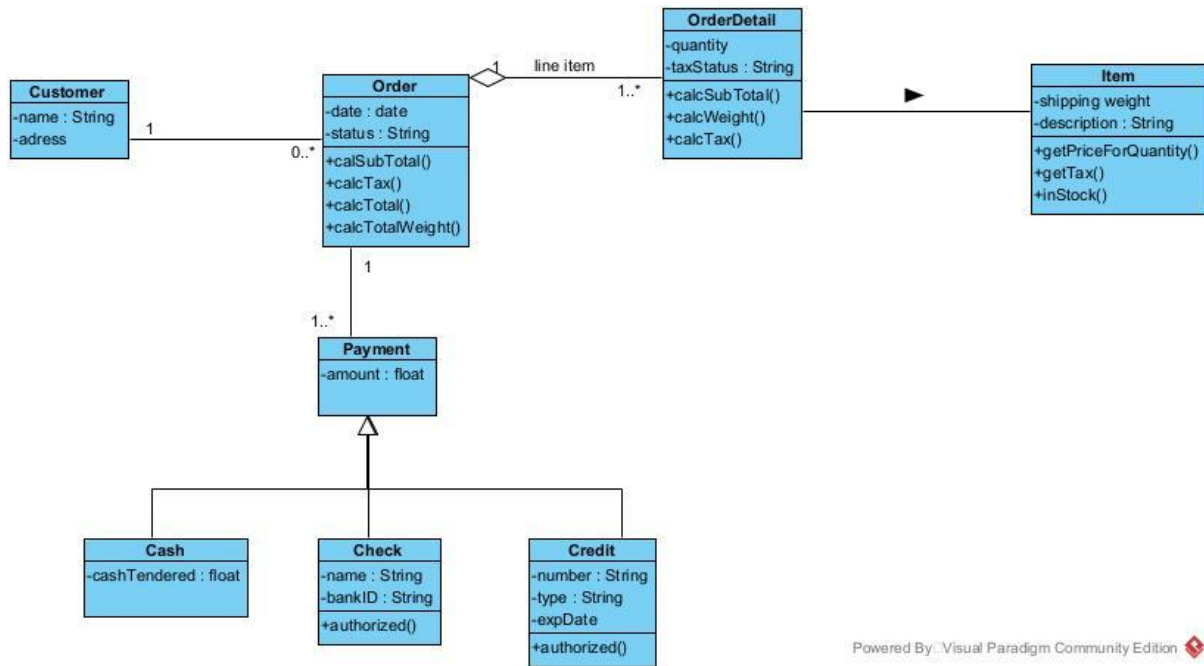


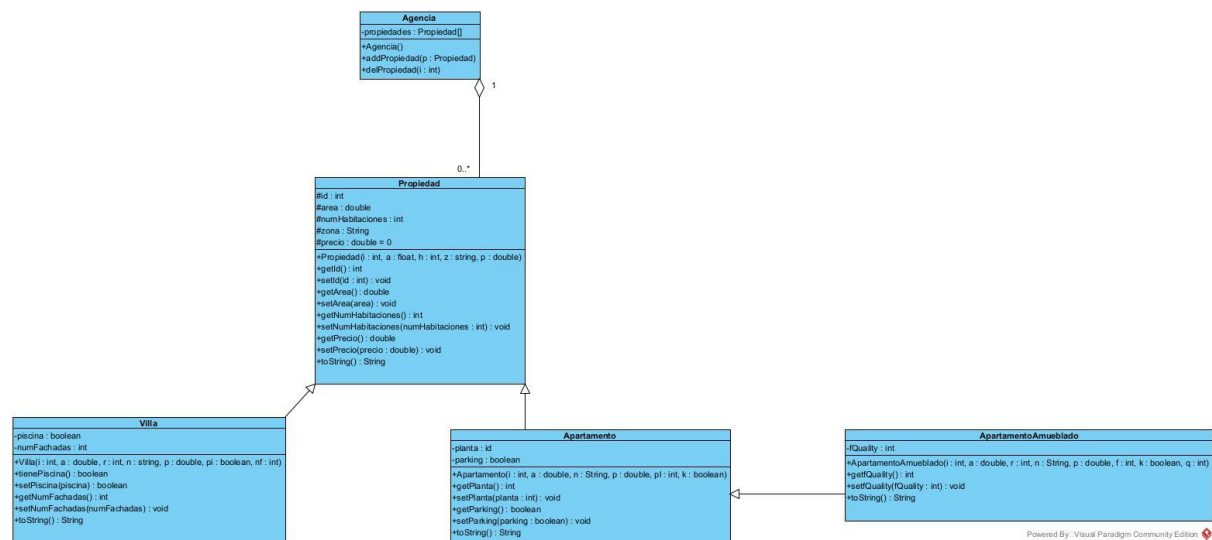
EJERCICIO 1.



