KIRAN KUMAR D NATIKAR
EMP ID: 2576955
SOURCE CODE:

```
class Vehicle {
    int regNumber;
    String brand;
    double cost;
    int velocity;
    int capacity;
    int vehicleType;
    double vehicleTax;

    // Constructors
    public Vehicle() {
    }

    public Vehicle(int regNumber, String brand, double cost, int velocity, int
capacity, int vehicleType) {
        this.regNumber = regNumber;
        this.brand = brand;
        this.cost = cost;
        this.velocity = velocity;
        this.capacity = capacity;
        this.vehicleType = vehicleType;
    }

    // Getters and setters
    public int getRegNumber() {
        return regNumber;
    }

    public void setRegNumber(int regNumber) {
        this.regNumber = regNumber;
    }

    public String getBrand() {
        return brand;
    }

    public void setBrand(String brand) {
        this.brand = brand;
    }

    public double getCost() {
        return cost;
    }

    public void setCost(double cost) {
        this.cost = cost;
    }

    public int getVelocity() {
        return velocity;
    }

    public void setVelocity(int velocity) {
        this.velocity = velocity;
    }

    public int getCapacity() {
        return capacity;
    }

    public void setCapacity(int capacity) {
        this.capacity = capacity;
    }

    public int getVehicleType() {
        return vehicleType;
    }

    public void setVehicleType(int vehicleType) {
        this.vehicleType = vehicleType;
    }

    public double getVehicleTax() {
        return vehicleTax;
    }

    public void setVehicleTax(double vehicleTax) {
        this.vehicleTax = vehicleTax;
    }
}

class Property {
int id;
    double baseValueOfLand;
    char isInCity;
    int ageOfProp;
    double propertyTax;

    // Constructors
    public Property() {
    }

    public Property(int id,double baseValueOfLand, char isInCity, int ageOfProp) {
this.id= id;
        this.baseValueOfLand = baseValueOfLand;
        this.isInCity = isInCity;
        this.ageOfProp = ageOfProp;
    }

    // Getters and setters
    public int getid() {
        return id;
    }

    public void setid(int id) {
        this.id = id;
    }
    public double getBaseValueOfLand() {
```

KIRAN KUMAR D NATIKAR
EMP ID: 2576955
SOURCE CODE:

```
        return baseValueOfLand;
    }

    public void setBaseValueOfLand(double baseValueOfLand) {
        this.baseValueOfLand = baseValueOfLand;
    }

    public char getIsInCity() {
        return isInCity;
    }

    public void setIsInCity(char isInCity) {
        this.isInCity = isInCity;
    }

    public int getAgeOfProp() {
        return ageOfProp;
    }

    public void setAgeOfProp(int ageOfProp) {
        this.ageOfProp = ageOfProp;
    }

    public double getPropertyTax() {
        return propertyTax;
    }

    public void setPropertyTax(double propertyTax) {
        this.propertyTax = propertyTax;
    }
}


class VehicleOperations {
    ArrayList<Vehicle> vehicles = new ArrayList<>();

    public void addVehicleDetails(Vehicle vehicle) {
        vehicles.add(vehicle);
    }

    public void viewVehicleDetails() {
        if (vehicles.isEmpty()) {
            System.out.println("No Data Present at This Moment");
        } else {
System.out.println("================================================================
=============");
             System.out.println("Reg
Number\tBrand\tCost\tVelocity\tSeatCapacity\tTax");

      System.out.println("==========================================================
====");
            for (Vehicle vehicle : vehicles) {
                double vehicleTax = calculateVehicleTax(vehicle);
                vehicle.setVehicleTax(vehicleTax);
```

KIRAN KUMAR D NATIKAR
EMP ID: 2576955
SOURCE CODE:

```
                System.out.println(vehicle.brand + "\t" + vehicle.cost + "\t" + 
vehicle.velocity + "\t" + vehicle.capacity + "\t" + vehicleTax + "/-");
            }
        }
    }

    public double calculateVehicleTax(Vehicle vehicle) {
        double cost = vehicle.cost;
        int velocity = vehicle.velocity;
        int capacity = vehicle.capacity;
        int vehicleType = vehicle.vehicleType;

        double tax;
        switch (vehicleType) {
            case 1:
                tax = velocity + capacity + (0.10 * cost);
                break;
            case 2:
                tax = velocity + capacity + (0.11 * cost);
                break;
            case 3:
                tax = velocity + capacity + (0.12 * cost);
                break;
            default:
                throw new IllegalArgumentException("Invalid input for vehicle type. 
Use 1, 2, or 3.");
        }

        return tax;
    }
}
class PropertyOperations {
    private ArrayList<Property> properties = new ArrayList<>();

    public void addPropertyDetails(Property property) {
        getProperties().add(property);
    }

    public void viewPropertyDetails() {
        if (getProperties().isEmpty()) {
            System.out.println("No Data Present at This Moment");
        } else {
System.out.println("================================================================
======");
            System.out.println("ID\tBase Value\tIs In City\tAge(years)\tBuilt-up 
Area\tTax");

      System.out.println("==========================================================
");
            for (Property property : getProperties()) {
                double propertyTax = calculatePropertyTax(property);
                property.setPropertyTax(propertyTax);
                System.out.printf(property.id + "\t" + property.baseValueOfLand + 
"\t" + property.isInCity + "\t" + property.ageOfProp + "\t" + propertyTax + "/-");
            }
        }
    }

    public double calculatePropertyTax(Property property) {
        double baseValue = property.baseValueOfLand;
        char isInCity = Character.toUpperCase(property.isInCity);
        int age = property.ageOfProp;

        double tax;
        if (isInCity == 'Y') {
            tax = (baseValue * age * 0.5) + (0.5 * baseValue);
        } else if (isInCity == 'N') {
            tax = baseValue * age * 0.5;
        } else {
            throw new IllegalArgumentException("Invalid input for property location. 
Use 'Y' or 'N'.");
        }

        return tax;
    }

      public ArrayList<Property> getProperties() {
             return properties;
      }

      public void setProperties(ArrayList<Property> properties) {
             this.properties = properties;
      }
}
public class taxCalculation {
    public static void main(String[] args) {
      System.out.println("+------------------------------------+");
      System.out.println("|   WELCOME TO TAX CALCULATION APP   |");
      System.out.println("+------------------------------------+");
      Scanner scanner = new Scanner(System.in);
      String username;
      String password;
      String id = "admin";
      String pass = "admin123";
        System.out.print("Username: ");
        username = scanner.nextLine();
        System.out.print("Password: ");
        password = scanner.nextLine();
        if (username.equals(id) && password.equals(pass)) {
            System.out.println("Login successful.");
        PropertyOperations propertyOperations = new PropertyOperations();
        VehicleOperations vehicleOperations = new VehicleOperations();
        Property property = null;
        Vehicle vehicle = null;
        while (true) {
            System.out.println("1. Property Tax");
            System.out.println("2. Vehicle Tax");
            System.out.println("3. Total");
            System.out.println("4. Exit");
```

KIRAN KUMAR D NATIKAR
EMP ID: 2576955
SOURCE CODE:

```
            System.out.print("Select an Option : ");

            int choice = scanner.nextInt();
            scanner.nextLine();

            switch (choice) {
                case 1:
int id1;
                    double baseValueOfLand;
                    char isInCity;
                    int ageOfProp;
                 while(true) {
                    System.out.println("1. Add property details:");
                    System.out.println("2. Calculate property tax:");
                    System.out.println("3. Display all properties:");
                    System.out.println("4. Back to main menu");

                    int subChoice1 = scanner.nextInt();
                    scanner.nextLine();

                    switch(subChoice1) {
                    case 1:
                          System.out.print("Enter id of Land: ");
                           id1 = scanner.nextInt();
                           scanner.nextLine();

                           System.out.print("Enter Base Value of Land: ");
                           baseValueOfLand = scanner.nextDouble();
                           scanner.nextLine();

                           System.out.print("Is the Property in the City? (Y/N): ");
                           isInCity = scanner.nextLine().charAt(0);

                           System.out.print("Enter Age of Construction: ");
                           ageOfProp = scanner.nextInt();
                           scanner.nextLine();
                           property = new Property(id1, baseValueOfLand, isInCity,
ageOfProp);
                           propertyOperations.addPropertyDetails(property);
                           continue;
                    case 2:
                        double propertyTax =
propertyOperations.calculatePropertyTax(property);
                       property.setPropertyTax(propertyTax);
                        System.out.println("Property Tax Calculated: " + propertyTax
+ "/-");
                        continue;
                    case 3:
                          propertyOperations.viewPropertyDetails();
                          continue;
                    case 4:
                          break;
                    }
                  break;
            }
```

