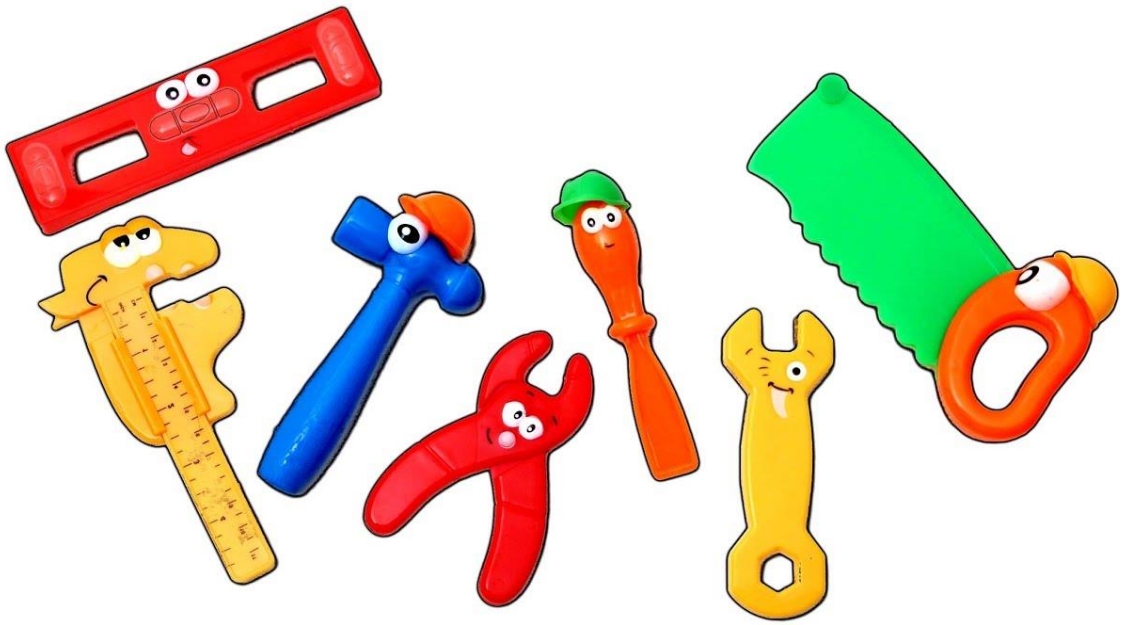
ZOEK DE SCHATKIST

In dit spel gaan we een schatkist zoeken. Ergens op de afbeelding hierboven is een schatkist begraven. Maar je weet niet waar. De bedoeling is dat bij elke mouse click je kan zien hoe dichtbij je bij de schatkist bent. En als je de schatkist vind, dan krijg je een leuke pop-up te zien. Om het spel te ontwerpen hebben we de volgende onderdelen(tools) nodig:



- We moeten eerst een **web pagina** maken.
- We hebben een **schatkist** nodig op de kaart
- We hebben een **click event handler** nodig. De click handler zorgt voor de volgende acties:
  - ✓ Een teller die het aantal clicks bij houdt
  - ✓ Die de afstand berekent vanaf de click en de schatkist
  - ✓ Hoe dichtbij de gok is van de gebruiker
  - ✓ Feliciteert de gebruiker als hij/zij de schatkist heeft gevonden.

We beginnen met de html pagina:

```
<!DOCTYPE html>
<html>
<head>
    <title>Zoek de schatkist!</title>
</head>
<body>
    <h1 id="heading">Zoek de schatkist!</h1>
     <img id="map" width=800 height=800
            src="_images/treasuremap.png">
     <p id="distance"></p>
    <script src="https://code.jquery.com/jquery-2.1.0.js"></script>
    <script>
        // Game code komt hier:

    </script>
</body>
</html>
```

Een random locatie

We moeten nu een random(willekeuring) locatie vinden waar de schatkist zich kan bevinden. Dit doe we door middel van twee random nummers te genereren tussen de 0 en 799. Waarom twee nummers en waarom tussen de 0 en 799? Bedenk nu zelf waarom deze getallen!!
We kunnen een random nummer generen doormiddel van onderstaande functie:

```
var getRandomNumber = function (size) {
return Math.floor(Math.random() * size);
};
```

