Factory Method Pattern

Extending and/or modifying the code and the information given, use the factory method pattern implemented in two ways:

(a) Create three classes that implement the CurrencyCreator interface, one for each of the three factory methods. Look at the associated class diagram for guidance.

(b) Create one class with a static factory method that creates the right Currency object based on the static factory method abbreviation parameter. Look at the associated class diagram for guidance.

Add the following currencies to the code:

1) Japanese Yen
2) US Dollar
3) Australian Dollar

Use the information below to implement the operations of the Currency interface:
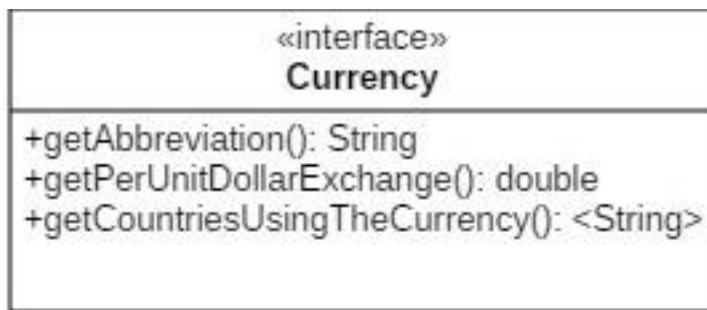
Japanese Yen:

Abbreviation: "JPY"
1.0 JPY = 0.0088 USD
Country using JPY: Japan

US Dollar:

Abbreviation: "USD"

1.0 USD = 1.0 USD

Countries using the USD are:
United States of America
The British Virgin Islands
The British Turks
Caicos islands
Marshall Islands

Republic of Palau
Federated States of Micronesia
Commonwealth of the Northern Mariana Islands
American Samoa
Democratic Republic of Timor-Leste
Virgin islands of the United States
Guam
Republic of Zimbabwe
Republic of El Salvador
Ecuador
Commonwealth of Puerto Rico
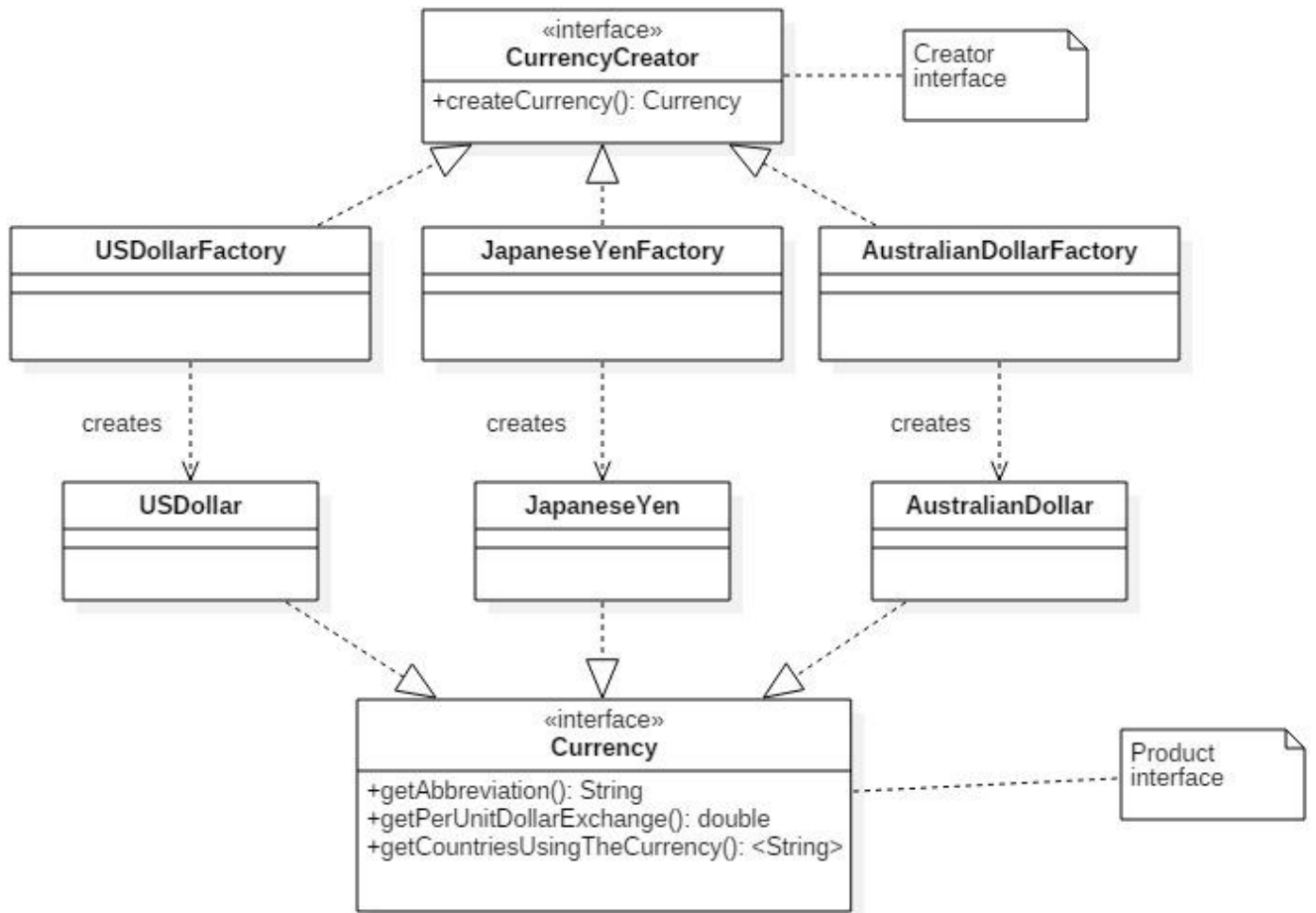
Australian Dollar:

Abbreviation: "AUD"

1.0 AUD = 0.75 USD

Countries using the AUD are:

Australia
Cocos (Keeling) Islands
Kiribati
Norfolk Island
Christmas Island
Heard and McDonald Islands
Nauru
Tuvalu

Implementation (a):

«interface»
CurrencyCreator

+createCurrency(): Currency

Creator interface

USDollarFactory

JapaneseYenFactory

AustralianDollarFactory

creates

creates

creates

USDollar

JapaneseYen

AustralianDollar

«interface»
Currency

+getAbbreviation(): String
+getPerUnitDollarExchange(): double
+getCountriesUsingTheCurrency(): <String>

Product interface

Implementation (b):