

4- PHP et GPIO sur raspberry pi

On veille bien à ce qu'il y ait un serveur php et apache2 sur notre pi. (cf [3-Héberger des pages web sur le pi](#))

On veille à donner les droits à tout le monde au fichier value: `chmod 777 value`.

IMPORTANT: pour que le serveur PHP puisse accéder aux GPIO, il faut ajouter le groupe "www-data" au groupe "gpio" et rebooter la carte:

```
sudo addgroup www-data gpio`
```

On écrit la page php suivante dans `/var/www/html/Site1`:

```
<!--index.php-->

<!DOCTYPE html>
<html lang="fr">
<head>
    <meta charset="UTF-8">
    <title>Commande LED D7</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <h1> Commande LED D7 </h1>
    <form name="LEDD7" method="post" action="index.php">
        ON: <input type="radio" name="option" value="on"/><br/>
        OFF: <input type="radio" name="option" value="off"/><br/>
        <input type="submit" name="valider" value="OK"/>
    </form>
        <!-- On crée un formulaire avec la balise form: on a 2 boutons radio et un bouton d envoie de la requête POST -->

    <!-- Partie PHP-->
<?php
// On récupère les valeurs du formulaire
if(isset($_POST['valider'])){
    system("echo out > /sys/class/gpio/gpio19/direction");
    if(isset($_POST['option'])){
        $optionSelec = $_POST['option'];
        if($optionSelec === 'on'){
            system("echo 0 > /sys/class/gpio/gpio19/value");
            echo "<p> LED allumée </p>";
        }
        elseif ($optionSelec === 'off') {
            system("echo 1 > /sys/class/gpio/gpio19/value");
            echo "<p> LED éteinte </p>";
        }
    }
    // bouton poussoir
    $etatBP = array();
    exec("cat /sys/class/gpio/gpio27/value", $etatBP, $return);
    if(trim($resul[0])=="1")
        echo '<p> le bouton poussoir est ON </p>';
    else
        echo '<p> le bouton poussoir est OFF </p>';
```

```

        }
    ?>
</body>
</html>
```

Explications:

1. Formulaire HTML:

- Un formulaire est créé avec deux boutons radio (ON et OFF) et un bouton de validation.
- Lorsque le formulaire est soumis (via le bouton "OK"), les données sont envoyées en méthode POST à la même page (`index.php`).

2. Partie PHP:

- Le code PHP commence par vérifier si le bouton "OK" a été soumis (`isset($_POST['valider'])`).
- Ensuite, il configure le GPIO 19 comme sortie en utilisant la commande système `echo out > /sys/class/gpio/gpio19/direction`.
- Il récupère la valeur sélectionnée dans les boutons radio (ON ou OFF) et contrôle la LED en conséquence :
 - Si "ON" est sélectionné, il écrit `0` dans `/sys/class/gpio/gpio19/value` pour allumer la LED et affiche un message indiquant que la LED est allumée.
 - Si "OFF" est sélectionné, il écrit `1` dans `/sys/class/gpio/gpio19/value` pour éteindre la LED et affiche un message indiquant que la LED est éteinte.
- Enfin, il lit l'état d'un bouton poussoir connecté au GPIO 27 en utilisant la commande `cat /sys/class/gpio/gpio27/value` et affiche un message indiquant si le bouton poussoir est activé (ON) ou désactivé (OFF).

Et le code css:

```

body {
    margin: 10px;
    padding: 0;
    background-image: url('fond.png');
    background-size: cover;
    color: white;
    font-family: 'Courier New', Courier, monospace;
    display: flex;
    flex-direction: column;
    justify-content: center;
    align-items: center;
}

p{
    font-weight: 600;
    font-size: 20px;
}

form{
    font-weight: 600;
    font-size: 20px;
}
```

Explications

1. Sélecteur `body`:

- `margin: 10px;`: Ajoute une marge de 10 pixels autour du corps de la page pour créer un espace avec les bords du navigateur.
- `padding: 0;`: Aucun espace interne (remplissage) pour le corps de la page.
- `background-image: url('fond.png');`: Utilise l'image 'fond.png' comme arrière-plan de la page.
- `background-size: cover;`: Ajuste la taille de l'image de fond pour couvrir l'intégralité de la page.
- `color: white;`: Définit la couleur du texte de la page en blanc.
- `font-family: 'Courier New', Courier, monospace;`: Spécifie la police de caractères utilisée pour le texte, avec une préférence pour 'Courier New' et 'Courier', suivie de toute police générique monospace.
- `display: flex;`: Utilise le modèle de disposition flexible (Flexbox) pour le positionnement des éléments enfants.
- `flex-direction: column;`: Aligne les éléments enfants verticalement, les uns au-dessus des autres.
- `justify-content: center;`: Centre les éléments enfants verticalement dans le conteneur Flexbox.
- `align-items: center;`: Centre les éléments enfants horizontalement dans le conteneur Flexbox.

2. Sélecteur `p` (paragraphe HTML):

- `font-weight: 600;`: Définit l'épaisseur de la police à 600 (ce qui est généralement interprété comme "gras").
- `font-size: 20px;`: Fixe la taille de la police à 20 pixels.

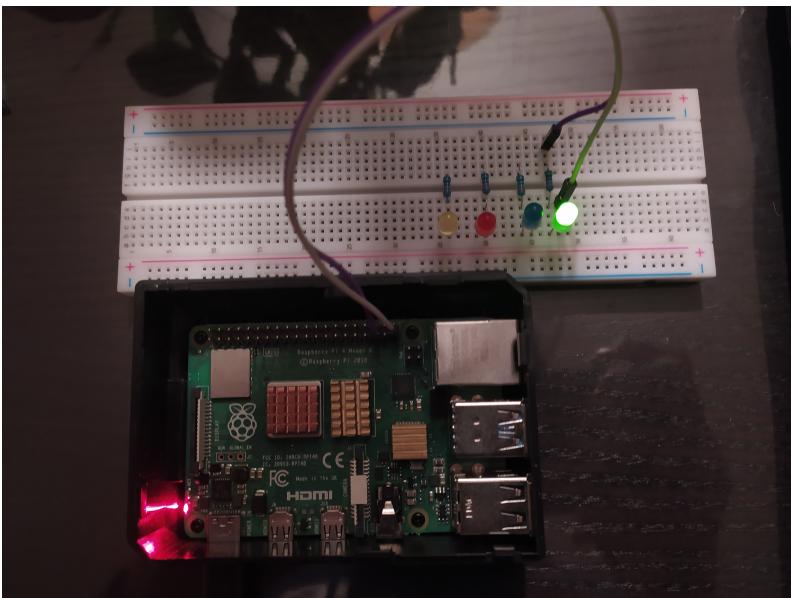
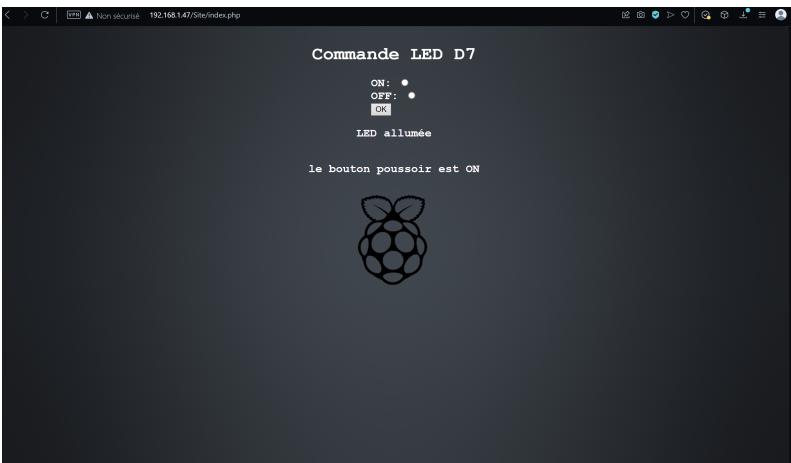
3. Sélecteur `form` (formulaire HTML):

- `font-weight: 600;`: Définit l'épaisseur de la police du formulaire à 600 (gras).
- `font-size: 20px;`: Fixe la taille de la police du formulaire à 20 pixels.