Om nu twee random getallen te generen tussen de 0 en 799 schrijven we dit:

```
var width = 400;
var height = 400;
 var target = {
x: getRandomNumber(width),
y: getRandomNumber(height)
};
```

Typ nu dit in je html pagina:

```
// Genereer nu een getal van 0 tot size
        var getRandomNumber = function (size) {
            return Math.floor(Math.random() * size);
        };

        var width = 400;
        var height = 400;
        var target = {
            x: getRandomNumber(width),
            y: getRandomNumber(height)
        };
```

Opdracht:
Hoe krijg je nu de getallen x en y in de console te zien?

De click handler
De gebruiker kan nu op de kaart klikken. Dus een click event handler is hiervoor nodig. Dit kan met onderstaande functie:

```
$("#map").click(function (event) {
// Click handler code goes here
});
```

Aangezien we het aantal kliks bijhouden dat de gebruiker nodig heeft om de schatkist te vinden. Hebben we een teller nodig. Als er nog niet geklikt is, dan staat de teller uiteraard op 0. Dit noteren we dus als volgt:

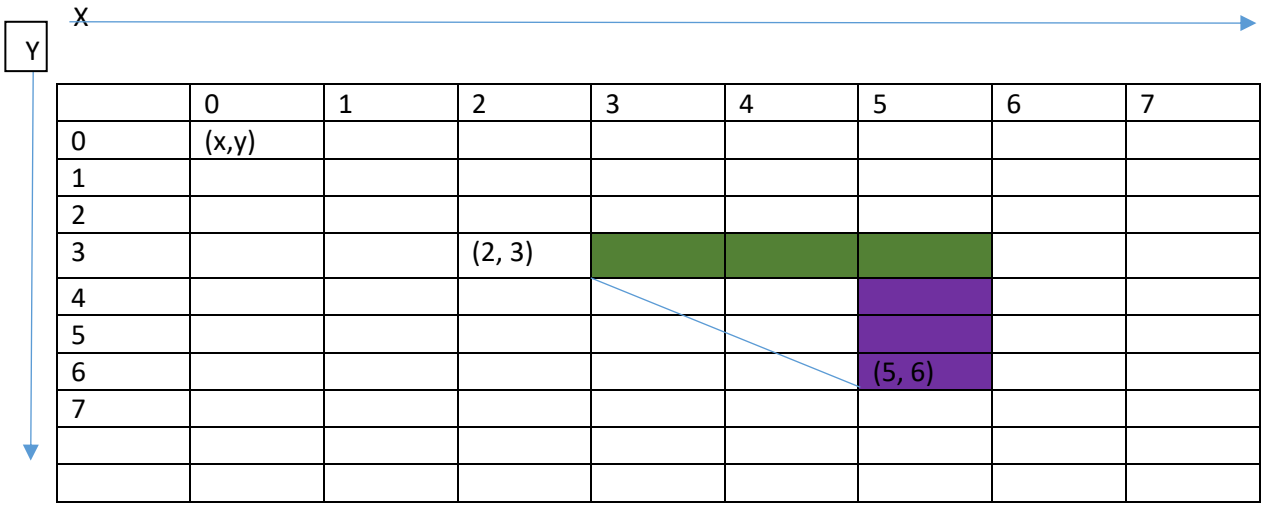
```
var clicks = 0;
```

En binnen de click event noteren we dit:

Clicks++; Dit betekent dat bij elke click de teller telkens met één wordt verhoogd.

