Vehicle

```
public class Vehicle {
    String name;
    int value;
    int seats;

    public Vehicle(String name, int value, int seats) {
        this.name = name;
        this.value = value;
        this.seats = seats;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public int getValue() {
        return value;
    }

    public void setValue(int value) {
        this.value = value;
    }

    public int getSeats() {
        return seats;
    }

    public void setSeats(int seats) {
        this.seats = seats;
    }

    public void print() {
        System.out.println(
                "Vehicle : name " + name + ", value $" + value + ", with "
+ seats + " seats");
    }
}
```

WheeledVehicle

```
public class WheeledVehicle extends Vehicle {


    public WheeledVehicle(String name, int value, int seats) {
        super(name, value, seats);
    }

    int wheels;

    public int getWheels() {
        return wheels;
    }

    public void setWheels(int wheels) {
        this.wheels = wheels;
    }

    public WheeledVehicle(String name, int value, int seats, int wheels) {
        super(name, value, seats);
        this.wheels = wheels;
    }

    @Override
    public void print(){
        System.out.println("WheeledVehicle : name " + name + ", value $" +
                value + ", with " + seats + " seats and " + wheels + "
wheels");
    }
}
```

MotorisedWheeledVehicle

```
public class MotorisedWheeledVehicle extends WheeledVehicle{
    public MotorisedWheeledVehicle(String name, int value, int seats, int
wheels) {
        super(name, value, seats, wheels);
    }

    int engineSize;

    public int getEngineSize() {
        return engineSize;
    }

    public void setEngineSize(int engineSize) {
        this.engineSize = engineSize;
    }

    public MotorisedWheeledVehicle(String name, int value, int seats, int
wheels, int engineSize) {
        super(name, value, seats, wheels);
        this.engineSize = engineSize;
    }
    @Override
    public void print(){
        System.out.println("MotorisedWheeledVehicle : name " + name + ",
value $" + value + ", with " + seats + " " +
                "seats and " + wheels + " wheels and " + "engine size " +
engineSize );
    }
}
```

FlyingVehicle

```
public class FlyingVehicle extends Vehicle{
    public FlyingVehicle(String name, int value, int seats) {
        super(name, value, seats);

    }
    int wings;

    public int getWings() {
        return wings;
    }

    public void setWings(int wings) {
        this.wings = wings;
    }

    public FlyingVehicle(String name, int value, int seats, int wings) {
        super(name, value, seats);
        this.wings = wings;
    }
    @Override
    public void print(){
        System.out.println("FlyingVehicle : name " + name + ", value $" +
                value + ", with " + seats + " seats and " + wings + "
wings");
    }
}
```

MotorisedFlyingVehicle

```
public class MotorisedFlyingVehicle extends FlyingVehicle{
    public MotorisedFlyingVehicle(String name, int value, int seats, int
wings) {
        super(name, value, seats, wings);
    }
    int engineSize;

    public MotorisedFlyingVehicle(String name, int value, int seats, int
wings, int engineSize) {
        super(name, value, seats, wings);
        this.engineSize = engineSize;
    }

    public int getEngineSize() {
        return engineSize;
    }

    public void setEngineSize(int engineSize) {
        this.engineSize = engineSize;
    }

    @Override
    public void print(){
        System.out.println("MotorisedWheeledVehicle : name " + name + ",
value $" + value + ", with " + seats + " " +
                "seats and " + wings + " wings and " + "engine size " +
engineSize );
    }
}
```

output:

```
public class Tester {
    public static void main(String[] args) {
        Vehicle[] v = new Vehicle[5];
        v[0] = new Vehicle( name: "Rowboat",  value: 500,  seats: 4);
        v[1] = new WheeledVehicle( name: "Skateboard",  value: 100,  seats: 0,  wheels: 4);
        v[2] = new MotorisedWheeledVehicle( name: "Car",  value: 10000,  seats: 5,  wheels: 4,  engineSize: 1200);
        v[3] = new FlyingVehicle( name: "Balloon",  value: 5000,  seats: 2,  wings: 0);
        v[4] = new MotorisedFlyingVehicle( name: "Airplane",  value: 500000,  seats: 20,  wings: 2,  engineSize: 6000);

        for (int i = 0; i < v.length; i++) {
            System.out.printf("Number %d is a ", i);
            v[i].print();

        }
    }
}
```

```
Tester
/Users/nancymacbook/Library/Java/JavaVirtualMachines/openjdk-20.0.2/Contents/Home/bin/java -javaagent:/Applications/IntelliJ
-Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath /Users/nancymacbook/Desktop/Prograr
Number 0 is a Vehicle : name Rowboat, value $500, with 4 seats
Number 1 is a WheeledVehicle : name Skateboard, value $100, with 0 seats and 4 wheels
Number 2 is a MotorisedWheeledVehicle : name Car, value $10000, with 5 seats and 4 wheels and engine size 1200
Number 3 is a FlyingVehicle : name Balloon, value $5000, with 2 seats and 0 wings
Number 4 is a MotorisedWheeledVehicle : name Airplane, value $500000, with 20 seats and 2 wings and engine size 6000
```

Requirement

Required output when the program runs

```
Number 0 is a Vehicle: name Rowboat, value $500, with 4 seats.
Number 1 is a WheeledVehicle: name Skateboard, value $100, with 0 seats and 4 wheels.
Number 2 is a MotorisedWheeledVehicle: name Car, value $10000, with 5 seats and 4 wheels and engine size 1200.
Number 3 is a FlyingVehicle: name Balloon, value $5000, with 2 seats and 0 wings.
Number 4 is a MotorisedFlyingVehicle: name Airplane, value $500000, with 20 seats and 2 wings and engine size 6000.
```