**GUÍA DE APRENDIZAJE Nº 2**

1. **IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Programa de Formación:  ANALISIS Y DESARROLLO DE SISTEMAS DE INFORMACION | Código:  Versión: | 228106  102 | | |
| Nombre del Proyecto:  IMPLEMENTACION DE UN APLICATIVO WEB PARA EL SEGUIMIENTO Y EVALUACION DE LA ETAPA PRODUCTIVA DE LOS APRENDICES SENA A TRAVES DE UNA BITACORA VIRTUAL | Código: | 1107755 | | |
| Fase del proyecto: | | ANALISIS | | |
| Actividad (es) del Proyecto:  Analizar el sistema | Actividad (es) de Aprendizaje:   * Implementar programas con la instrucción condicional if/then/else | Ambiente de formación ESCENARIO Aula 2030A | MATERIALES DE FORMACIÓN | |
| DEVOLUTIVO Televisor, tablero | CONSUMIBLE (unidades empleadas durante el programa) |
| **Resultados de Aprendizaje:**  Construir el modelo conceptual del macrosistema frente a los requerimientos del cliente, | Competencia:  Analizar los requerimientos del cliente para construir el sistema de información. | | | |
| Duración de la guía ( en horas): | 24 | | | |

1. **INTRODUCCIÓN**

|  |
| --- |
| Las instrucciones iterativas o los también llamados bucles, permiten ejecutar una serie de instrucciones repetidas veces, siempre y cuando una condición se cumpla. Python cuenta con dos instrucciones cíclicas: **while** y **for.** |

1. **ESTRUCTURACION DIDACTICA DE LAS ACTIVIDADES DE APRENDIZAJE**

|  |
| --- |
| * 1. **Actividades de Reflexión inicial.** * Que ventajas ofrece python frente a otros lenguajes de programación * Que se requiere para trabajar con python |
|  |

|  |
| --- |
| * 1. **Actividades de transferencia del conocimiento (Conceptualización y Teorización).**   + **Instrucción while**   **Sintaxis:**  **while** condición:  Instrucción 1  Instrucción 2  Instrucción n  La instrucción **while** evalúa la condición, si el resultado de esta es verdadero, se ejecuta la o las instrucciones que se encuentran dentro del bloque **while**. Cuando se ejecuta la última instrucción que se encuentra dentro de dicho bloque, se evalúa nuevamente la condición y así sucesivamente, hasta que el resultado de la condición sea falsa  **Ejemplo 1:** generar y mostrar los números del 1 al 10  i = 1  while i <= 10:  print (i)  i = i + 1  **Ejemplo 2:** generar una tabla de multiplicar del 1 al 10  mcando = int(input("Ingrese numero de tabla: "))  producto, mcador = 0,1  while(mcador <= 10):  poducto = mcando \* mcador  print(mcando, " x ", mcador, " = ", mcando \* mcador)  mcador = mcador + 1   * Realice los siguientes programas con el ciclo while   + Leer un número y determinar si es par o impar. El programa debe ejecutarse mientras el número sea diferente de cero(0)   + Elaborar un algoritmo repetitivo que permita encontrar el área de un triángulo. Para terminar el algoritmo se le debe preguntar al usuario cuando terminar; es decir, una vez se han pedido la base y la altura al usuario, calculado el área y mostrado el resultado por pantalla, se le debe preguntar al usuario si desea continuar, si el usuario digita ