

Tutorial 1 - Abstract Factory Design Pattern

- 1) Re-structure the following class Meal into a set of classes using the Abstract Factory Pattern.

Hint: Define a CourseFactory interface first that provides methods to make starter, main and dessert courses. The products of all of these methods will be Strings.

```
package factory;

public class Meal {
    private String starter, main, dessert;

    public void setMeal (String meal) {
        if (meal.equals("Vegan")) {
            starter = "asparagus";
            main = "nut cutlet";
            dessert = "fruit";
        } else {
            starter = "liver pate";
            main = "fillet steak";
            dessert = "ice cream";
        }
    }

    public String toString () {
        StringBuffer buffer = new StringBuffer();
        buffer.append("Starter: - ");
        buffer.append(starter);
        buffer.append("\n");
        buffer.append("Main: - ");
        buffer.append(main);
        buffer.append("\n");
        buffer.append("Dessert: - ");
        buffer.append(dessert);
        buffer.append("\n\n");
        return buffer.toString();
    }
}
```

- 2) Extend your set of classes to provide “Omnivore” meals that consist of: “Starter: tomato soup, Main: steak & kidney pie, Dessert: fruit”.