

## Functions on java script

These are like methods in java.

You can have functions expressions where you declare the function as a variable. However these can later be overwritten.

- To replace some letters in a string
  - `string.replace(/what to replace/g "to replace with");`

## JS scopes.

Scopes are the variables which are stored in functions.

You can only access that scope in the method itself unless it is a global scope.

## High order functions

SetInterval takes your function and when to repeat it.

You don't need `any()` after the input function

## Arrays

- `Variable.push()`; to add at the end.
- `Variable .pop()`; removes the last element.
- `.unshift()`; add at beginning
- `.shift()`; remove from beginning.
- `.indexOf()`; returns index of searched parameter.
- `.slice(2,6)`; copies elements 2 to 6 from an array.

To iterate through arrays, `forEach` loop can be used.

- `.forEach(someFunction)`
- 
- `var fruits =["orange", "apple", "pear"];`
- `function print (fruits){`
- `console.log(fruits);`
- `}`
- `fruits.forEach(print);`

## Objects

- when creating objects, there is two ways of calling a certain field.
- `.name` dot notation or
- `["name"]` bracket notation

```
Var person ={  
  Name : "Ayana",  
  Age : 8,
```

```
Gender : "girl"  
};
```

- If the field name starts with a number, or the field has a space in between then you need to use bracket notation.

Methods can be added in the objects.

```
Add : function hello(){  
}.
```

Benefit of adding methods in the object. Due to method overriding.