Webpack

Task list:

1. Create project
2. Npm init and install webpack
3. Create index.html
4. Create module-a.js
5. Create module-b.js
6. Create main.js
7. Create color.scss
8. Create text.scss
9. Create main.scss
10. Find images
11. Convert JS,Sass and images in bundle.js

JavaScript

I create file module-a.js

Make use of the arrow function

```
// Arrow function
let number_x = 10;
const number_y = 15;
let func = (val) => {
    console.log("Hey im arrow " + val + ". And this is my value from num x and y: " + number_x + ", " + number_y);
}
```

Make use of the new ECMAScript 6 class structure

```
// Class

class Person {

   constructor(){

       this.name = "wojtek";

   }

}
```

Make use of the Template Strings

```
//template string
   sayName(){
      console.log('Hi, I am a ' + this.name + '.');
   }
```

Make use of Let and Const

```
let number_x = 10;
const number_y = 15;
```

A file called module-b.js that contains a javascript module that makes use of the JQuery library (JQuery must be imported via webpack not manually)

```
import $ from 'jquery';
window.jQuery = $;
window.$ = $;



   $("#content").html("Sass is working");
```

A main.js file that will be your main Javascript file responsible for calling the modules that your application needs (at least the module-a and module-b)

```
const {$} = require("./module-b.js");
const {Person,fun,img8kb} = require("./module-a.js");
import './style/main.scss';
import crown from '../assets/img8.png';
import skull from '../assets/svgimg8.svg';
import jpg from '../assets/jgpimage.jpg';

const img8b = document.getElementById('image');
const svgimg = document.getElementById('imageSvg');
const jpgimage = document.getElementById('imgJpg');
img8b.src = crown;
svgimg.src = skull;
jpgimage.src = jpg;

fun("Function");
var test = new Person();
test.sayName();
```

CSS

A sass file called colors.scss that applies the css style rules related to colors.

```
@import './text';

*{
   margin: 0;
   padding: 0;
}
$image: url('/assets/svgimg8.svg');
$colorRed: rgb(100, 25, 25);
$colorBlue: blue;

@mixin backG($color,$width,$height) {

   background-color: $color;
   width: $width;
   height: $height;
   display: flex;
   justify-content: center;
   text-align: center;
   margin: 0 auto;
}

#content{

   @include backG(rgb(63, 138, 148),80%,100px);
   @include fontS(55px,$colorBlue);


}
#imageSvg{

   width: 250px;
   height: 250px;
}
#img-test{
   @include backG(rgb(63, 138, 148),80%,100px);
   background-image: $image;

}
#imgJpg{
   width: 250px;
   height: 250px;
}
.test{

   @include backG(rgb(42, 133, 30),80%,100px);

   p {
       @include fontS(55px,$colorRed);


   }

}
```

A sass file called text.scss that applies the css style rules related to text.

```

@mixin fontS($pixels,$color) {
   font-size: $pixels;
   color: $color;
}
```

A sass file called main.scss (which will be responsible for importing the colors.scss and text.scss files)

```
@import './colors';

@import './text';

```

Images

```
import crown from '../assets/img8.png';
import skull from '../assets/svgimg8.svg';
import jpg from '../assets/jgpimage.jpg';

const img8b = document.getElementById('image');
const svgimg = document.getElementById('imageSvg');
const jpgimage = document.getElementById('imgJpg');
```

```
img8b.src = crown;
svgimg.src = skull;
jpgimage.src = jpg;
```

Webpack config

```
const path = require('path');
const webpack = require('webpack');
```

Path is for routing
Webpack is for import jquery

```
 entry: './src/main.js',
```

Entry is a main file js when webpack start

```
output: {
       path: path.resolve(__dirname),
       filename: 'bundle.js'
   },
```

Output is a file after converting bundle.js.

