**Guvi tasks 5**

**Oops:**

1. “person” class to hold all the details

class Person1 {

constructor(name, section) {

this.name = "Poorni";

this.section = "12-B";

}

displayName() {

console.log("Method");

}

greet() {

console.log(this.name);

}

}

class Person2 {

constructor(name, section, subject) {

this.name = name || "Suresh";

this.section = section || "12-B";

this.subject = subject || "physics";

}

displayName() {

console.log("Method");

}

}

class Person3 {

constructor(name) {

this.name = name || "Saravanan";

}

displayName() {

console.log("Method");

}

}

class Geeks {

constructor(subject, duration, teacher) {

this.subject = subject || "chemistry";

this.duration = duration || "2 hrs";

this.teacher = teacher || "Manjula";

}

}

const personPrototype = {

greet: function () {

console.log(this.name);

}

};

1. class to calculate the uber price

class UberPrice {

constructor(carType, passengers, driverName) {

this.carType = 'Toyota'

this.passengers =1;

this.driverName = 'Siddharth';

}

calculatePrice(distanceInKm) {

const baseFare = 25;

const perKmRate = 8

if (distanceInKm <= 0) {

return 0; // No charge for zero or negative distance

}

else{

return extra charge;

}

const totalPrice = baseFare + perKmRate \* distanceInKm;

return totalPrice;

}

} const totalPrice = uberRide.calculatePrice(distance);

console.log(`Estimated price for your Uber ride in a ${uberRide.carType} with ${uberRide.passengers} passengers driven by ${uberRide.driverName} for ${distance} km: Rs. ${totalPrice}`);