

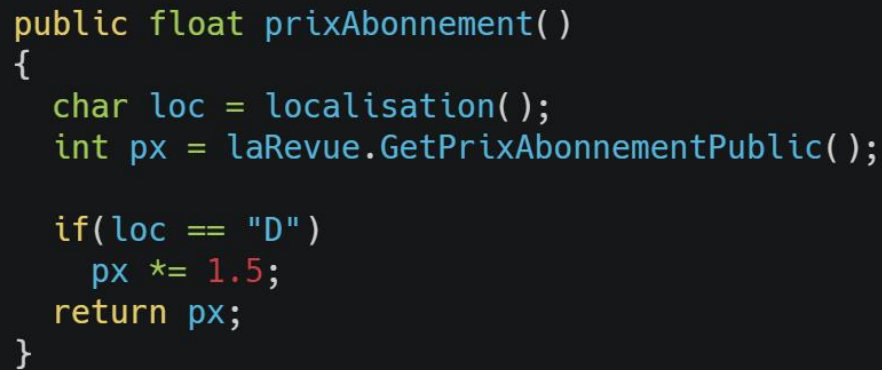
Cas EPOKA PRESS

3.1



```
public char localisation()  
{  
    string code = codePostal.Substring(1,2);  
    char localisation = "M";  
    if(code == "97")  
        localisation = "D";  
    return localisation;  
}
```

3.2



```
public float prixAbonnement()  
{  
    char loc = localisation();  
    int px = laRevue.GetPrixAbonnementPublic();  
  
    if(loc == "D")  
        px *= 1.5;  
    return px;  
}
```

3.3

Class AbonnementMultiple héritant de la classe Abonnement

Attributs privés :

- nbrExemplaire <int>
- tauxRemise <int>

Méthodes de la classe :

- constructeur AbonnementMultiple(num<int>, nom<string>, prenom<string>, raisonSoc<string>, cdp <int>, ville<string>, dtDeb<DateTime>, dtFin<DateTime>, revue<Revue>, nbrExemplaire<int>, tauxRemise<int>)

3.4

```
class AbonnementMultiple : Abonnement
{
    private int nbrExemplaire;
    private int tauxRemise;

    AbonnementMultiple(int num, string nom, string prenom, string raisonSoc, int cdp, string ville,
        DateTime dtDeb, DateTime dtFin, Revue revue, int nbrExemplaire, int tauxRemise)
        : base(num, nom, prenom, raisonSoc, cdp, ville, dtDeb, dtFin)
    {
        this.nbrExemplaire = nbrExemplaire;
        this.tauxRemise = tauxRemise;
    }
}
```

3.5

```
public float prixAbonnement()
{
    return base.prixAbonnement() * nbrExemplaire * (100 - tauxRemise) / 100f;
}
```

3.6



```
public List<Abonnement> relancerAbonnement()  
{  
    List<Abonnement> lA = new List<Abonnement>();  
    int nbJDiff = 0;  
    Abonnement a;  
  
    for(int i = 0; i < lesAbonnements.Count; i++)  
    {  
        a = lesAbonnements[i];  
        nbJDiff = DateTime.Today.Subtract(a.GetDateFin()).Day;  
        if(nbJD > 0 && nbJDiff < 30)  
            la.Add(a);  
    }  
    return lA;  
}
```