Javascript

Program the following functionality in the file **myscript.js**.
In the attached zip you'll find 4 files:

- **index.html**: Open this file in the browser. It contains links to the further Javascript files.
- **person_data.js**: An array of Person objects is defined here
- **students.js**: Array with Student objects
- **myscript.js**: Your code comes here.

Timestamp

Program a function **getTimestamp(includeDate, includeMilliseconds)** that builds a string from the current time. Both parameters should default to true.
The format should look like this: **2019-04-25 17:34:16.353** (i.e. with leading zeros).
Depending on the specification of the parameters, the date or the milliseconds should not be taken into account.
Extend the function by two optional additional parameters that specify whether the month and day are to be output as text. So, for example, **getTimestamp(true, false, true, true)** should give the following output: **Friday 25. April 2019, 17:34:16**

Personenlist

In the file **students.js** your names are given as array of objects.

```
const students = [
    { firstname: 'Isabelle', lastname: 'Arthofer', clazz: '4a' },
    { firstname: 'Fabian', lastname: 'Buchenberger', clazz: '4a' },
    { firstname: 'Oliver', lastname: 'Doppelbauer', clazz: '4a' },
    { firstname: 'Andreas', lastname: 'Eckerstorfer', clazz: '4a' }
```

Log this array as a string to the console and then build a Javascript object with the class names as keys and the list of students as value.
This object then has also to be output to the console.

Function as parameter

Write the following functions that create a different object from a Student object:

- `{ firstname: 'aaa', lastname: 'bbb', clazz: '4a' }` → `{ name: 'aaa bbb', clazz: '4a' }`
- `{ firstname: 'aaa', lastname: 'bbb', clazz: '4a' }` → `'aaa bbb [4a]'`

Then use these functions to transform the whole students array with **map**.

Callback

Write a function **function loopStudents(callback)** that iterates through the students array and invokes the callback each time the current object has a different class name than the previous one.

Array functions

In the file **person_data.json** a Javascript-array with 100 Person objects is available.
Log the following query results to the console, using **filter, sort, map,** and **forEach**.

Males

All males younger than 50 years, sorted alphabetically:

```
Andrews Raymond [22]
Austin Anthony [29]
Bishop Antonio [27]
Burns Ernest [39]
…
```

Registered

All registered persons, sorted by year of birth. The result array should contain objects as shown below: