

***INSTITUTO TECNOLÓGICO SUPERIOR  
DE***

***CHICONTEPEC***

***INGENIERÍA EN: SISTEMAS  
COMPUTACIONALES.***

***MATERIA: PROGRAMACIÓN WEB.***

***TEMA: RESUMEN DEL FUNCIONAMIENTO  
DEL MÉTODO POST Y GET.***

***DOCENTE: ING. EFREN FLORES CRUZ.***

***ALUMNO: JOSÉ RICARDO MIGUELES  
GUERRA.***

***GRUPO: "ISC-8"***

***FECHA DE ENTREGA: 03 DE ABRIL 2020.***

Nombre: Resumen Funcionamiento del

Día

Mes

Año

Folio

Tema: Método Post y Get

### Método Get

El método GET envía la información codificada del usuario en el header del HTTP request, directamente en la URL. La página web y la información codificada se separan por un interrogante.

www.ejemplo.com/index.htm?Key1=Value1&Key2=Value2&Key3=Value3...

- El método GET envía la información en la propia URL, estando limitado a 2000 caracteres.
- La información es visible por lo que con este método nunca se envía información sensible.
- No se pueden enviar datos binarios (archivos, imágenes).
- En PHP los datos se administran con el array asociativo `$_GET`.

Ejemplo sencilla de formulario HTML con el método GET.

```
<html>
```

```
<body>
```

```
<form action="formgen.php" method="get">
```

```
Nombre: <input type="text" name="nombre"> <br>
```

```
email: <input type="text" name="email"> <br>
```

```
<input type="submit" value="Enviar">
```

```
</form>
```

```
Hola <?php isset($_GET["nombre"]) ? print $_GET
```

```
["nombre"] : " " : "> <br>
```

```
Tu email es: <?php isset($_GET["email"]) ? print $_GET
```

```
["email"] : " " : ">
```

```
</body>
```

```
</html>
```

La url que resulta de hacer click en submit es de la forma:

formget.php?nombre=peter&email=peter240ejemplo.com  
En este caso @ es un carácter especial y se codifica.

## Metodo Post

Con el metodo post tambien se codifica la información, pero esta se envia a través del body del HTTP Request por lo que aparece la url.

- El metodo Post no tiene límite de cantidad de información a enviar.
- La información proporcionada no es visible, porque se puede enviar información sensible.
- Se puede usar para enviar texto normal así como datos binarios (archivos, imágenes...)
- PHP proporciona el array asociativo `$_POST` para acceder a la información enviada.

```
<html>
```

```
<body>
```

```
<form action="formpost.php" method="post">  
  nombre: <input type="text" name="nombre"> <br>  
  email: <input type="text" name="email"> <br>  
  <input type="submit" value="Enviar">  
</form>
```

```
Hola <?php isset($_POST["nombre"]) ? print $_POST["nombre"]  
: "<?>" > <br>
```

```
Tu email es: <?php isset($_POST["email"]) ?  
</body>
```

```
</html>
```

se puede mostrar la información no se muestra en la url