**3. Art Gallery**

**Your Task**

Using **Mocha** and **Chai** write **JS Unit Tests** to test a variable named **artGallery**, 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 should have the following functionality:

* **addArtwork** **(title, dimensions, artist) -** A function that accepts **three** parameters, all of them must be **strings.**
* There is a **need for validation** for the input.
* **title** and **artist** should be **non-empty** strings.
* If any parameter is invalid, **throw an error**:

**"Invalid Information!"**

* **dimensions** should be a **string** in the format **"width x height"**, where both width and height are positive numbers (e.g., "30 x 40").
* If dimensions does not match the required format, **throw an error**:

**"Invalid Dimensions!"**

* If the **artist** is not one of["Van Gogh", "Picasso", "Monet"]**, throw an error**:

**"This artist is not allowed in the gallery!"**

* If the input is valid, **return** the message:

**"Artwork added successfully: '{title}' by {artist} with dimensions {dimensions}."**

* **calculateCosts (exhibitionCosts, insuranceCosts, sponsor) -** A function that accepts three parameters: **number, number,** and **boolean**.
* There is a **need for validation** for the input.
* **exhibitionCosts** and **insuranceCosts** should be positive **numbers**.
* **sponsor** should be a **boolean**.
* If any parameter is invalid, **throw an error**:

**"Invalid Information!".**

* Calculate the **total cost** by summing **exhibitionCosts** and **insuranceCosts.**
* If sponsor is **true**, apply a **10%** discount to the total cost. **Return**:

**"Exhibition and insurance costs are {totalPrice}$, reduced by 10% with the help of a donation from your sponsor."**

* If sponsor is **false**, **return**:

**"Exhibition and insurance costs are {totalPrice}$."**

* **organizeExhibits (artworksCount, displaySpacesCount) -** A function that accepts two parameters: **number, number**.
* There is a **need for validation** for the input.
* Both **artworksCount** and **displaySpacesCount** should be positive **numbers**.
* If any parameter is **not** a number or is **negative**, **throw an error**:

**"Invalid Information!"**

* Calculate the **artworksPerSpace** by divide **artworksCount** by **displaySpacesCount** and rounded down**.**
* If the number of artworks per display space is **less** than 5, **return**:

**"There are only {artworksPerSpace} artworks in each display space, you can add more artworks."**

* Otherwise, **return**:

**"You have {displaySpacesCount} display spaces with {artworksPerSpace} artworks in each space."**

**JS Code**

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

|  |
| --- |
| artGallery.js |
| const artGallery = {    addArtwork(title, dimensions, artist) {      if (typeof title !== "string" || typeof artist !== "string") {        throw new Error("Invalid Information!");      }      if (!/^\d+ x \d+$/.test(dimensions)) {        throw new Error("Invalid Dimensions!");      }      const validArtists = ["Van Gogh", "Picasso", "Monet"];      if (!validArtists.includes(artist)) {        throw new Error("This artist is not allowed in the gallery!");      }      return `Artwork added successfully: '${title}' by ${artist} with dimensions ${dimensions}.`;    },    calculateCosts(exhibitionCosts, insuranceCosts, sponsor) {      if (        typeof exhibitionCosts !== "number" ||        typeof insuranceCosts !== "number" ||        typeof sponsor !== "boolean" ||        exhibitionCosts < 0 ||        insuranceCosts < 0      ) {        throw new Error("Invalid Information!");      }      let totalPrice = exhibitionCosts + insuranceCosts;      if (sponsor) {        totalPrice \*= 0.9;        return `Exhibition and insurance costs are ${totalPrice}$, reduced by 10% with the help of a donation from your sponsor.`;      } else {        return `Exhibition and insurance costs are ${totalPrice}$.`;      }    },    organizeExhibits(artworksCount, displaySpacesCount) {      if (        typeof artworksCount !== "number" ||        typeof displaySpacesCount !== "number" ||        artworksCount <= 0 ||        displaySpacesCount <= 0      ) {        throw new Error("Invalid Information!");      }      let artworksPerSpace = Math.floor(artworksCount / displaySpacesCount);      if (artworksPerSpace < 5) {        return `There are only ${artworksPerSpace} artworks in each display space, you can add more artworks.`;      } else {        return `You have ${displaySpacesCount} display spaces with ${artworksPerSpace} artworks in each space.`;      }    },  }; |

**Submission**

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