KIRAN KUMAR D NATIKAR
EMP ID: 2576955
SOURCE CODE:

```
                break;
                    case 2:
                    int regNumber;
                    String brand;
                    double cost;
                    int velocity;
                    int capacity;
                    int vehicleType;
                    while(true) {
                          System.out.println("1. Add vehicle details:");
                          System.out.println("2. Calculate vehicle tax:");
                          System.out.println("3. Display all vehicles:");
                          System.out.println("4. Back to main menu");

                    int subChoice2= scanner.nextInt();
                    scanner.nextLine();

                    switch(subChoice2) {
                    case 1:
                          System.out.print("Enter Registration Number: ");
                          regNumber = scanner.nextInt();
                          scanner.nextLine();

                          System.out.print("Enter Vehicle Brand: ");
                          brand = scanner.nextLine();

                          System.out.print("Enter Maximum Velocity (kmph): ");
                          velocity = scanner.nextInt();
                          scanner.nextLine();

                          System.out.print("Enter Capacity (Number of Seats): ");
                          capacity = scanner.nextInt();
                          scanner.nextLine();

                          System.out.println("Select Vehicle Type:");
                          System.out.println("1. Petrol-driven");
                          System.out.println("2. Diesel-driven");
                          System.out.println("3. CNG/LPG-driven");
                          System.out.print("Enter Vehicle Type (1/2/3): ");
                          vehicleType = scanner.nextInt();
                          scanner.nextLine();

                          System.out.print("Enter Cost of Vehicle: ");
                          cost = scanner.nextDouble();
                          scanner.nextLine();
                          vehicle = new Vehicle(regNumber, brand, cost, velocity,
capacity, vehicleType);
                          vehicleOperations.addVehicleDetails(vehicle);
                          continue;
                    case 2:
                          double vehicleTax =
vehicleOperations.calculateVehicleTax(vehicle);
                          vehicle.setVehicleTax(vehicleTax);
                          System.out.println("Vehicle Tax Calculated: " + vehicleTax
+ "/-");;
```

KIRAN KUMAR D NATIKAR
EMP ID: 2576955
SOURCE CODE:

```
                           continue;
                    case 3:
                          vehicleOperations.viewVehicleDetails();
                          continue;
                    case 4:
                          break;
            }
                    break;
                    }
                    break;
                case 3:
                                double totalPropertyTax =
calculateTotalTax(propertyOperations);
                                double totalVehicleTax =
calculateTotalTax(vehicleOperations);
                                double totalTaxPayable = totalPropertyTax +
totalVehicleTax;
                                System.out.println("Property Total Tax : " +
totalPropertyTax + "/-");
                                System.out.println("Vehicle Total Tax  : " +
totalVehicleTax + "/-");
                                System.out.println("Total              : " +
totalTaxPayable + "/-");
                    break;

                case 4:
                    scanner.close();
                    System.exit(0);

                default:
                    System.out.println("Invalid option, please choose again.");
            }
        }
    }
        else
        {
             System.out.println("Enter correct credentials");
        }
    }
    private static double calculateTotalTax(PropertyOperations propertyOperations) {
        double totalPropertyTax = 0;
        for (Property property : propertyOperations.getProperties()) {
            totalPropertyTax += property.getPropertyTax();
        }
        return totalPropertyTax;
    }

    private static double calculateTotalTax(VehicleOperations vehicleOperations) {
        double totalVehicleTax = 0;
        for (Vehicle vehicle : vehicleOperations.vehicles) {
            totalVehicleTax += vehicle.getVehicleTax();
        }
        return totalVehicleTax;
    }
```

KIRAN KUMAR D NATIKAR
EMP ID: 2576955
SOURCE CODE:

```
}
```