Explicación de la resolución junto con ideas que surgieron:

Una de las ideas fue que Prenda sea una clase y los tipos de prendas (saco, pantalón y camisa) sean objetos que heredan de la clase prenda, pero esta idea fue descartada ya que en este caso no me modifica que sea un saco de un pantalón de una camisa debido que (por ahora) no tienen comportamientos distintos.

Otra de las ideas pero que también fue descartada, fue que tipoDePago sea una interfaz ya que me cambiaba si era efectivo o tarjeta, debido a que en esta ultima se le adicionaban la cantidad de cuotas x un valor.

Para finalizar me quede con la idea de que Prenda sea una clase la cual va a tener tres atributos (precioBase, tipoProducto y estadoPrenda) y el método calcularPrecio, en donde esta clase va a utilizar lo que le brinde la clase abstracta EstadoPrenda que tiene un método calcularPrecio, el cual voy a utilizar en la clase Prenda; Además EstadoPrenda va a tener como subclases a NuevaPrenda,PromocionPrenda y LiquidacionPrenda en donde utilizo polimorfismo mediante su método calcularPrecio; Es una clase abstracta porque no existe un “estado de prenda” pero si existe una prenda nueva, una en liquidación y una en promoción.

Luego tengo la clase Venta la cual esta relacionada con la clase Prenda, ya que una Venta tiene prendas y esta tiene como atributos fecha, prendas y tipoPago y dos métodos (calcularTotal y calcularSubTotal); Este último método lo utilizo de la clase TipoDePago, en donde va a tener dos subclases Efectivo y Tarjeta en donde utilizo polimorfismo con el método calcularSubTotal.

La ultima clase que agregue fue la de RegistroVenta la cual va a tener una lista de ventas y un método gananciaDelDia que dependiendo un día (el cual es pasado por parámetro) filtra las ventas de ese día y hace una suma del total vendido.