**02. Car Management System**

You can check your solutions in [Judge.](https://judge.softuni.org/Contests/Compete/Index/5189" \l "1)

**Your Task**

Your task is to write a single **function** solve() that performs various data manipulation operations on a given array of car objects in a car dealership. You will **add**, **remove**, **update**, and **filter** cars within this array. The function solve() should take an **array** of **cars** as its **parameter**. **Inside** this **function**, you will define several **methods** to handle specific operations, and **each method** should **return** its **respective message**.

You are provided with an **array** called **cars** that contains objects. Each object represents a car and has the following **properties**:

|  |
| --- |
| let cars = [  { id: 1, brand: "Toyota", model: "Corolla", year: 2020, price: 20000, inStock: true },  { id: 2, brand: "Honda", model: "Civic", year: 2019, price: 22000, inStock: true },  { id: 3, brand: "Ford", model: "Mustang", year: 2021, price: 35000, inStock: false }  ]; |

Write the following **methods** **inside** the solve() function:

**Filter cars by brand**

* Write a function getCarsByBrand(brand) that takes a brand parameter and returns an array of cars belonging to the given brand.
* The method should return **filtered cars**.

#### Add a new car

* Write a function addCar(id, brand, model, year, price, inStock) that takes six parameters (**id**, **brand**, **model**, **year**, **price, inStock**) and adds a new car to the **cars** array.
* The method should return **cars array with new car included**.

#### Find car by ID

* Write a function getCarById(id) that takes an **id parameter** and returns the **car object** with the **given id**.
* The method should return **car object** or `Car with ID ${id} not found`

#### Remove car by ID

* Write a function removeCarById(id) that takes an **id parameter** and **removes** the **car** with the **given id** from the **cars** array.
* The method should return **cars array without removed car**.
* If the car with this ID does not exist, return: `Car with ID ${id} not found`

#### Update car price

* Write a function updateCarPrice(id, newPrice) that takes an **id** parameter and a **newPrice** parameter and updates the price of the car with the given id.
* The method should return **cars array with updated car price**.
* If the car with this ID does not exist, return: `Car with ID ${id} not found`

#### Update car stock status

* Write a function updateCarStock(id, inStock) that takes an **id** parameter and a boolean **inStock** parameter to update the stock status of the car with the given id.
* The method should return **cars array with updated car stock status**.
* If the car with this ID does not exist, return: `Car with ID ${id} not found`

At the end of the solve() function, **do not forget to include the final** **return** **statement** that returns an object with references to all the methods you have defined. This will allow the methods to be called from outside the solve() function.

A computer code with text

Description automatically generated with medium confidence

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| const dealership = solve(cars);  dealership.**getCarsByBrand**("Toyota"); | [{id: 1, brand: 'Toyota', model: 'Corolla', year: 2020, price: 20000, inStock: true}] |
| dealership.**addCar**(4, "Tesla", "Model S", 2022, 80000, true); | [{id: 1, brand: 'Toyota', model: 'Corolla', year: 2020, price: 20000, inStock: true}, {id: 2, brand: 'Honda', model: 'Civic', year: 2019, price: 22000, inStock: true}, {id: 3, brand: 'Ford', model: 'Mustang', year: 2021, price: 35000, inStock: false}, {id: 4, brand: 'Tesla', model: 'Model S', year: 2022, price: 80000, inStock: true}] |
| dealership.**getCarById**(2); | {id: 2, brand: 'Honda', model: 'Civic', year: 2019, price: 22000, inStock: true} |
| dealership.**removeCarById**(3); | [{id: 1, brand: 'Toyota', model: 'Corolla', year: 2020, price: 20000, inStock: true}, {id: 2, brand: 'Honda', model: 'Civic', year: 2019, price: 22000, inStock: true}] |
| dealership.**updateCarPrice**(1, 85000); | [{id: 1, brand: 'Toyota', model: 'Corolla', year: 2020, price: 85000, inStock: true}, {id: 2, brand: 'Honda', model: 'Civic', year: 2019, price: 22000, inStock: true}, {id: 3, brand: 'Ford', model: 'Mustang', year: 2021, price: 35000, inStock: false}] |
| dealership.**updateCarStock**(10, false); | Car with ID 10 not found |

**Submission**

In Judge submit only **solve()** function.