



## ISIS-1221 INTRODUCCIÓN A LA PROGRAMACIÓN

### Nivel 3 – Laboratorio1. Instrucciones Iterativas – While

#### ***Objetivo general***

---

El objetivo general de este laboratorio es conocer y practicar el uso de instrucciones iterativas. En especial utilizar la estructura `while` en varios puntos de un programa, incluyendo la interfaz por consola.

#### ***Actividad 1: Entendiendo la instrucción `while`***

---

1. Vea y analice la información contenida en los siguientes videos:
  - Básico: <https://www.youtube.com/watch?v=TCmU2-3oPAA>
  - Intermedio: [https://www.youtube.com/watch?v=u6Hqs0bL\\_Ew](https://www.youtube.com/watch?v=u6Hqs0bL_Ew). Asegúrese de conservar las buenas prácticas de programación que hacen parte de este curso.

#### ***Actividad 2: Reino de los Pares***

---

Un número entero es un rey par si tanto la suma de sus dígitos como su propio valor son pares. Si solamente la suma de sus dígitos es par, se dice que es un noble par, mientras que, si solamente su propio valor es par, se conoce como un guerrero par. De lo contrario, el número es un rebelde impar.

Escriba una función llamada `dar_clase_numero()`, que reciba como parámetro un número entero, evalúe las condiciones anteriormente descritas y retorne un string con la clase a la cual pertenece dicho número.

Ejemplos:

- 222 es un “Rey par”.
- 123 es un “Noble par”.
- 612 es un “Guerrero par”.
- 12345 es un “Rebelde impar”



## ISIS-1221 INTRODUCCIÓN A LA PROGRAMACIÓN

### **Actividad 3: Juego del PUM**

---

Complete la función `jugar_PUM()` que se encuentra en archivo `modulo.py`, siguiendo las instrucciones:

- Para jugar al PUM los N jugadores se sientan en círculo y van diciendo números consecutivos a partir del 1 (1, 2, 3, ...). Para el juego se escoge un número X menor que 10 (por ejemplo 7) y a la persona a la que le corresponda decir un múltiplo de ese número (en el caso del 7: 7, 14, 21, 28, ...), debe decir, en vez del número, la palabra PUM. Escriba una función que escriba, a partir del número de jugadores N y del número escogido para el PUM, X, el desarrollo del juego para los primeros 500 números, indicando el número del jugador y lo que dijo. Por ejemplo, si son 3 jugadores y X es el 4 el juego comenzaría así:

Jugador	Jugada
1	1
2	2
3	3
1	pum
2	5
3	6
1	7
2	Pum...

### **Actividad 4: Adivinar un número de 1 a 9.**

---

Complete la función `adivinar_numero()` que se encuentra en el archivo `modulo.py`. Su función permite adivinar un número que el sistema eligió de 1 a 9.

- El número ingresado por el usuario debe compárarlo con la opción (el número) que el sistema eligió aleatoriamente mediante la función `randint(a,b)` del módulo `random()`. Si el usuario NO adivina el número indíquele que debe volver a ingresar otra opción (nuevo número del 1 al 9). Su programa debe continuar ejecutándose hasta que el usuario adivine el número elegido por el sistema.

Nota: Si tiene dudas acerca de la función `randint(a,b)`, por favor diríjase a al laboratorio “Nivel 2: Laboratorio condicionales & módulos `random` y `math`”. La documentación completa se puede consultar en: <https://docs.python.org/3/library/random.html>



## ISIS-1221 INTRODUCCIÓN A LA PROGRAMACIÓN

### ***Actividad 5: Construcción de la interfaz de usuario***

---

Estudie el archivo `consola.py` y complete de la consola de su programa de acuerdo con la documentación brindada. Verifique como utilizar `while` en la implementación de su consola.

### ***Entrega***

---

Cree un archivo comprimido `.zip` con los dos archivos correspondientes a los programas que escribió o modificó anteriormente. Entregue el archivo comprimido a través del BloqueNeon en el laboratorio del Nivel 3 designado como “Nivel 3 – Laboratorio1. Instrucciones Iterativas - while”.