

Einführung in Web Engineering Sommersemester 2015

Übung 2

Aufgabenbeschreibung:

Ziel dieser und der nächsten Übung ist es, den Schiffsverkehr in Bonn zwischen Nord- und Südbrücke auf einer Karte im Web darzustellen. Dazu verwenden wir in dieser Übung das HTML5 Canvas Element um Schiffe auf einen Kartenausschnitt zu zeichnen. In der nächsten Übung werden Sie live Daten nutzen um den Schiffsverkehr in Echtzeit darzustellen.

Im BSCW finden Sie die Datei „resources.zip“, welche folgende Dateien enthält:

- js/aisData.js: enthält einige empfangene Positionsdaten
- js/ais.js: enthält den Rumpf für die Programmlogik
- js/test.js: Ein Test, mit dem Sie einen Teil Ihrer Implementierungen testen können.
- images/map.png: Kartenausschnitt
- images/mapWithShips.png: Wenn Sie alles richtig gemacht haben, sollte ihr Canvas so aussehen wie auf diesem Bild.

Wichtige Anmerkungen:

- Die Datei js/test.js ist **nicht** ausreichend um Ihre gesamte Logik zu testen. Es reicht also nicht, dass der Test erfolgreich ist um die Übung zu bestehen. Sie soll Ihnen einfach bei der Entwicklung helfen. Sie können diese Datei auch erweitern oder den Code auskommentieren wenn Sie möchten.
- Für die Javascript Aufgaben finden Sie einen Kommentar in ais.js (z.B. für Aufgabe 2 den Kommentar „// Aufgabe 2“), der Ihnen zeigt wo Sie die Programmlogik unterbringen müssen.
- Ändern Sie keine existierenden Namen von Funktionen, Properties, usw.. Selbstverständlich dürfen Sie Ihre eigenen Variablen nennen wie Sie möchten.
- Die Datei ais.js definiert Werte für Dimensionen und Farben, nutzen Sie diese in Ihrem Code.
- **Legen Sie keine weiteren Variablen im Global Scope an!**

Sollten Sie etwas nicht verstehen oder Probleme haben, melden Sie sich.

Aufgabe 1: Vorbereitungen, Canvas Tag und Javascript Dateien

Öffnen Sie eclipse und erstellen Sie zunächst eine Kopie des Projektes aus der letzten Übung, nennen Sie diese Kopie „Uebung2“ (copy & paste innerhalb von eclipse). Integrieren sie folgendes Tag in die Seite:

```
<canvas id="map" height="800" width="766"></canvas>
```

Erstellen Sie nun einen Ordner „js“ unter WebContent und kopieren Sie die bereitgestellten Dateien „aisData.js“, „ais.js“ und „test.js“ in diesen.

Lassen Sie nun diese Dateien in der angegebenen Reihenfolge in der Seite laden, indem Sie die entsprechenden Tags in die Seite integrieren.

Aufgabe 2: Color Iterator

Ziel ist es, einen Iterator für das Array mit Namen AISShipColors zu erstellen. Der Iterator gibt beim ersten Aufruf die erste Farbe zurück, beim zweiten die zweite, usw. Nach der letzten Farbe wird wieder vorne angefangen.

Implementieren Sie die Funktion so, dass

- die Funktion eine Funktion zurückgibt.
- die zurückgegebene Funktion bei jedem Aufruf die nächste Farbe nach oben beschriebenen Muster zurückgibt.
- mehrere Iteratoren gleichzeitig genutzt werden können und alle nach dem beschriebenen Muster arbeiten.

Aufgabe 3: Ship Konstruktor

Der Konstruktor bekommt 4 Werte übergeben. Weisen Sie dem neu erstellen Objekt **alle** diese Werte unter gleichnamigen Properties zu. Heißt der Parameter z.B. „mmsi“, muss das aus dem Konstruktor resultierende Objekt also eine Property namens „mmsi“ mit dem Wert des Parameters haben.

Aufgabe 4: ShipList filter Methode

Objekte welche wir mit dem ShipList Konstruktor erstellen, besitzen eine Property mit Namen „list“ (ein Array). Ziel der filter Methode ist es, die „list“ anhand einer übergebenen Funktion zu filtern.

Der Parameter „filterFunction“ ist eine Funktion mit **einem Parameter** und einem **Boolean Rückgabewert**. Die filter Methode soll nun jedes Element mittels der filterFunction auf Verbleib in der Liste prüfen. Gibt die filterFunction für ein Element aus „list“ false zurück, muss es aus der Liste entfernt werden.

Sie dürfen die Property „list“ durch ein anderes Array ersetzen.

Aufgabe 5: Karte zeichnen und Zeichen der Schiffe anstoßen

Kopieren Sie die Datei map.png nach WebContent/images/map.png. Nutzen Sie das Canvas um die Bilddatei zu rendern. Lesen Sie hierzu:

https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Canvas_tutorial/Using_images

Insbesondere der Abschnitt „Creating images from scratch“ ist für Sie interessant.

Ist das Bild auf dem Canvas gerendert, rufen Sie „AIS.drawAllShips(canvas);“ auf, um alle Schiffe rendern zu lassen. Damit dies auch funktioniert, lösen Sie noch Aufgabe 6.

Aufgabe 6: Schiffe zeichnen

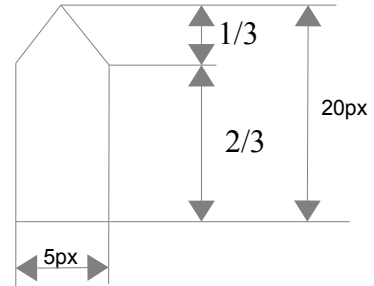
Für diese Aufgabe finden Sie eine gute Beschreibung der benötigten API hier:

https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Canvas_tutorial/Drawing_shapes
https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Canvas_tutorial/Transformations

Implementieren Sie die draw Methode. Zeichnen Sie das Schiff auf das übergebene Canvas.

Ein Schiff soll folgende Dimensionen haben:

- 5 Pixel breit (bereits definiert: shipWidth)
- 20 Pixel hoch (bereits definiert: shipHeight)
- nach 2/3 der Länge soll das Schiff vorne spitz zulaufen



Der Schiffskörper muss mit der übergebenen Farbe befüllt werden, die Umrandung sollte schwarz bleiben.

Die die Property „courseOverGround“ gibt an, in welche Richtung das Schiff zeigt. 0 ist Norden, 90 Osten, 180 Süden, 270 Westen.

Die Grenzen der Karte sind in den Variablen „top“, „bottom“, „left“ und „right“ definiert. Wir verzichten auf die korrekte (Mercator-) Projektion und positionieren die Schiffe relativ zu den Grenzen der Karte.

Beispiel: Wäre „left“ 1000 und „right“ 1500, würden wir einen Wert mit 1250 in die vertikale Mitte positionieren.

Positionieren Sie die Mitte des Schiffskörpers auf die angegebenen Koordinaten.

Die Dimensionen der Karte in Pixel sind als Variablen „height“ und „width“ in Pixeln vordefiniert.

Viel Erfolg!