie das in EcmaScript 5 programmierte Puzzle-Spiel, das auf der Webseite

<http://www.sitepoint.com/image-manipulation-with-html5-canvas-a-sliding-puzzle-2/>

zu finden ist, nach EcmaScript 6. Benutzen Sie bei der Konvertierung so viele neue ES6-Sprachfeatures wie möglich. Zeigen Sie, dass die Anwendung in aktuellen Browsern läuft. Transpilieren Sie Ihre Software statisch oder dynamisch.