REQUISITOS FUNCIONALES:

1. Calcular el precio del pedido así como su peso
2. Calcular los impuestos sobre el precio
3. Confirmar autorización

EJERCICIO 2.



Código de Java:

```

public class Propiedad {
    public int id;
    private double area;
    private int numHabitaciones;
    private String zona;
    private double precio=0;

    public Propiedad(int i, double a, int h, String z, double p) {
        id = i;
        area = a;
        numHabitaciones = h;
        zona = z;
        precio = p;
    }

    public int getId() { return id; }
    public double getArea() { return area; }
    public int getNumHabitaciones() { return numHabitaciones; }
    public String getZona() { return zona; }
    public int getPrecio() { return precio; }
    public String toString() {
        return id+":\tin "+getZona()+" tiene "+getNumHabitaciones()+" superficie "+getArea()+" m2";
    }
}

public class Villa extends Propiedad {
    private boolean piscina;
    private int numFachadas;
    public Villa(int i, double a, int r, String n, double p, boolean pi, int nf) {
        super(i, a, r, n, p);
        piscina = pi;
        numFachadas = nf;
    }
    public boolean tienePiscina() { return piscina; }
    public int getNumFachadas() { return numFachadas; }
    public String toString() {
        return "Villa #"+super.toString()+" piscina "+ tienePiscina() + " fachadas "+getNumFachadas();
    }
}

public class Apartamento extends Propiedad {
    private int planta;
    private boolean parking;
    public Apartamento(int i, double a, int r, String n, double p, int pl, boolean k) {
        super(i, a, r, n, p);
        planta = pl;
        parking = k;
    }
    public boolean tieneParking() { return parking; }
}

```

```

public int getPlanta() { return planta; }
public String toString() {
return "Apartamento #" + super.toString() + (tieneParking()? " con garaje":"" ) + " ubicado en
"+getPlanta();
}
}
public class ApartamentoAmueblado extends Apartamento {
private int fQuality;
public ApartamentoAmueblado(int i, double a, int r, String n, double p, int f, boolean k, int q) {
super(i, a, r, n, p, f, k);
fQuality = q;
}
public int getFQuality() { return fQuality; }
public String toString() {
return "Apartamento #" + super.toString() + (tieneParking()? " con garaje ":"") + " ubicado en
"+getPlanta()+ " amueblado
con " + (fQuality==5? "excellent":(fQuality==4? "v.good":(fQuality==3? "good": (fQuality==2?
"not bad":"poor quality")))) + "
mobiliario";
}
}
public class Agencia{
private Propiedad[] propiedades;
public Agencia() {
propiedades = new Propiedad[100];
for (int i=0; i<100; i++) propiedades[i] = null;
}
public void addPropiedad(Propiedad p) { //Código para añadir propiedad
}
public void delPropiedad(int i) { //código para eliminar la propiedad cuyo id coincida con "i"
}
}

```

EXPLICACIÓN:

He intentado hacer la ejecución pero no he podido porque no me acordaba de muchas de las cosas de Java. Por lo que he ido poco a poco copiándolo con la solución.