

Using the Abstract Factory Pattern

Imagine you are writing an electronic personal information manager application. As part of this application you need to be able to record and save the phone numbers and addresses of your users. Your users are very international people, and have lots of friends all over the world. You want to be able to record and save phone numbers and addresses of any country.

The challenge is that every country has its own rules governing what constitutes a valid phone number and address. You want your software to validate (make sure the format for that country is correct) all the addresses and phone numbers that your users save for any country. Using the code supplied, create the necessary classes or interfaces as the case may be to add the capability to store a US and French address and phone number using the Abstract Factory pattern.

US Address Format:

Line 1: House Number + **COMMA** + **SPACE** + Street Name

Line 2: City Name + **COMMA** + **SPACE** + State Name + **SPACE** + Zip Code

Line 3: United States

US Phone Format:

Country Code: 01

Number Length: 10 digits

French Address Format:

Line 1: House Number + **COMMA** + **SPACE** + Street Name

Line 2: Postal Code + **SPACE** + City Name

Line 3: France

French Phone Format:

Country Code: 33

Number Length: 9 digits

The code supplied includes client code creating a U.S. phone number and address as well as a French phone number and address.

Follow this diagram:

