**JS Advanced Retake Exam**

**Problem 3. Unit Testing**

**Your Task**

Using **Mocha** and **Chai** write **JS Unit Tests** to test a variable named **bookSelection**, which represents an object. You may use the following code as a template:

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

The object that should have the following functionality:

**isGenreSuitable (genre, age) -** A function that accepts two parameters: **string** and **number**.

* If the value of the string **genre** is equal to "**Thriller**" or "**Horror**" and the value of **age** is less or equal to **12**,

return: **`Books with ${genre} genre are not suitable for kids at ${age} age`**

* Otherwise, if the above conditions are not met, **return** the following message:

**`Those books are suitable`**

* There is **no** need for **validation** for the **input**, you will always be given string and number.
* **isItAffordable (price, budget) -** A function that accepts two parameters: **number** and **number**.
* You need to **calculate** if you can afford buying the book by **subtracting** the **price** of the book from your **budget**.
* If the **result** is lower than **0,** return**:**

**"You don't have enough money"**

* Otherwise, if the above conditions are not met, **return** the following message:

**`Book bought. You have ${result}$ left`**

* You need to validate the input, if the **price** and **budget** are not a **number**, **throw** an error: "**Invalid input**".
* **suitableTitles** **(books, wantedGenre) -** A function that accepts an array and string.
  + The **books** array will store the titles and the genre of its books ([{ **title**: "**The** **Da** **Vinci** **Code**", **genre**: "**Thriller**" }, ...])
  + You must **add** the **title** of the book that its genre is equal to the **wantedGenre**.
* Finally, **return** the changed array of book titles.
  + There is a need for validation for the input, an **array** and **string** may not always be valid. In case of submitted **invalid** parameters, **throw** an error "**Invalid input**":
    - If passed **books** parameteris not an array.
    - If the **wantedGenre** is not a string.

**JS Code**

To ease you in the process, you are provided with an implementation that meets all of the specification requirements for the **bookSelection** object:

|  |
| --- |
| bookSelection.js |
| const bookSelection = {    isGenreSuitable(genre, age) {      if (age <= 12 && (genre === "Thriller" || genre === "Horror")) {        return `Books with ${genre} genre are not suitable for kids at ${age} age`;      } else {        return `Those books are suitable`;      }    },    isItAffordable(price, budget) {      if (typeof price !== "number" || typeof budget !== "number") {        throw new Error("Invalid input");      }      let result = budget - price;      if (result < 0) {        return "You don't have enough money";      } else {        return `Book bought. You have ${result}$ left`;      }    },    suitableTitles(array, wantedGenre) {      let resultArr = [];      if (!Array.isArray(array) || typeof wantedGenre !== "string") {        throw new Error("Invalid input");      }      array.map((obj) => {        if (obj.genre === wantedGenre) {          resultArr.push(obj.title);        }      });      return resultArr;    },  }; |

**Submission**

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