```java
package edu.cpp.cs5800.macronutrient;

public abstract class Food {
    public String toString() {
        return String.format("%s(%s)", this.getType(), this.getName());
    }

    public abstract String getName();

    public abstract String getType();
}

```

```java
package edu.cpp.cs5800.macronutrient;

import edu.cpp.cs5800.macronutrient.carbs.Carbs;
import edu.cpp.cs5800.macronutrient.fats.Fats;
import edu.cpp.cs5800.macronutrient.protein.Protein;

public class Meal {
    private Carbs carbs;
    private Fats fats;
    private Protein protein;

    public Meal(Carbs carbs, Fats fats, Protein protein) {
        this.carbs = carbs;
        this.fats = fats;
        this.protein = protein;
    }

    @Override
    public String toString() {
        return String.format("Meal: %s, %s, %s", this.carbs, this.fats, this.protein);
    }
}

```

```java
package edu.cpp.cs5800.macronutrient;

import edu.cpp.cs5800.macronutrient.diet.Diet;

public class Customer {
    private String name;
    private Diet diet;

    public Customer(String name, Diet diet) {
        this.name = name;
        this.diet = diet;
    }

    public String getName() {
        return this.name;
    }

    public Diet getDiet() {
        return this.diet;
    }

    @Override
    public String toString() {
        return "Customer (name=" + this.name + ", diet=" + this.diet + ")";
    }
}

```

```java
package edu.cpp.cs5800.macronutrient;

import edu.cpp.cs5800.macronutrient.diet.Diet;

import java.util.List;
import java.util.Random;

public abstract class FoodFactory {
    private final Random rand = new Random();

    public String getFoodName(Diet diet) {
        List<String> nonEatableItems = diet.getNonEatableItems();
        List<String> eatableItems = this.getItems()
                .stream()
                .filter(item -> !nonEatableItems.contains(item))
                .toList();
        return eatableItems.get(rand.nextInt(eatableItems.size()));
    }

    public abstract List<String> getItems();
}

