# Problem 3. Movie

class Movie {  
 *// TODO: implement this class...*  
}

Your Task

Write a class Movie, which implements the following functionality:

Functionality

#### constructor ( movieName, ticketPrice )

Receives 2 parameters at initialization of the class - movieName and ticketPrice.

Should have these **3** properties:

* **movieName** – property of type **string**;
* **ticketPrice** – property of type **number,** may come as a **string**;
* **screenings**– initially an empty **array**;

**Hint:** Here you can add more properties to help you finish the task.

#### newScreening(date, hall, description)

The date, hall and description are of type string.

* Check if there is already entered screening with the same date and hall and throw an Error:

"Sorry, {hall} hall is not available on {date}"

* Otherwise this function should add the screening to the screenings array and return:

"New screening of {movieName} is added."

#### endScreening(date, hall, soldTickets)

* Check if the given **screening array** has a screening with the given date and hall, if **NOT** **throw an Error**:

"Sorry, there is no such screening for {movieName} movie."

* Otherwise calculate the current screening profit of sold tickets, add the profit to the total profit for the movie, also add the sold tickets count to the total sold movie tickets, delete the screening from the screenings array and return:

"{movieName} movie screening on {date} in {hall} hall has ended. Screening profit: {currentProfit}"

#### toString ()

In the end is the toString() method where we return the full information of the movie.

* At the first line:

"{movieName} full information:"

* On the second two lines comes the collected profit, fixed to 0 and count of all sold tickets:

"Total profit: {profit}$

Sold Tickets: {soldTickets}"

* If there are screenings into the **screening array**, add on new line:

**"Remaining film screenings:"**

Sort screenings **alphabetically** by hall name and add a message for each of them on new line:

**"{hall} - {date} - {desrtiption}"**

* If there are no screenings into screenings array add this line to the returned message**:**

**"No more screenings!"**

Examples

|  |
| --- |
| Sample code usage |
| let m = new Movie('Wonder Woman 1984', '10.00');  console.log(m.newScreening('October 2, 2020', 'IMAX 3D', `3D`));  console.log(m.newScreening('October 3, 2020', 'Main', `regular`));  console.log(m.newScreening('October 4, 2020', 'IMAX 3D', `3D`));  console.log(m.endScreening('October 2, 2020', 'IMAX 3D', 150));  console.log(m.endScreening('October 3, 2020', 'Main', 78));  console.log(m.toString());  m.newScreening('October 4, 2020', '235', `regular`);  m.newScreening('October 5, 2020', 'Main', `regular`);  m.newScreening('October 3, 2020', '235', `regular`);  m.newScreening('October 4, 2020', 'Main', `regular`);  console.log(m.toString()); |
| Corresponding output |
| **New screening of Wonder Woman 1984 is added.**  **New screening of Wonder Woman 1984 is added.**  **New screening of Wonder Woman 1984 is added.**  **Wonder Woman 1984 movie screening on October 2, 2020 in IMAX 3D hall has ended. Screening profit: 1500**  **Wonder Woman 1984 movie screening on October 3, 2020 in Main hall has ended. Screening profit: 780**  **Wonder Woman 1984 full information:**  **Total profit: 2280$**  **Sold Tickets: 228**  **Remaining film screenings:**  **IMAX 3D - October 4, 2020 - 3D**    **Wonder Woman 1984 full information:**  **Total profit: 2280$**  **Sold Tickets: 228**  **Remaining film screenings:**  **235 - October 4, 2020 - regular**  **235 - October 3, 2020 - regular**  **IMAX 3D - October 4, 2020 - 3D**  **Main - October 5, 2020 - regular**  Main - October 4, 2020 - regular |

*GOOD LUCK!*