

## Exercice 1

Exercice 1 :

```
class Bim:
    def __init__(self, nature, surface, prix_moy):
        self.nt=nature
        self.sf=float(surface)
        self.pm=float(prix_moy)
    def estim_prix(self):
        return self.sf*self.pm
```

Exercice 2 :

La fonction `b1.estim_prix()` renverras une valeur de type float qui est 140000.0

Exercice 3 :

La fonction `estim_prix(self)` raffinée sera :

```
class Bim:
    def __init__(self, nature, surface, prix_moy):
        self.nt=nature
        self.sf=float(surface)
        self.pm=float(prix_moy)
    def estim_prix(self):
        if self.nt=="maison":
            return self.sf*self.pm*1.1
        elif self.nt=="bureau":
            return self.sf*self.pm*0.8
        else:
            return self.sf*self.pm
```

Exercice 4 :

La fonction `nb_maison(lst)` est :

```
def nb_maison(lst):
    nb_maison=0
    for i in range(len(lst)):
        if lst[i]=="maison":
            nb_maison+=1
    return nb_maison
```

Exercice 5 : a)

La liste triée est :

b2,b4,b1,b5,b3,b6

b1	c
b2	g
b3	d
b4	d
b5	g
b6	d

Exercice 5 : b)

La fonction récursive **contient** complétée est :

```
def contient(surface,abr):  
    if abr.est_vide():  
        return False  
    elif abr.get_v().sf>=surface:  
        return True  
    else:  
        return contient(surface,abr+1)
```