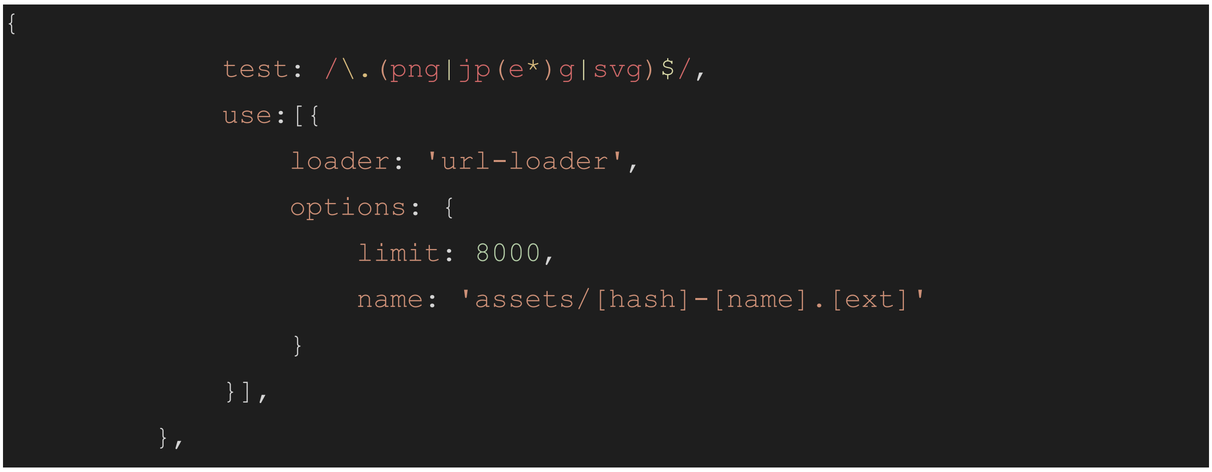
Plugins in this case is jQuery.

```
plugins: [
       new webpack.ProvidePlugin({
            $: "jquery",
            jQuery: "jquery"
        })
   ],
```

Modules for sass:

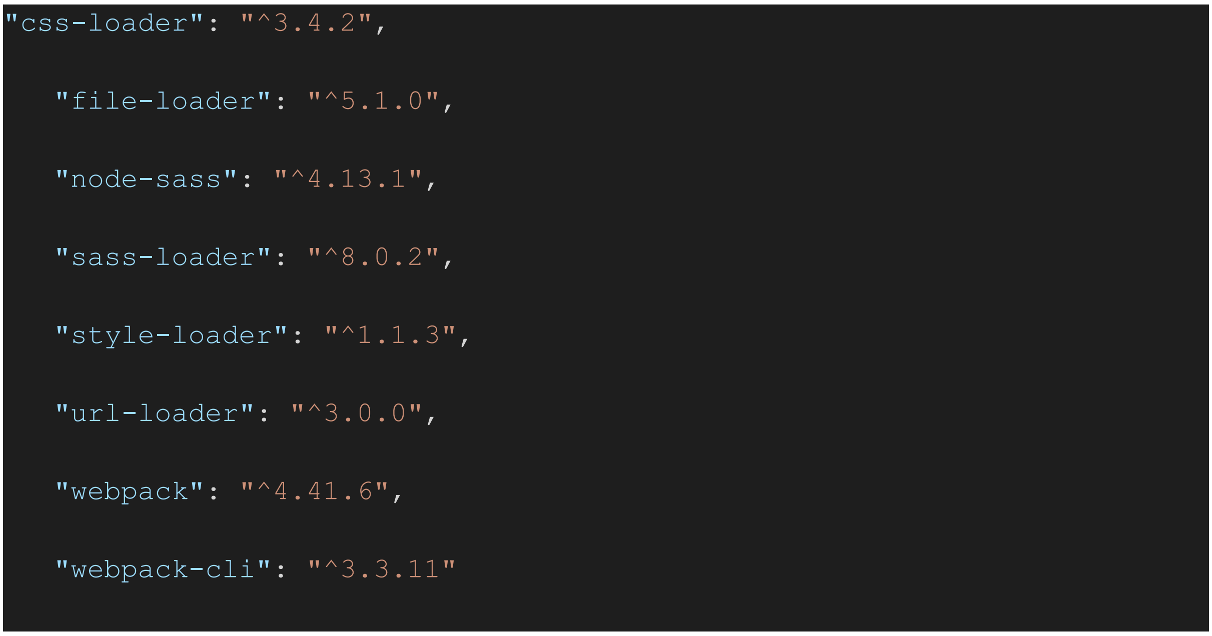
```
module:{
       rules:[
           {
           test: /\.s[ac]ss$/i,
           use: [
               'style-loader',
               'css-loader',
               'sass-loader',
           ],
         },
```

Modules for images compression



```
{
            test: /\.(png|jp(e*)g|svg)$/,
            use:[{
                loader: 'url-loader',
                options: {
                    limit: 8000,
                    name: 'assets/[hash]-[name].[ext]'
                }
            }],
        },
```

Documentation about the tools used in the project



```
"css-loader": "^3.4.2",

   "file-loader": "^5.1.0",

   "node-sass": "^4.13.1",

   "sass-loader": "^8.0.2",

   "style-loader": "^1.1.3",

   "url-loader": "^3.0.0",

   "webpack": "^4.41.6",

   "webpack-cli": "^3.3.11"
```

Record of lessons learned.

I learn about webpack how to manage dependencies i learn what is a different between devdependeccies and dependencies how to work with modules, how to use sass.