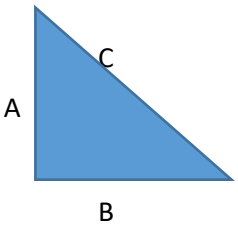
De afstand van de schatkist en de click van de gebruiker

Hiervoor hebben we nu een beetje wiskunde voor nodig. We beginnen met een coördinaten stelsel. We weten dat een scherm is opgebouwd uit pixels. En X en Y coördinaten.
Stel we hebben het volgende coördinaten stelsel:



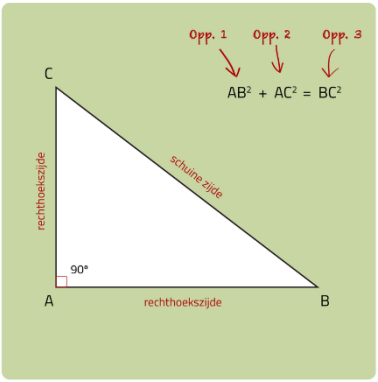
Dan is de lengte van het groene gedeelte: x1 – x2 = 5-2 = 3 = x
En het paarse gedeelte is dus: y1 – y2 = 6-3 = 3 = y

De afstand(de blauwe lijn) tussen de punten x en y wordt nu berekend met behulp van de stelling van Pythagoras:



$A^2 + B^2 = C^2$. Als voorbeeld kunnen we nemen: a=3, b=4. Dan geldt dus: $3*3 + 4*4 = 25 = C^2$. Nu is wortel trekken het tegenovergestelde van kwadrateren. En geldt dus: $\sqrt{25} = 5$. De afstand van C is dus 5.

En denk er aan dat een schuine zijde altijd korter is dan twee rechte zijdes, zie figuur beneden:



Opdracht:
Met nu al deze informatie. Probeer nu de functie te schrijven die ervoor zorgt dat de afstand tussen de gebruikers klik met de mouse en de schat te berekenen. Hier alvast het geraamte van de functie:

```
var getDistance = function (event, target) {

            //Schrijf hier je eigen invulling

        };
```

Vertel de speler hoe dichtbij je bent bij de schat
Nu we de afstand weten van de speler's klik en de schat, moeten we nog de speler vertellen hoe ver, of hoe dichtbij hij bij de schat is. We kunnen de speler dit bijvoorbeeld vertellen met tekst:

- Afstand < 320 = Koud
  Dat de speler ver verwijdert is van de schat
- Afstand < 160 = minder Koud
  Dat de speler wat dichterbij is bij de schat
- Afstand < 80 = warm
  Dat de speler dichtbij de schat is
- Afstand < 40 = heel warm
- Afstand < 20 = kokend warm

Opdracht
Hoe zou je nu een eigen functie hiervoor schrijven?

```
var getDistanceHint = function (distance) {

       };
```

Vul nu deze code:

```
var distance = getDistance(event, target);
var distanceHint = getDistanceHint(distance);
$("#distance").text(distanceHint)
```

In de klik function:

```
$("#map").click(function (event) {
// Click handler code komt hier
});
```

Vertel de speler dat hij/zij gewonnen heeft

Wat over blijft is dat we de speler moeten vertellen dat hij/zij gewonnen heeft. Dit kunnen we voor elkaar krijgen doormiddel van het volgende code fragment:

```
if (distance < 8) {
alert("De schatkist gevonden in:  " + clicks + " clicks!");
}
```

Opdracht

Vul nu dit stukje code:

```
if (distance < 8) {
alert("De schatkist gevonden in:  " + clicks + " clicks!");
}
```

In de goede methode

YOE HOE!!!!!!!!!!!!!!

Ok!! We hebben nu alle functionaliteit om het spel te spelen. Probeer nu zelf om het spel te spelen. Gaat het goed? Of werken er nog functies niet?

Probeer nu zelf er achter te komen wat er fout gaat en los het zo nodig op.

Extra Opdrachten

1. Zorg nu voor nog meer berichten als de speler goed zit of juist fout zit. Bijvoorbeeld: Heel heel erg koud, of : zo warm als lava
2. Programmeer een limiet op de kliks dat een gebruiker mag klikken. Bijvoorbeeld dat een speler maximaal 7 keer mak klikken. Na de 7 keer mag de speler niet meer klikken en komt er een bericht naar voren: game over
3. Laat de gebruiker zien hoeveel mouse klikken hij/zij nog heeft. Bijvoorbeeld u heeft nog 3 van 7 klikken over