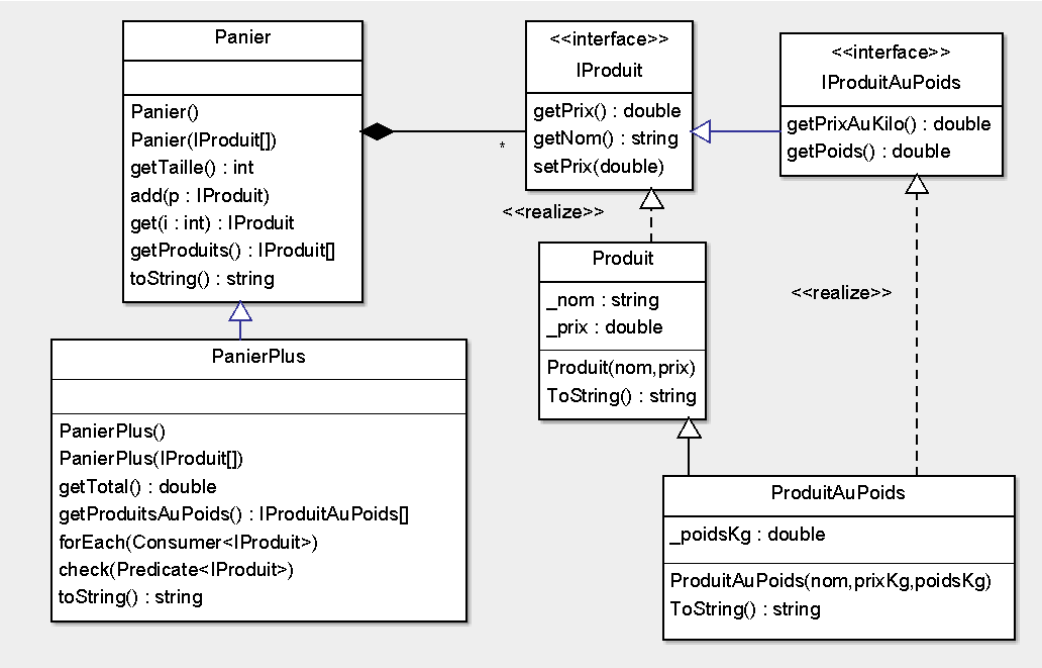
Java TD1 Partie 4 A : langage Java (classes et interfaces)

Packages : pta.sagi.magasin (classes ci-dessous) pta.sagi.ui.CExec (classe exec)



Implémentation : un panier stocke des références à des produits dans une collection de type **`ArrayList<IProduit>`** (tableau de taille modulable)

Note : il convient d'ajouter une classe exécutable (contenant le point d'entrée main)

Mettre en œuvre quelques tests unitaires JUnit (cf. synthèse JUnit)

1) vérifier le nom et le prix d'un produit, vérifier le prix d'un produit au poids

2) vérifier la taille d'un panier vide, d'un panier avec deux produits

3) vérifier le prix total d'un PanierPlus sans produit, d'un panier avec deux produits

5) Diminuer le prix des produits d'un PanierPlus de 10% et vérifier le total.

