Частное учреждение образования

«Колледж бизнеса и права»

Тема: «Паттерны в языке. Порождающие паттерны»

Учащийся М.Н. Гончаров.

группа: Т-992

2022

Задание (Вариант 6) & (Реализовать через абстрактную фабрику)

Имеется фабрика по производству сыра, молока и сметаны из

коровьего, козьего и птичьего молока. Пользователь выбирает нужную

ему категорию.

// inf, init, main

/\* ============================ interfeces of products ============================ \*/  
interface Cheeze {  
 long getCheezePrice();  
}  
interface Milk {  
 long getMilkPrice();  
}  
interface SoureScream {  
 long getSoureScreamPrice();  
}  
/\* ============================ Initelization Abstract Factory ============================ \*/  
interface InteAbsFactory {  
 Cheeze getCheeze();  
 Milk getMilk();  
 SoureScream getSoureScream();  
}  
  
/\*===================== Classes of type COW (Корова), GOAT (Коза), BIRD (Птица) ======================\*/  
// START =================================================================================================================== COW (Корова)  
class Cow\_CheezeImpl implements Cheeze {  
 public long getCheezePrice() {  
 return 1;  
 }  
}  
class Cow\_MilkImpl implements Milk {  
 public long getMilkPrice() {  
 return 2;  
 }  
}  
class Cow\_SoureScreamImpl implements SoureScream {  
 public long getSoureScreamPrice() {  
 return 3;  
 }  
}  
class Cow\_MilkPriceAbsFactory implements InteAbsFactory {  
 public Cheeze getCheeze() {  
 return new Cow\_CheezeImpl();  
 }  
 public Milk getMilk() {  
 return new Cow\_MilkImpl();  
 }  
 public SoureScream getSoureScream() {  
 return new Cow\_SoureScreamImpl();  
 }  
}// END =================================================================================================================== COW (Корова)  
  
  
  
// START =================================================================================================================== GOAT (Коза)  
class Goat\_CheezeImpl implements Cheeze {// вторая  
 public long getCheezePrice() {  
 return 10;  
 }  
}  
class Goat\_MilkImpl implements Milk {  
 public long getMilkPrice() {  
 return 20;  
 }  
}  
class Goat\_SoureScreamImpl implements SoureScream {  
 public long getSoureScreamPrice() {  
 return 30;  
 }  
}  
class Goat\_MilkPriceAbsFactory implements InteAbsFactory {  
 public Cheeze getCheeze() {  
 return new Goat\_CheezeImpl();  
 }  
 public Milk getMilk() {  
 return new Goat\_MilkImpl();  
 }  
 public SoureScream getSoureScream() {  
 return new Goat\_SoureScreamImpl();  
 }  
}// END =================================================================================================================== GOAT (Коза)  
  
  
  
// START =================================================================================================================== BIRD (Птица)  
class Bird\_CheezeImpl implements Cheeze {// вторая  
 public long getCheezePrice() {  
 return 100;  
 }  
}  
class Bird\_MilkImpl implements Milk {  
 public long getMilkPrice() {  
 return 200;  
 }  
}  
class Bird\_SoureScreamImpl implements SoureScream {  
 public long getSoureScreamPrice() {  
 return 300;  
 }  
}  
class Bird\_MilkPriceAbsFactory implements InteAbsFactory {  
 public Cheeze getCheeze() {  
 return new Goat\_CheezeImpl();  
 }  
 public Milk getMilk() {  
 return new Goat\_MilkImpl();  
 }  
 public SoureScream getSoureScream() {  
 return new Goat\_SoureScreamImpl();  
 }  
}// END =================================================================================================================== BIRD (Птица)  
  
/\* OUR class and main\*/  
public class Main {  
 public static void main(String[] args) {  
 ////////////////////////////////////////////// Type of product (Cow Goat Bird)  
 //String Type\_Product = "Cow";  
 String Type\_Product = "Goat";  
 //String Type\_Product = "Bird";  
  
 // Create Abs Factory (Create variors Product)  
 InteAbsFactory ifactory = null;  
  
 // Check our product  
 if(Type\_Product.equals("Cow"))  
 {  
 ifactory = new Cow\_MilkPriceAbsFactory();  
 }  
 else if (Type\_Product.equals("Goat"))  
 {  
 ifactory = new Goat\_MilkPriceAbsFactory();  
 }  
 else if (Type\_Product.equals("Bird"))  
 {  
 ifactory = new Bird\_MilkPriceAbsFactory();  
 }  
  
 // Ini.. products  
 Cheeze Cheeze = ifactory.getCheeze();  
 Milk Milk = ifactory.getMilk();  
 SoureScream SoureScream = ifactory.getSoureScream();  
  
 // print fot check  
 System.*out*.println(Cheeze.getCheezePrice());  
 System.*out*.println(Milk.getMilkPrice());  
 System.*out*.println(SoureScream.getSoureScreamPrice());  
 }  
}

// output

