

```

001 | # Correction TD01
002 |
003 | import math as m
004 |
005 | #-----
006 | # Exercice 1
007 |
008 | def cercle(rayon):
009 |     """fonction qui reçoit le rayon d'un cercle et renvoie le périmètre et l'aire"""
010 |
011 |     # calcul du périmètre
012 |     périmètre = 2*m.pi*rayon
013 |
014 |     # calcul de l'aire
015 |     aire = m.pi*rayon**2
016 |
017 |     return périmètre,aire
018 |
019 | #-----
020 | # Exercice 2
021 |
022 | def conversion(duree):
023 |     """ Conversion des secondes en hh:mm:ss"""
024 |
025 |     # calcul des heures
026 |     h = duree//3600
027 |
028 |     # calcul des minutes
029 |     m = (duree-3600*h)//60
030 |
031 |     # calcul des secondes
032 |     s = duree%60
033 |
034 |     return h,m,s
035 |
036 | #-----
037 | # Exercice 3
038 |
039 | def equation2(a,b,c):
040 |     """Résolution de l'équation  $ax^2+bx+c=0$ """
041 |
042 |     delta = b**2 - 4*a*c      # calcul du discriminant
043 |
044 |     if a == 0:                # cas d'une équation du 1er degré
045 |         x = -c / b
046 |         return x
047 |
048 |     else :
049 |
050 |         if delta > 0:         # 2 solutions réelles
051 |             x1 = (-b + m.sqrt(delta)) / (2*a)
052 |             x2 = (-b - m.sqrt(delta)) / (2*a)
053 |             return x1,x2
054 |
055 |         elif delta == 0:     # 1 solution double
056 |             x = -b / (2*a)
057 |             return x
058 |
059 |         elif delta < 0:      # pas de solution
060 |             return False
061 |
062 |
063 | #-----
064 | # Exercice 4
065 |
066 | def bissextile(année):
067 |     """Détermine si une année est bissextile"""
068 |
069 |     if (année%4 == 0 and année%100 !=0 ) or (année%400 == 0):
070 |         return True
071 |
072 |     else:
073 |         return False
074 |
075 | #-----

```

```

076 | # Exercice 5
077 |
078 | def calcul_part(marié,nbenfants):
079 |     """marié=True ou False, renvoie le nombre de parts"""
080 |
081 |     # Prise en compte du mariage
082 |     if marié == True:
083 |         nbpart = 2
084 |     else:
085 |         nbpart = 1
086 |
087 |     # Prise en compte du nombre d'enfants
088 |     if nbenfants <=2:
089 |         nbpart = nbpart + nbenfants*0.5
090 |     else:
091 |         nbpart = nbpart + 1 + (nbenfants-2)
092 |
093 |     return nbpart
094 |
095 |
096 | def calcul_impot(revenu,nbpart):
097 |     """calcul l'impôt et le taux d'imposition avec le revenu et le nombre de parts"""
098 |
099 |     # Calcul de la tranche d'imposition
100 |     tranche = 0.9*revenu/nbpart
101 |
102 |     # Calcul de l'impôt pour une part
103 |     if tranche <= 12294:
104 |         impot = 0
105 |     elif tranche <= 28797:
106 |         impot = (tranche-11294)*0.11
107 |     elif tranche <= 82341:
108 |         impot = (28797-11294)*0.11+(tranche-28797)*0.30
109 |     elif tranche <= 177106:
110 |         impot = (28797-11294)*0.11+(82341-28797)*0.30+(tranche-82341)*0.41
111 |     else:
112 |         impot = (28797-11294)*0.11+(82341-28797)*0.30+(177106-82341)*0.41+
(tranche-177106)*0.45
113 |
114 |     # Calcul de l'impôt total
115 |     impot = round(impot*nbpart)
116 |
117 |     # Calcul du taux d'imposition
118 |     taux = round(100*impot/revenu)
119 |
120 |     return impot,taux
121 |

```