## 8.\*Class Storage

Create a **class** **Storage**. It should have the following **properties**, while the **constructor** should only receive a **capacity**:

* **capacity** – a number that **decreases when adding a given quantity** of products to storage
* **storage** – **list of products** (object). **Each product** should have:
* **name** - a string
* **price** – a number (price is for a single piece of product)
* **quantity** – a number
* **totalCost** – the sum of the cost of the products

The class should also have the following **methods:**

* **addProduct** – a function that receives a product and adds it to the storage
* **getProcuts** – a function that returns all the products in storage in **JSON** format, each on a new line

Paste only the **class** **Storage in judge** (**Note: all names should be as described**)

### Example

Test your Storage class.

|  |  |
| --- | --- |
| **Input** | **Output** |
| let productOne = {name: 'Cucamber', price: 1.50, quantity: 15};  let productTwo = {name: 'Tomato', price: 0.90, quantity: 25};  let productThree = {name: 'Bread', price: 1.10, quantity: 8};  let storage = new Storage(50);  storage.addProduct(productOne);  storage.addProduct(productTwo);  storage.addProduct(productThree);  console.log(storage.getProducts());  console.log(storage.capacity);  console.log(storage.totalCost); | {"name":"Cucamber","price":1.5,"quantity":15}  {"name":"Tomato","price":0.9,"quantity":25}  {"name":"Bread","price":1.1,"quantity":8}  2  53.8 |
| let productOne = {name: 'Tomato', price: 0.90, quantity: 19};  let productTwo = {name: 'Potato', price: 1.10, quantity: 10};  let storage = new Storage(30);  storage.addProduct(productOne);  storage.addProduct(productTwo);  console.log(storage.totalCost); | 28.1 |