public class Computadores {  
  
    protected final static char CONSUMO\_W = 'F';  
    protected final static Double PRECIO\_BASE = 100.0;  
    protected final static Integer PESO\_BASE = 5;  
  
    protected Double precioBase;  
    protected char consumoW;  
    protected Integer peso;  
  
    // Constructores  
    public Computadores() {  
        this(PRECIO\_BASE, PESO\_BASE, CONSUMO\_W);  
    }  
  
    public Computadores(Double precioBase, Integer peso) {  
        this(precioBase, peso, CONSUMO\_W);  
    }  
  
    public Computadores(Double precioBase, Integer peso, char consumoW) {  
        this.precioBase = precioBase;  
        this.peso = peso;  
        this.consumoW = consumoW;  
    }  
  
    // Metodos  
    public Double calcularPrecio() {  
        Double adicion = 0.0;  
        switch (consumoW) {  
            case 'A':  
                adicion += 100.0;  
                break;  
            case 'B':  
                adicion += 80.0;  
                break;  
            case 'C':  
                adicion += 60.0;  
                break;  
            case 'D':  
                adicion += 50.0;  
                break;  
            case 'E':  
                adicion += 30.0;  
                break;  
            case 'F':  
                adicion += 10.0;  
                break;  
        }  
  
        if (peso >= 0 && peso < 19) {  
            adicion += 10.0;  
        } else if (peso >= 20 && peso < 49) {  
            adicion += 50.0;  
        } else if (peso >= 50 && peso <= 79) {  
            adicion += 80.0;  
        } else if (peso >= 80) {  
            adicion += 100.0;  
        }  
  
        return precioBase + adicion;  
    }  
  
    public Double getPrecioBase() {  
        return precioBase;  
    }  
  
    public char getconsumoW() {  
        return consumoW;  
    }  
  
    public Integer getPeso() {  
        return peso;  
    }  
}  
  
public class ComputadoresMesa extends Computadores {  
    private final static Integer ALMACENAMIENTO\_BASE = 50;  
  
    private Integer almacenamiento;  
  
    // Constructor  
    public ComputadoresMesa() {  
        this(PRECIO\_BASE, PESO\_BASE, CONSUMO\_W, ALMACENAMIENTO\_BASE);  
    }  
  
    public ComputadoresMesa(Double precioBase, Integer peso) {  
        this(precioBase, peso, CONSUMO\_W, ALMACENAMIENTO\_BASE);  
    }  
  
    public ComputadoresMesa(Double precioBase, Integer peso, char consumoW, Integer almacenamiento) {  
        super(precioBase, peso, consumoW);  
        this.almacenamiento = almacenamiento;  
    }  
  
    // Métodos  
    public Double calcularPrecio() {  
        Double adicion = super.calcularPrecio();  
        if (almacenamiento > 100) {  
            adicion += 50.0;  
        }  
  
        return adicion;  
    }  
  
    public Integer getCarga() {  
        return almacenamiento;  
    }  
}  
  
public class ComputadoresPortatiles extends Computadores {  
    private final static Integer PULGADAS\_BASE = 20;  
  
    private Integer pulgadas;  
    private boolean camaraITG;  
  
    // Constructor  
    public ComputadoresPortatiles() {  
        this(PRECIO\_BASE, PESO\_BASE, CONSUMO\_W, PULGADAS\_BASE, false);  
    }  
  
    public ComputadoresPortatiles(Double precioBase, Integer peso) {  
        this(precioBase, peso, CONSUMO\_W, PULGADAS\_BASE, false);  
    }  
  
    public ComputadoresPortatiles(Double precioBase, Integer peso, char consumoW, Integer pulgadas, boolean camaraITG) {  
        super(precioBase, peso, consumoW);  
        this.pulgadas = pulgadas;  
        this.camaraITG = camaraITG;  
    }  
  
    // Métodos  
    public Double calcularPrecio() {  
        Double adicion = super.calcularPrecio();  
        if (pulgadas > 40) {  
            adicion += precioBase \* 0.3;  
        }  
        if (camaraITG) {  
            adicion += 50.0;  
        }  
  
        return adicion;  
    }  
}  
  
public class PrecioTotal {  
    private Double totalComputadores = 0.0;  
    private Double totalComputadoresPortatiles = 0.0;  
    private Double totalComputadoresMesa = 0.0;  
    private Computadores[] listaComputadores;  
  
    // Constructor  
    PrecioTotal(Computadores[] pComputadores){  
        this.listaComputadores = pComputadores;  
    }  
  
    public void mostrarTotales() {  
        for (int i = 0; i < listaComputadores.length; i++) {  
  
            if (listaComputadores[i] instanceof Computadores) {  
                totalComputadores += listaComputadores[i].calcularPrecio();  
            }  
            if (listaComputadores[i] instanceof ComputadoresMesa) {  
                totalComputadoresMesa += listaComputadores[i].calcularPrecio();  
            }  
            if (listaComputadores[i] instanceof ComputadoresPortatiles) {  
                totalComputadoresPortatiles += listaComputadores[i].calcularPrecio();  
            }  
        }  
  
        // Mostramos los resultados  
        System.out.println("La suma del precio de los computadores es de " + totalComputadores);  
        System.out.println("La suma del precio de los computadores de mesa es de " + totalComputadoresMesa);  
        System.out.print("La suma del precio de los computadores portátiles es de " + totalComputadoresPortatiles);  
    }  
}