Practical 4

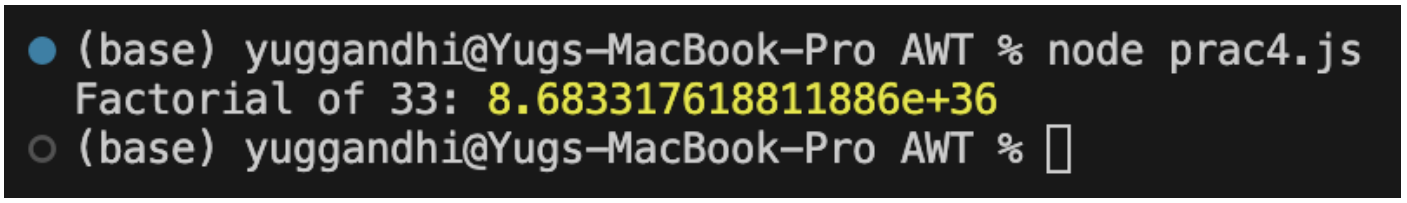
Problem Definition

Develop a calculator feature that finds the factorial of a number.Implement nested functions and demonstrate scope handling.

CODE:

```
function factorialCalculator() {

function computeFactorial(n) {

function recursiveCalc(x) {

if (x === 0 || x === 1) return 1;

return x * recursiveCalc(x - 1);

}

return recursiveCalc(n);

}

return function (input) {

if (typeof input !== 'number' || input < 0 || !Number.isInteger(input)) {

return 'Enter a valid non-negative integer';

}

return computeFactorial(input);

};

}

const calc = factorialCalculator();

console.log("Factorial of 33:", calc(33));
```

OUTPUT:



References:

1. JavaScript MDN Documentation
2. Lecture notes and class materials
3. W3Schools JavaScript Guide

Practical 5

Problem Definition

A transport company needs a system to manage different types of vehicles. Build a base class and extend it for cars and motorcycles.

CODE:

```
class Vehicle {

constructor(make, model, year, fuelConsumed, distanceTravelled) {

this.make = make;

this.model = model;

this.year = year;

this.fuelConsumed = fuelConsumed;

this.distanceTravelled = distanceTravelled;

}

calculateFuelEfficiency() {

return this.distanceTravelled / this.fuelConsumed;

}

getMaintenanceSchedule() {

return "General check every 10,000 km";
```

```
}
}
class Car extends Vehicle {
constructor(make, model, year, fuelConsumed, distanceTravelled, doors) {
super(make, model, year, fuelConsumed, distanceTravelled);
this.doors = doors;
}
getMaintenanceSchedule() {
return "Oil change every 5,000 km, tire rotation every 10,000 km";
}
}
class Motorcycle extends Vehicle {
constructor(make, model, year, fuelConsumed, distanceTravelled, hasCarrier) {
super(make, model, year, fuelConsumed, distanceTravelled);
this.hasCarrier = hasCarrier;
}
getMaintenanceSchedule() {
return "Chain lubrication every 3,000 km, oil change every 5,000 km";
}
}
const car1 = new Car("BMW", "M5", 2023, 60, 720, 4);
const car2 = new Car("Mercedes", "E-Class", 2022, 55, 660, 4);
const bike1 = new Motorcycle("Royal Enfield", "Hunter 350", 2023, 15, 420, true);
const bike2 = new Motorcycle("BMW", "G 310 RR", 2024, 18, 500, false);
console.log(car1.make, car1.model, car1.calculateFuelEfficiency(), car1.getMaintenanceSchedule());
console.log(car2.make, car2.model, car2.calculateFuelEfficiency(), car2.getMaintenanceSchedule());
```

```
console.log(bike1.make, bike1.model, bike1.calculateFuelEfficiency(), bike1.getMaintenanceSchedule());

console.log(bike2.make, bike2.model, bike2.calculateFuelEfficiency(), bike2.getMaintenanceSchedule());
```

OUTPUT:

```
(base) yuggandhi@Yugs-MacBook-Pro AWT % node prac5.js
BMW M5 12 Oil change every 5,000 km, tire rotation every 10,000 km
Mercedes E-Class 12 Oil change every 5,000 km, tire rotation every 10,000 km
Royal Enfield Hunter 350 28 Chain lubrication every 3,000 km, oil change every 5,000 km
BMW G 310 RR 27.77777777777778 Chain lubrication every 3,000 km, oil change every 5,000 km
(base) yuggandhi@Yugs-MacBook-Pro AWT %
```

References:

4. JavaScript MDN Documentation
5. Lecture notes and class materials
6. W3Schools JavaScript Guide