**2. Parking Lot**

class ParkingLot {

    //TODO...

}

Write a class **ParkingLot**, which implements the following functionality:

**Functionality**

**Constructor**

The **ParkingLot** class should have the following properties:

* **totalSpaces –** number (Total number of parking spaces available in the parking lot).
* **hourlyRate –** number (Hourly rate for parking).
* **availableSpaces –** number (Tracks the number of available parking spaces dynamically).
* **parkedVehicles –** array (Stores details of vehicles currently parked in the lot).
* **revenue -** number (Keeps track of the total revenue generated from parking fees).

**At the initialization of the ParkingLot class,** the constructor accepts two parameters: **totalSpaces** and **hourlyRate**! The rest of the properties must be empty or initialized accordingly.

**Hint:** You can add more properties to help you finish the task.

**parkVehicle (licensePlate) -** This method accepts one argument (**licensePlate**), a string representing the vehicle’s registration number. The method performs the following:

* Check if there is space available in the parking lot.
* If the parking lot is **full**, **return**:

**"The parking lot is full. No available spaces."**

* If the vehicle with the same license plate is already parked, **return**:

**"Vehicle with license plate {licensePlate} is already parked."**

* Otherwise, park the vehicle:
  + Decrease the **availableSpaces** by 1 and store the vehicle's **licensePlate** in the **parkedVehicles** array and **return**:

**"Vehicle with license plate {licensePlate} parked successfully."**

**unparkVehicle (licensePlate, hoursParked) –** This method accepts two arguments **(string and number).** The method performs the following**:**

* Check if the vehicle is parked in the lot. If not, **return**:

**"No vehicle found with license plate {licensePlate}."**

* If the vehicle is found, calculate the parking fee by multiplying **hoursParked** with **hourlyRate**.
* Add the calculated fee to revenue and increment **availableSpaces** by 1.
* Delete the car from **parkedVehicles** and **return**:

**"Vehicle with license plate {licensePlate} has been unparked. Parking fee: ${parkingFee}. Duration: {hoursParked} hours."**

**showAvailableSpaces () –** This method simply **returns** a message showing the available parking spaces along with the total spaces in the parking lot. The format should be:

**"Available parking spaces: {availableSpaces} out of {totalSpaces}."**

**listParkedVehicles ()** – This method lists all the currently parked vehicles. If there are vehicles parked, it **returns** the list of license plates in the format:

**"Currently parked vehicles:**

**A vehicle with registration number {licensePlate1} is in the parking lot.**

**A vehicle with registration number {licensePlate2} is in the parking lot.**

**A vehicle with registration number {licensePlate3} is in the parking lot.**

**…"**

* If no vehicles are parked, **return**:

**"No vehicles currently parked."**

**getTotalRevenue ()** – This method **returns** the total revenue rounded to the second decimal, generated from parking fees in the format:

**"Total revenue earned from parking fees:** ${revenue.toFixed(2)}"

**Example**

|  |
| --- |
| **Input 1** |
| const parkingLot = new ParkingLot(3, 5);  console.log(parkingLot.parkVehicle("ABC123"));  console.log(parkingLot.parkVehicle("ABC123"));  console.log(parkingLot.parkVehicle("XYZ789"));  console.log(parkingLot.parkVehicle("XYZ789")); |

|  |
| --- |
| **Output 1** |
| Vehicle with license plate ABC123 parked successfully.  Vehicle with license plate ABC123 is already parked.  Vehicle with license plate XYZ789 parked successfully.  Vehicle with license plate XYZ789 is already parked. |

|  |
| --- |
| **Input 2** |
| const parkingLot = new ParkingLot(3, 5);  console.log(parkingLot.parkVehicle("ABC123"));  console.log(parkingLot.parkVehicle("XYZ789"));  console.log(parkingLot.showAvailableSpaces());  console.log(parkingLot.listParkedVehicles()); |

|  |
| --- |
| **Output 2** |
| Vehicle with license plate ABC123 parked successfully.  Vehicle with license plate XYZ789 parked successfully.  Available parking spaces: 1 out of 3.  Currently parked vehicles:  A vehicle with registration number ABC123 is in the parking lot.  A vehicle with registration number XYZ789 is in the parking lot. |

|  |
| --- |
| **Input 3** |
| const parkingLot = new ParkingLot(3, 5);  console.log(parkingLot.parkVehicle("ABC123"));  console.log(parkingLot.parkVehicle("XYZ789"));  console.log(parkingLot.unparkVehicle("ABC123", 4));  console.log(parkingLot.unparkVehicle("BBC123", 14)); |

|  |
| --- |
| **Output 3** |
| Vehicle with license plate ABC123 parked successfully.  Vehicle with license plate XYZ789 parked successfully.  Vehicle with license plate ABC123 has been unparked. Parking fee: $20. Duration: 4 hours.  No vehicle found with license plate BBC123. |

|  |
| --- |
| **Input 4** |
| const parkingLot = new ParkingLot(3, 5);  console.log(parkingLot.parkVehicle("ABC123"));  console.log(parkingLot.parkVehicle("XYZ789"));  console.log(parkingLot.unparkVehicle("ABC123", 4));  console.log(parkingLot.unparkVehicle("XYZ789", 24));  console.log(parkingLot.getTotalRevenue()); |

|  |
| --- |
| **Output 4** |
| Vehicle with license plate ABC123 parked successfully.  Vehicle with license plate XYZ789 parked successfully.  Vehicle with license plate ABC123 has been unparked. Parking fee: $20. Duration: 4 hours.  Vehicle with license plate XYZ789 has been unparked. Parking fee: $120. Duration: 24 hours.  Total revenue earned from parking fees: $140.00 |