

Igor Ramon da conceição

1.

Estratégia (interface): Empresa.java/Distancia.java/EmpresasEntrega.java

Estratégia (implementação):

ABC.java/XPTO.java/SENAC.java/ListaEmpresas.java

=====

2.

```
package prova;
```

```
public class Pulando implements EstadoPlayer{
```

```
    public Pular() {
        System.out.println("Pulando!");
    }
```

```
    @Override
    public EstadoPlayer esperar() {
        return new Esperando();
    }
```

```
    @Override
    public EstadoPlayer pular() {
        System.out.println("está pulando");
        return this;
    }
```

```
}
```

```
-----
package prova;
```

```
public class Agachando implements EstadoPlayer{
```

```
    public abaixar() {
        System.out.println("abaixando");
    }
```

```
    @Override
    public EstadoPlayer esperar() {
        return new Esperando();
    }
```

```
    @Override
    public EstadoPlayer abaixar() {
        System.out.println("Já está abaixando");
        return this;
    }
```

```
}
```

```
-----
package prova;
```

```

public interface EstadoPlayer {
    public EstadoPlayer esperar();
    public EstadoPlayer correr();
    public EstadoPlayer agachar();
    public EstadoPlayer pular();
}

```

```

-----
package Prova;

```

```

public class ControlaAcoes {
    private EstadoPlayer estado;

    public ControlaAcoes() {
        estado=new Esperando();
    }

    public void esperar(){
        estado=estado.esperar();
    }

    public void correr(){
        estado=estado.correr();
    }
    public void pular(){
        estado=estado.pular();
    }
    public void agachar(){
        estado=estado.agachar();
    }
}

```

```

-----

package prova;

```

```

public class GameState {

    public static void main(String[] args) {
        ControlaAcoes acao=new ControlaAcoes();
        acao.esperar();
        acao.correr();
        acao.esperar();
        acao.agachar();
        acao.pular();
    }
}

```

```

=====
3.

```

```

=====

4.

```

```

interface Produtos{
    public abstract double calcularProduto(Compra compra,ListaProdutos
produto);
}

    public static void main(String[] args) {
        Compra detergente = new Compra(50);
        System.out.println(obterProduto(compra,
ListaProdutos.values()[2]));

        Compra chá = new Compra(25);
        System.out.println(obterProduto(compra,
ListaProdutos.values()[1]));
    }

public static double obterProduto(Compra compra, ListaProdutos produto){
    return produto.calcularProduto(compra, produto);
}

public class Compra {
    private double compra;

    public Compra(double compra) {
        this.compra = compra;
    }

    public double getCompra() {
        return compra;
    }
}

public enum ListaProdutos implements Produtos{
    varejo{
        @Override
        public double calcularProduto(Compra compra,ListaProdutos
produto) {
            return compra.getCompra()*1;
        }

    },
    atacado{
        @Override
        public double calcularProduto(Compra compra,ListaProdutos
produto) {
            return compra.getCompra()*5;
        }

    };
}

```