```

```java
package edu.cpp.cs5800.macronutrient;

import edu.cpp.cs5800.macronutrient.carbs.Carbs;
import edu.cpp.cs5800.macronutrient.carbs.CarbsFactory;
import edu.cpp.cs5800.macronutrient.diet.Diet;
import edu.cpp.cs5800.macronutrient.diet.NutAllergy;
import edu.cpp.cs5800.macronutrient.diet.Paleo;
import edu.cpp.cs5800.macronutrient.diet.Vegan;
import edu.cpp.cs5800.macronutrient.fats.Fats;
import edu.cpp.cs5800.macronutrient.fats.FatsFactory;
import edu.cpp.cs5800.macronutrient.protein.Protein;
import edu.cpp.cs5800.macronutrient.protein.ProteinFactory;

public class PaleoMealFactory extends MacronutrientFactory {
    private static PaleoMealFactory instance;
    private final Diet diet = new Paleo();

    private PaleoMealFactory() {
    }

    @Override
    public Carbs createCarbs() {
        return CarbsFactory.getInstance().getCarbs(this.diet);
    }

    @Override
    public Protein createProtein() {
        return ProteinFactory.getInstance().getProtein(this.diet);
    }

    @Override
    public Fats createFats() {
        return FatsFactory.getInstance().getFats(this.diet);
    }

    public static PaleoMealFactory getInstance() {
        if (instance == null) {
            instance = new PaleoMealFactory();
        }
        return instance;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient;

import edu.cpp.cs5800.macronutrient.carbs.Carbs;
import edu.cpp.cs5800.macronutrient.carbs.CarbsFactory;
import edu.cpp.cs5800.macronutrient.diet.Diet;
import edu.cpp.cs5800.macronutrient.diet.Vegan;
import edu.cpp.cs5800.macronutrient.fats.Fats;
import edu.cpp.cs5800.macronutrient.fats.FatsFactory;
import edu.cpp.cs5800.macronutrient.protein.Protein;
import edu.cpp.cs5800.macronutrient.protein.ProteinFactory;

public class VeganMealFactory extends MacronutrientFactory {
    private static VeganMealFactory instance;
    private final Diet diet = new Vegan();

    private VeganMealFactory() {
    }

    @Override
    public Carbs createCarbs() {
        return CarbsFactory.getInstance().getCarbs(this.diet);
    }

    @Override
    public Protein createProtein() {
        return ProteinFactory.getInstance().getProtein(this.diet);
    }

    @Override
    public Fats createFats() {
        return FatsFactory.getInstance().getFats(this.diet);
    }

    public static VeganMealFactory getInstance() {
        if (instance == null) {
            instance = new VeganMealFactory();
        }
        return instance;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient;

import edu.cpp.cs5800.macronutrient.carbs.Carbs;
import edu.cpp.cs5800.macronutrient.diet.NoRestriction;
import edu.cpp.cs5800.macronutrient.diet.NutAllergy;
import edu.cpp.cs5800.macronutrient.diet.Paleo;
import edu.cpp.cs5800.macronutrient.diet.Vegan;
import edu.cpp.cs5800.macronutrient.fats.Fats;
import edu.cpp.cs5800.macronutrient.protein.Protein;

public class MacroNutrientDriver {
    public static void main(String[] args) {
        Customer[] customers = new Customer[6];
        customers[0] = new Customer("John", new NoRestriction());
        customers[1] = new Customer("Robertson", new NutAllergy());
        customers[2] = new Customer("Alicia", new Paleo());
        customers[3] = new Customer("Paxton", new Vegan());
        customers[4] = new Customer("Jimmy", new NoRestriction());
        customers[5] = new Customer("Jayvon", new Paleo());

        for (Customer customer : customers) {
            MacronutrientFactory factory = MacroNutrientFactoryCreator.createFactory(
customer.getDiet());
            Carbs carbs = factory.createCarbs();
            Fats fats = factory.createFats();
            Protein protein = factory.createProtein();
            Meal meal = new Meal(carbs, fats, protein);
            String message = String.format("%s, %s", customer, meal);
            System.out.println(message);
        }
    }
}

```

```java
package edu.cpp.cs5800.macronutrient;

import edu.cpp.cs5800.macronutrient.carbs.Carbs;
import edu.cpp.cs5800.macronutrient.fats.Fats;
import edu.cpp.cs5800.macronutrient.protein.Protein;

public abstract class MacronutrientFactory {
    public abstract Carbs createCarbs();
    public abstract Protein createProtein();
    public abstract Fats createFats();
}

