

## Problema "Foo Bar"

En esta parte de la prueba, se te solicita resolver el problema "Foo Bar". Se te pide escribir un programa que realice lo siguiente:

Dado un número entero positivo  $n$ , el programa debe realizar las siguientes acciones:

- Si el número es divisible por 3, en lugar del número, imprimir "Foo".
- Si el número es divisible por 5, en lugar del número, imprimir "Bar".
- Si el número es divisible tanto por 3 como por 5, imprimir "FooBar".
- Si el número no cumple ninguna de las condiciones anteriores, imprimir el número mismo.

Ejemplo de entrada y salida:

```
Entrada: 9
Salida: Foo
Entrada: 10
Salida: Bar
Entrada: 15
Salida: FooBar
Entrada: 16
Salida: 16
```

## Validación de Número Feliz

En esta parte de la prueba, se te solicita escribir una función que determine si un número dado es un "número feliz" o no, de acuerdo con las reglas descritas en el enunciado original.

Se te pide implementar la función `es_numero_feliz(n)` que recibe como parámetro un número entero positivo  $n$  y devuelve `True` si el número es feliz y `False` si no lo es.

Recuerda que un número es feliz si, al seguir el proceso de reemplazar el número por la suma de los cuadrados de sus dígitos de forma repetida, eventualmente se llega al número 1, o si se entra en un ciclo infinito que no incluye al número 1.

Ejemplo:

```
es_numero_feliz(19) => True
es_numero_feliz(4)  => False
```

## Prueba Técnica para Test Automation Engineer - Nivel Medio

### Objetivo

El objetivo de esta prueba técnica es evaluar tus habilidades en la creación de pruebas automatizadas utilizando el marco de pruebas Selenium WebDriver con el lenguaje de programación Python.

### Requisitos previos

- Asegúrate de tener instalados los siguientes elementos:
  - Python y pip (última versión estable)
  - Selenium WebDriver (asegúrate de tener el controlador adecuado para el navegador que utilizarás, por ejemplo, Chromedriver)
  - Un entorno de desarrollo integrado (IDE) para escribir el código (puedes usar PyCharm, Visual Studio Code, etc.)

### Instrucciones

- Utiliza la página web DemoQA.com (<https://demoqa.com/>) para realizar las pruebas automatizadas.
- Crea un nuevo proyecto de automatización y configura el entorno con los requisitos previos mencionados.
- Utiliza el patrón Page Object Model (POM) para organizar tus pruebas y páginas web.
- Implementa las pruebas automatizadas para las siguientes funcionalidades:
  - Caso de prueba 1: Verificar el formulario de registro
    - Navega a la sección "Forms" en el menú y selecciona "Practice Form".
    - Llena el formulario con la siguiente información:
      - Nombre: [Tu nombre]
      - Apellido: [Tu apellido]
      - Correo electrónico: [Tu dirección de correo electrónico]

- Género: [Selecciona "Male" o "Female"]
- Número de teléfono: [Tu número de teléfono]
- Fecha de nacimiento: [Selecciona una fecha]
- Envía el formulario y verifica que se muestre un mensaje de éxito con los detalles del formulario enviado.
- Caso de prueba 2: Verificar la funcionalidad de los enlaces
  - Navega a la sección "Elements" en el menú y selecciona "Links".
  - Haz clic en cada uno de los enlaces disponibles y verifica que se cargue la página correcta y que el contenido sea visible.
  - Obtén todos los links, y clickea el primer link en orden alfabético.
- Caso de prueba 3: Verificar la funcionalidad del carrusel
  - Navega a la sección "Slider" en el menú.
  - Arrastra el control deslizante (slider) hasta la posición 3 y verifica que el número mostrado sea "3".

## **Entrega:**

Crea un repositorio en una plataforma de alojamiento de código (por ejemplo, GitHub) y comparte el enlace con nosotros para revisar tu código. Agrega un archivo README.md al repositorio que contenga instrucciones claras sobre cómo configurar y ejecutar las pruebas automatizadas.

## **Importante:**

Se valorará el uso de buenas prácticas de programación, como comentarios, nombres de variables significativos, y la utilización del patrón Page Object Model. La prueba debe ejecutarse sin errores y los casos de prueba deben pasar correctamente.

Asegúrate de que el código esté bien organizado y sea fácil de entender.

## **Tiempo estimado:**

Aproximadamente 3 horas para completar todos los casos de prueba.

Recuerda que esta prueba técnica tiene como objetivo evaluar tus habilidades en pruebas automatizadas y el uso de herramientas como Selenium WebDriver con Python. ¡Buena suerte! Si tienes alguna pregunta o necesitas aclaraciones adicionales, no dudes en hacerlas.

