- **Explain the difference between the == operator and the === operator.**

When you would use '==' it compares only the value:
If y = 1 then y == 4 returns false and y == 1 returns true
When you would use '===' it checks also if it is the same type.
So y === '1' returns fals and y === 1 returns true

- **Explain what a closure is. (Note that JavaScript programs use closures very often)**

Closures are inner functions inside of an outer function.
They have their own local scope and has access to outer function's scope, parameters (but NOT arguments object), and they also have access to global variables.
https://medium.com/@rlynjb/js-interview-question-what-is-a-closure-and-how-why-would-you-use-one-b6fd45ea95f6

In my own words:
Every time you create a function, there arises a closure. A closure means that you can use the function also outside the loop of the function.
To be able to use a closure you must expose a function within another function and return it or pass it on.

- **Explain what higher order functions are.**

First I want to quote a part of the book that will explain what Abstractions are, after that it is easier to understand what higher order functions are. Marijn Haverbeke said the following about abstractions:

'In the context of programming, these kinds of vocabularies are usually called abstractions. *Abstractions hide details and give us the ability to talk about problems at a higher (or more abstract) level.'.*

Then the following is being said about higher order funcions.

Functions that operate on other functions, either by taking them as arguments or by returning them, are called higher-order functions. *Since we have already seen that* *functions are regular values, there is nothing particularly remarkable about the fact that such functions exist. The term comes from mathematics, where the distinction between functions and other values is taken more seriously.*

Higher-order functions allow us to abstract over actions, *not just values. They come in several forms. For example, we can have functions that create new functions.* (Haverbeke)

So in summary a higher order function can take another function as an argument, or return a function as a result.

- **Explain what a query selector is and give an example line of JavaScript that uses a query selector.**

Gives the first element in the document that matches the specified selector, or group of selectors, or null if no matches are found.

```
var el = document.querySelector(".mijnklasse");
```

Bronnenlijst

Haverbeke, M. (2011). *Eloquent Javascript*. (3rd edition). No Starch Press.

https://medium.com/@rlynjb/js-interview-question-what-is-a-closure-and-how-why-would-you-use-one-b6fd45ea95f6

https://developer.mozilla.org/nl/docs/Web/API/Document/querySelector