JS Advanced: Exam 23 July 2017

Problems for exam preparation for the "JavaScript Advanced" course @ SoftUni. Submit your solutions in the SoftUni judge system at https://judge.softuni.bg/Contests/699/.

Problem 3. Task (Simple Class)

Write a **JavaScript class Task** that has a **title**, a **deadline** and a **status**.

```
class Task {
    // TODO: implement this class
}
```

The class **constructor** should receive two parameters – **title** (string), and **deadline** (JS Date object) and initialize the instance with **status** "Open". Implement the following features:

- Property **title**
- Property **deadline**
- Property **status** – will be either "Open", "In Progress" or "Complete" (there is no need to check)
- Property **isOverdue** – holds **true** if the **deadline** has been exceeded (the value is in the **past**) **and** the task is not completed
- **Static** function **comparator(a, b)** for **sorting** – see bellow for sorting instructions
- Function **toString()** – returns a string representation of a task in the following format:

`[{status icon}] {title} {(deadline: {deadline})|(overdue)}`

Print all **deadlines** in their **default format** (no additional formatting is required). If a task is **overdue**, print "**(overdue)**" instead of a deadline, and if a task is **complete**, do not print anything after the title. A task's **status icon** depends on its **status and deadline**:

Status	Icon	UTF Code
"Open"	✱	"\u2731"
"In Progress"	↝	"\u219D"
"Complete"	✓	"\u2714"
Overdue tasks (except "Complete")	⚠	"\u26A0"

See the examples for more formatting details.

Additionally, each instance must meet some **requirements**. A task should **not be created**, if its **deadline** is set to a **past date** – it should **throw an Error**. After creation, if the deadline is **changed** to a past date, an Error should be thrown. To determine whether a date is in the past, compare it with **Date.now()**.

When printing, sorting, or returning **isOverdue**, consider the task's **status** – "Completed" tasks are **not** overdue.

The **comparator** is function that **takes two** task instances as **parameters** and **returns a number**, depending on how they compare. If **a** should sort higher than **b**, return a **negative number**, if they are **equal**, return **zero**, and if **b** should sort higher than **a**, return a **positive number**. Overdue tasks are sorted highest (unless their **status** is "Complete"), "In Progress" tasks come second, "Open" tasks come third and "Complete" tasks sort lowest. If two tasks have the same status or are both overdue, sort them by **deadline** in ascending order (see the examples for details).

Examples

This is an example how the **Task** class is **intended to be used**:

Sample code usage

```javascript
let date1 = new Date();
date1.setDate(date1.getDate() + 7); // Set date 7 days from now
let task1 = new Task('JS Homework', date1);
let date2 = new Date();
date2.setFullYear(date2.getFullYear() + 1); // Set date 1 year from now
let task2 = new Task('Start career', date2);
console.log(task1 + '\n' + task2);
let date3 = new Date();
date3.setDate(date3.getDate() + 3); // Set date 3 days from now
let task3 = new Task('football', date3);
// Create two tasks with deadline set to current time
let task4 = new Task('Task 4', new Date());
let task5 = new Task('Task 5', new Date());
task1.status = 'In Progress';
task3.status = 'In Progress';
task5.status = "Complete";
let tasks = [task1, task2, task3, task4, task5];
setTimeout(() => {
  tasks.sort(Task.comparator);
  console.log(tasks.join('\n'));
}, 1000); // Sort and print one second later

// should throw an Error
let overdueTask = new Task('Overdue Task', new Date(2005, '4', '20'));
// should throw an Error
task1.deadline = new Date(2005, '4', '20');
```

Corresponding output

```
[✱] JS Homework (deadline: Wed Jul 26 2017 11:52:23 GMT+0300 (FLE Daylight Time))
[✱] Start career (deadline: Thu Jul 19 2018 11:52:23 GMT+0300 (FLE Daylight Time))

[⚠] Task 4 (overdue)
[↝] football (deadline: Sat Jul 22 2017 11:52:23 GMT+0300 (FLE Daylight Time))
[↝] JS Homework (deadline: Wed Jul 26 2017 11:52:23 GMT+0300 (FLE Daylight Time))
[✱] Start career (deadline: Thu Jul 19 2018 11:52:23 GMT+0300 (FLE Daylight Time))
[✓] Task 5
```

Task 5 is overdue, but it's status is "Complete". Tasks "football" and "JS Homework" has the same status and are sorted by deadline. Note that date **formatting is not considered** in the Judge and may differ in your solution!

Constraints

- Your class will be tested with both **valid and invalid deadlines** and should throw and Error for invalid values
- Your class will be tested with **only valid statuses** – "Open", "In Progress" and "Complete"

Submission

Submit **only** your class **Task**.

Follow us: