

```
//ce programme permet d'étudier le fonctionnement d'une photorésistance
```

```
//zone de définition des constantes ou des variables
```

```
//zone qui contient les instructions d'initialisation
```

```
void setup()  
{  
  Serial.begin(9600);           //vitesse de communication entre le PC et l'Arduino  
}
```

```
//zone qui contient les instructions du programme qui seront répétées indéfiniment
```

```
void loop()  
{  
  int valeur = analogRead(A0);           //la variable valeur (nombre entier) correspond au nombre lu sur la broche A0  
  Serial.print("valeur mesurée en A0:"); //texte affiché sur le moniteur série  
  Serial.println(valeur);               //valeur lue en A0 affichée sur le moniteur série, puis retour à la ligne  
  Serial.print("tension en V:");        //texte affiché sur le moniteur série  
  Serial.println(valeur*0.00488759,5);   //valeur de la tension affichée sur le moniteur série, puis retour à la ligne  
  Serial.println();                      //saut de ligne  
  delay(2000);                           //attente de 2 s avant d'avoir la prochaine mesure  
}
```

```
//ce programme allume une DEL automatiquement lors de la baisse de la luminosité
```

```
//zone de définition des constantes ou des variables
```

```
int led = 3; //led connectée sur la broche 3
```

```
//zone qui contient les instructions d'initialisation
```

```
void setup()  
{  
  Serial.begin(9600); //vitesse de communication entre le PC et l'arduino  
}
```

```
//zone qui contient les instructions du programme qui seront répétées indéfiniment
```

```
void loop()  
{  
  int valeur = analogRead(A0); //la variable valeur (nombre entier) correspond au nombre lu sur la broche A0  
  Serial.print ("valeur mesurée en A0:"); //texte affiché sur le moniteur série  
  Serial.println(valeur); //valeur lue en A0 et affichée sur le moniteur série, puis retour à la ligne  
  Serial.print ("tension en V:"); //texte affiché sur le moniteur série  
  Serial.println(valeur*0.00488759,5); //valeur de la tension affichée sur le moniteur série avec 5 décimales  
  Serial.println(); //saut de ligne  
  if(valeur<90) //si la valeur lue en A0 est inférieure à 90, photorésistance à l'obscurité  
  {  
    digitalWrite(led,HIGH); //la led s'allume  
  }  
  else  
  {  
    digitalWrite(led,LOW); //la led s'éteint  
  }  
  delay(200);  
}
```

```
//ce programme fait sonner un buzzer lors d'une intrusion, lorsque la photorésistance est cachée par l'intrus
```

```
//zone de définition des constantes ou des variables
```

```
int buzzer = 3; //buzzer connecté sur la broche
```

```
//zone qui contient les instructions d'initialisation
```

```
void setup()  
{  
  Serial.begin(9600); //vitesse de communication entre le PC et l'arduino  
}
```

```
//zone qui contient les instructions du programme qui seront répétées indéfiniment
```

```
void loop()  
{  
  int valeur = analogRead(A0); //la variable valeur (nombre entier) correspond a nombre lu sur la broche A0  
  Serial.print ("valeur mesurée en A0:"); //texte affiché sur le moniteur série  
  Serial.println (valeur); //valeur lue en A0 et affichée sur le moniteur série, puis retour à la ligne  
  Serial.print("tension en V:"); //texte affiché sur le moniteur série  
  Serial.println(valeur*0.00488759,5); //valeur de la tension affichée sur le moniteur série avec 5 décimales  
  Serial.println(); //saut de ligne  
  if(valeur<90) //si la valeur lue en A0 est inférieure à 90, photorésistance à l'obscurité  
  {  
    tone(buzzer,494,100); //fonctionnement du buzzer à la fréquence 494 Hz pendant 0,1s  
    delay(200); //pendant 200 mS  
    tone(buzzer,440,100); //fonctionnement du buzzer à la fréquence 440 Hz pendant 0,1s  
    delay(200); //pendant 200 mS  
    tone(buzzer,494,100); //fonctionnement du buzzer à la fréquence 494 Hz pendant 0,1s  
    delay(200); //pendant 200 mS  
    tone(buzzer,440,100); //fonctionnement du buzzer à la fréquence 440 Hz pendant 0,1s  
    delay(200); //pendant 200 mS  
  }  
  else{}  
}
```