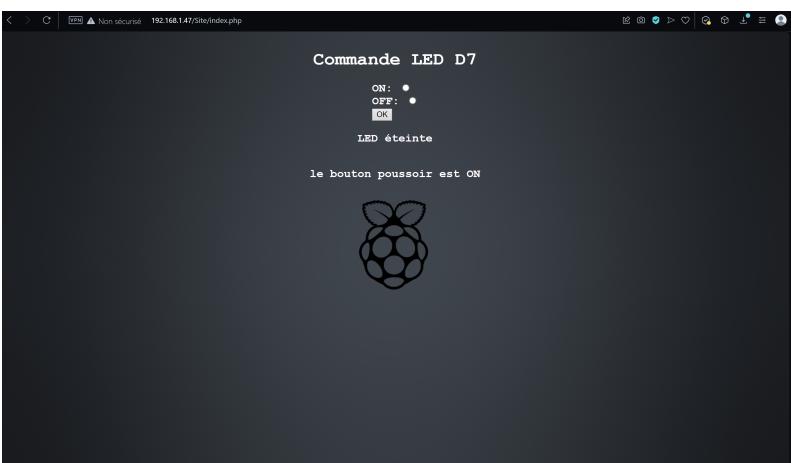
En résumé, ces règles de style définissent l'apparence générale de la page web, notamment l'arrière-plan, la couleur du texte, la police, et les dispositions flexibles pour le centrage des éléments. Les règles spécifiques pour les paragraphes et les formulaires ajustent l'épaisseur et la taille de la police pour ces éléments particuliers.

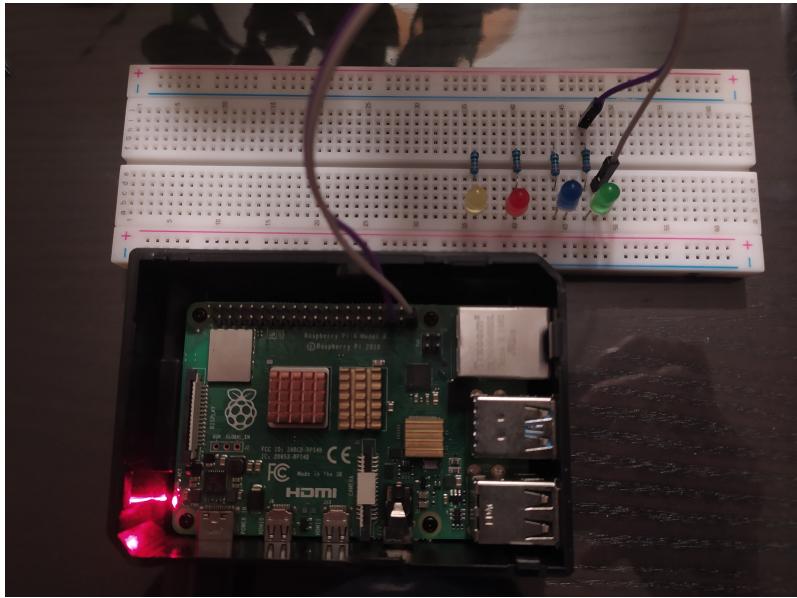
On obtient le résultat suivant pour les LEDs:

Led allumée



Led éteinte





On peut faire de même avec les autres LEDs en modifiant les fichiers value des GPIO.

Maintenant il est donc possible de manipuler les GPIO à distance grâce à une interface web de la même manière que [1.1- Pilotage de GPIO en ligne de commande](#). On peut même lancer des programmes en C pour des applications plus complexes. Il faudra bien veiller à ce que les permissions soient accordées.