# JS Advanced: 30.08.2018

# Problem 2. Holiday Package (Unit Testing)

You are given the following **JavaScript class**:

|  |
| --- |
| HolidayPackage.js |
| class HolidayPackage {  constructor(destination, season) {  this.vacationers = [];  this.destination = destination;  this.season = season;  this.insuranceIncluded = false; // Default value  }  showVacationers() {  if (this.vacationers.length > 0)  return "Vacationers:\n" + this.vacationers.join("\n");  else  return "No vacationers are added yet";  }  addVacationer(vacationerName) {  if (typeof vacationerName !== "string" || vacationerName === ' ') {  throw new Error("Vacationer name must be a non-empty string");  }  if (vacationerName.split(" ").length !== 2) {  throw new Error("Name must consist of first name and last name");  }  this.vacationers.push(vacationerName);  }  get insuranceIncluded() {  return this.\_insuranceIncluded;  }  set insuranceIncluded(insurance) {  if (typeof insurance !== 'boolean') {  throw new Error("Insurance status must be a boolean");  }  this.\_insuranceIncluded = insurance;  }  generateHolidayPackage() {  if (this.vacationers.length < 1) {  throw new Error("There must be at least 1 vacationer added");  }  let totalPrice = this.vacationers.length \* 400;  if (this.season === "Summer" || this.season === "Winter") {  totalPrice += 200;  }  totalPrice += this.insuranceIncluded === true ? 100 : 0;  return "Holiday Package Generated\n" +  "Destination: " + this.destination + "\n" +  this.showVacationers() + "\n" +  "Price: " + totalPrice;  }  } |

### Functionality

*For this one your contribution to the project is to make sure that your colleagues have correctly implemented the HolidayPackage class, which takes care of generating an offer for a complete holiday package for the clients of the travel agency.*​

The above code defines a **class** that contains information about a **holiday package**. An **instance** of the class should support the following operations:

* Must be **instantiated** with two parameters – a string destination and a string season.
* Accessor insuranceIncluded – used to get and set the value of the insurance status.
* Function showVacationers() – returns a string with the vacationers’ names. If there aren’t any, returns a corresponding message.
* Function addVacationer() – adds a vacationer to the array of vacationers. In case of an invalid name, the function throws an error.
* Function generateHolidayPackage() – calculates the total price for the holiday package and returns a string, containing an overview of the generated holiday package. However, in case that there are no vacationers, it throws an error. The total price includes 400BGN per vacationer, an extra 200BGN season fee for the summer and winter seasons and 100BGN for the insurance, if included in the package.

When creating an instance, changing any of the property values, or adding a new vacationer, the parameters are validated. They must follow these rules:

* insuranceIncluded – Boolean
* vacationerName – non-empty string, containing a valid name. A name is considered valid when it consists of both first name and last name.

***Scroll down for examples and details about submitting to Judge.***

### Examples

This is an example how this code is **intended to be used**:

|  |
| --- |
| Sample code usage |
| let holidayPackage = new HolidayPackage('Italy', 'Summer');  console.log(holidayPackage.showVacationers());  // should throw an error  try {  holidayPackage.generateHolidayPackage();  } catch (err) {  console.log('Error: ' + err.message);  }  // should throw an error  try {  holidayPackage.addVacationer('');  } catch (err) {  console.log('Error: ' + err.message);  }  // should throw an error  try {  holidayPackage.addVacationer('Ivan');  } catch (err) {  console.log('Error: ' + err.message);  }  holidayPackage.addVacationer('Ivan Ivanov');  holidayPackage.addVacationer('Petar Petrov');  holidayPackage.addVacationer('Georgi Georgiev');  console.log(holidayPackage.showVacationers());  // should throw an error  try {  holidayPackage.insuranceIncluded = 'true';  } catch (err) {  console.log('Error: ' + err.message);  }  holidayPackage.insuranceIncluded = true;  console.log(holidayPackage.generateHolidayPackage()); |
| Corresponding output |
| No vacationers are added yet  Error: There must be at least 1 vacationer added  Error: Vacationer name must be a non-empty string  Error: Name must consist of first name and last name  Vacationers:  Ivan Ivanov  Petar Petrov  Georgi Georgiev  Error: Insurance status must be a boolean  Holiday Package Generated  Destination: Italy  Vacationers:  Ivan Ivanov  Petar Petrov  Georgi Georgiev  Price: 1500 |

### Your Task

Using **Mocha** and **Chai** write **JS unit tests** to test the entire functionality of the HolidayPackage class. Make sure instances of it have all the required functionality and validation. You may use the following code as a template:

|  |
| --- |
| describe("*TODO* …", function() {  *it*("*TODO …*", function() {  *// TODO:* …  });  *// TODO:* …  }); |

### Submission

Submit your tests inside a describe() statement, as shown above.