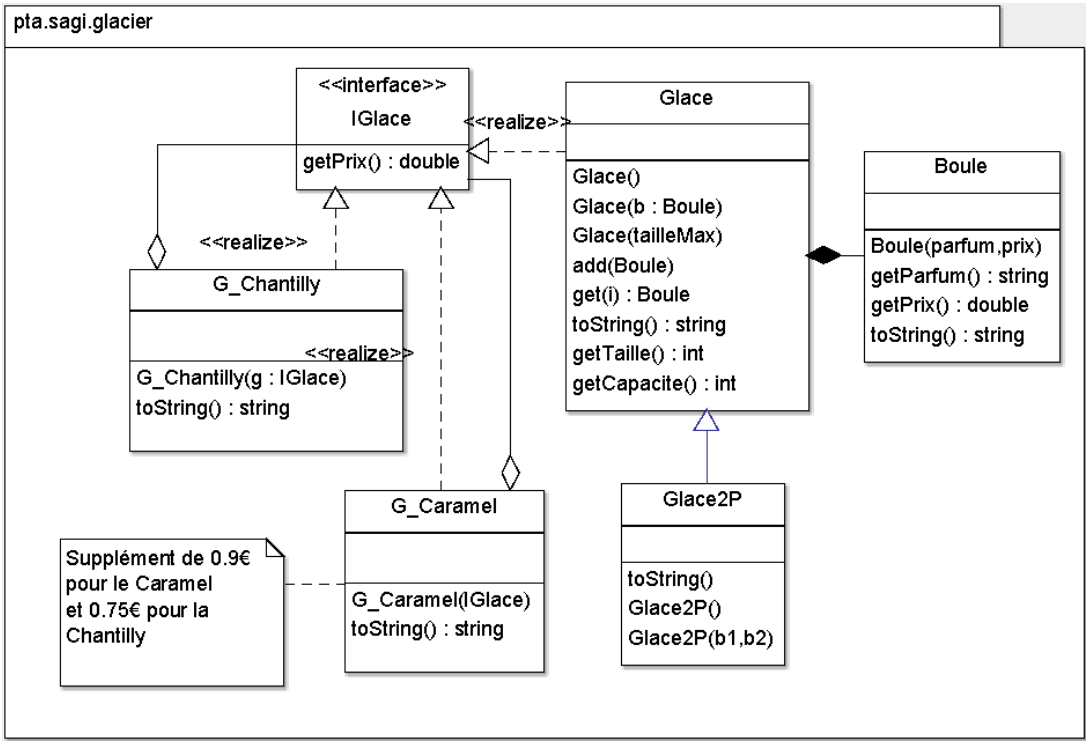
```
package test;
...

public class TestProduitAuPoids {
	@Test
	public void test() {
		Produit prod1 = new Produit("Piles",2.2);
		assertEquals("Piles",prod1.getNom());
		assertEquals(2.2,prod1.getPrix(),0.00001);
		assertTrue(prod1 instanceof IProduit);

		ProduitAuPoids prod2=new ProduitAuPoids("Bananes",2,0.4);
		assertTrue(prod2 instanceof IProduit);
		assertTrue(prod2 instanceof IProduitAuPoids);

		assertEquals("Bananes",prod2.getNom());
		assertEquals(2,prod2.getPrixAuKilo(),0.00001);
		assertEquals(0.8,prod2.getPrix(),0.00001);
		assertEquals(0.4,prod2.getPoidsKg(),0.00001);
	}
}
```

TD1 Partie 4B : projet Java avec classes et interfaces du diagramme suivant



Classe **Boule** : décrit une boule de glace (attributs Parfum (String) Prix (double))

Classe **Glace** : implémente l'interface **`IGlace`**. Stocke des boules de glace dans un tableau de taille fixe. La taille du tableau définit la capacité de la glace (nombre maximal de boules). La taille d'une glace est le nombre effectif de boules (<= capacité)

Constr. sans argument : glace de capacité 3 avec 0 boule
Constr. 1 boule : capacité 1 et taille 1
Constr. tailleMax : capacité tailleMax avec 0 boule
add : ajoute (si possible) une boule ou lèvre une exception
get(i) : fournit la boule d'indice i (ou null si n'existe pas)

Classe **Glace2P** : spécialisation pour les glaces 2 parfums (capacité 2)

Classe **G_Chantilly/G_Caramel** : décrivent une glace avec supplément [un Décorateur].

```
Glace g = new Glace();
g.add(new Boule("Vanille",1.5));
g.add(new Boule("Chocolat",2));

IGlace maglace = g;
System.out.println(maglace);

maglace=new G_Caramel(maglace); //on décore avec un supplément
System.out.println(maglace);
System.out.printf("prix %f \n",maglace.getPrix());
```

```
Glace 2 boules 3,50 €
Glace 2 boules 3,50 €  [+ Caramel 0.9]
prix 4,400000
```

Mettre en œuvre des tests unitaires JUnit:

- Vérifier le prix d'une glace 2 boules
- Vérifier que l'ajout lève une exception quand capacité=taille