```

```java
package edu.cpp.cs5800.macronutrient;

import edu.cpp.cs5800.macronutrient.carbs.Carbs;
import edu.cpp.cs5800.macronutrient.carbs.CarbsFactory;
import edu.cpp.cs5800.macronutrient.diet.Diet;
import edu.cpp.cs5800.macronutrient.diet.NutAllergy;
import edu.cpp.cs5800.macronutrient.fats.Fats;
import edu.cpp.cs5800.macronutrient.fats.FatsFactory;
import edu.cpp.cs5800.macronutrient.protein.Protein;
import edu.cpp.cs5800.macronutrient.protein.ProteinFactory;

public class NutAllergyMealFactory extends MacronutrientFactory {
    private static NutAllergyMealFactory instance;
    private final Diet diet = new NutAllergy();

    private NutAllergyMealFactory() {
    }

    @Override
    public Carbs createCarbs() {
        return CarbsFactory.getInstance().getCarbs(this.diet);
    }

    @Override
    public Protein createProtein() {
        return ProteinFactory.getInstance().getProtein(this.diet);
    }

    @Override
    public Fats createFats() {
        return FatsFactory.getInstance().getFats(this.diet);
    }

    public static NutAllergyMealFactory getInstance() {
        if (instance == null) {
            instance = new NutAllergyMealFactory();
        }
        return instance;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient;

import edu.cpp.cs5800.macronutrient.carbs.Carbs;
import edu.cpp.cs5800.macronutrient.carbs.CarbsFactory;
import edu.cpp.cs5800.macronutrient.diet.Diet;
import edu.cpp.cs5800.macronutrient.diet.NoRestriction;
import edu.cpp.cs5800.macronutrient.fats.Fats;
import edu.cpp.cs5800.macronutrient.fats.FatsFactory;
import edu.cpp.cs5800.macronutrient.protein.Protein;
import edu.cpp.cs5800.macronutrient.protein.ProteinFactory;

public class NoRestrictionMealFactory extends MacronutrientFactory {
    private static NoRestrictionMealFactory instance;
    private final Diet diet = new NoRestriction();

    private NoRestrictionMealFactory() {
    }

    @Override
    public Carbs createCarbs() {
        return CarbsFactory.getInstance().getCarbs(this.diet);
    }

    @Override
    public Protein createProtein() {
        return ProteinFactory.getInstance().getProtein(this.diet);
    }

    @Override
    public Fats createFats() {
        return FatsFactory.getInstance().getFats(this.diet);
    }

    public static NoRestrictionMealFactory getInstance() {
        if (instance == null) {
            instance = new NoRestrictionMealFactory();
        }
        return instance;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient;

import edu.cpp.cs5800.macronutrient.diet.Diet;
import edu.cpp.cs5800.macronutrient.diet.NutAllergy;
import edu.cpp.cs5800.macronutrient.diet.Paleo;
import edu.cpp.cs5800.macronutrient.diet.Vegan;

public class MacroNutrientFactoryCreator {
    public static MacronutrientFactory createFactory(Diet diet) {
        return switch (diet) {
            case Vegan vegan -> VeganMealFactory.getInstance();
            case NutAllergy nutAllergy -> NutAllergyMealFactory.getInstance();
            case Paleo paleo -> PaleoMealFactory.getInstance();
            case null, default -> NoRestrictionMealFactory.getInstance();
        };
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.diet;

import java.util.Arrays;
import java.util.List;

public abstract class Diet {
    public abstract List<String> getNonEatableItems();
    @Override
    public String toString() {
        return String.format("%s: Restricted%s", this.getClass().getSimpleName(), Arrays.toString(this.getNonEatableItems().toArray()));
    }
}

```

PART-2 (Macronutrient Meals Implementation)

```java
package edu.cpp.cs5800.macronutrient.diet;

import java.util.List;

public class Paleo extends Diet {
    @Override
    public List<String> getNonEatableItems() {
        return List.of("Cheese", "Bread", "Lentils", "Tofu", "Cheese", "Sour cream");
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.diet;

import java.util.List;

public class Vegan extends Diet {
    @Override
    public List<String> getNonEatableItems() {
        return List.of("Fish", "Chicken", "Tuna", "Cheese", "Sour cream");
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.diet;

import java.util.List;

public class NutAllergy extends Diet {
    @Override
    public List<String> getNonEatableItems() {
        return List.of("Pistachio", "Peanuts");
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.diet;

import java.util.List;

public class NoRestriction extends Diet {
    @Override
    public List<String> getNonEatableItems() {
        return List.of();
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.fats;

import edu.cpp.cs5800.macronutrient.Food;

public abstract class Fats extends Food {
    private final static String type = "Fats";

    public String getType() {
        return type;
    }

}

```

```java
package edu.cpp.cs5800.macronutrient.fats;

public class Tuna extends Fats {
    private final static String name = "Tuna";

    public String getName() {
        return name;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.fats;

public class Avocado extends Fats {
    private final static String name = "Avocado";

    public String getName() {
        return name;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.fats;

public class Peanuts extends Fats {
    private final static String name = "Peanuts";

    public String getName() {
        return name;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.fats;

public class SourCream extends Fats {
    private final static String name = "Sour cream";

    public String getName() {
        return name;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.fats;

import edu.cpp.cs5800.macronutrient.FoodFactory;
import edu.cpp.cs5800.macronutrient.diet.Diet;

import java.util.List;

public class FatsFactory extends FoodFactory {
    private static FatsFactory instance;

    private FatsFactory() {
    }

    public Fats getFats(Diet diet) {
        String foodName = this.getFoodName(diet);
        return createFats(foodName);
    }

    private Fats createFats(String carbsName) {
        switch (carbsName) {
            case "Avocado" -> {
                return new Avocado();
            }
            case "Peanuts" -> {
                return new Peanuts();
            }
            case "Sour cream" -> {
                return new SourCream();
            }
            default -> {
                return new Tuna();
            }
        }

    }

    public List<String> getItems() {
        return List.of("Avocado", "Peanuts", "Sour cream", "Tuna");
    }

    public static FatsFactory getInstance() {
        if (instance == null) {
            instance = new FatsFactory();
        }

        return instance;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.carbs;

public class Bread extends Carbs {
    private final static String name = "Bread";

    public String getName() {
        return name;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.carbs;

import edu.cpp.cs5800.macronutrient.Food;

public abstract class Carbs extends Food {
    private final static String type = "Carbs";

    public String getType() {
        return type;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.carbs;

public class Cheese extends Carbs {
    private final static String name = "Cheese";

    public String getName() {
        return name;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.carbs;

public class Lentils extends Carbs {
    private final static String name = "Lentils";

    public String getName() {
        return name;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.carbs;

public class Pistachio extends Carbs {
    private final static String name = "Pistachio";

    public String getName() {
        return name;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.carbs;

import edu.cpp.cs5800.macronutrient.FoodFactory;
import edu.cpp.cs5800.macronutrient.diet.Diet;

import java.util.List;

public class CarbsFactory extends FoodFactory {
    private static CarbsFactory instance;

    private CarbsFactory() {
    }

    public Carbs getCarbs(Diet diet) {
        String foodName = this.getFoodName(diet);
        return createCarbs(foodName);
    }

    private Carbs createCarbs(String carbsName) {
        switch (carbsName) {
            case "Cheese" -> {
                return new Cheese();
            }
            case "Bread" -> {
                return new Bread();
            }
            case "Lentils" -> {
                return new Lentils();
            }
            default -> {
                return new Pistachio();
            }
        }
    }

    public List<String> getItems() {
        return List.of("Cheese", "Bread", "Lentils", "Pistachio");
    }

    public static CarbsFactory getInstance() {
        if (instance == null) {
            instance = new CarbsFactory();
        }

        return instance;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.protein;

public class Beef extends Protein {
    private final static String name = "Beef";

    public String getName() {
        return name;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.protein;

public class Fish extends Protein {
    private final static String name = "Fish";

    public String getName() {
        return name;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.protein;

public class Tofu extends Protein {
    private final static String name = "Tofu";

    public String getName() {
        return name;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.protein;

public class Chicken extends Protein {
    private final static String name = "Chicken";

    public String getName() {
        return name;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.protein;

import edu.cpp.cs5800.macronutrient.Food;

public abstract class Protein extends Food {
    private final static String type = "Protein";

    public String getType() {
        return type;
    }
}

```

```java
package edu.cpp.cs5800.macronutrient.protein;

import edu.cpp.cs5800.macronutrient.FoodFactory;
import edu.cpp.cs5800.macronutrient.diet.Diet;

import java.util.List;

public class ProteinFactory extends FoodFactory {
    private static ProteinFactory instance;

    private ProteinFactory() {
    }

    public Protein getProtein(Diet diet) {
        String foodName = this.getFoodName(diet);
        return createProtein(foodName);
    }

    private Protein createProtein(String carbsName) {
        switch (carbsName) {
            case "Beef" -> {
                return new Beef();
            }
            case "Chicken" -> {
                return new Chicken();
            }
            case "Fish" -> {
                return new Fish();
            }
            default ->  {
                return new Tofu();
            }
        }
    }

    public List<String> getItems() {
        return List.of("Beef", "Chicken", "Fish", "Tofu");
    }

    public static ProteinFactory getInstance() {
        if (instance == null) {
            instance = new ProteinFactory();
        }

        return instance;
    }
}

```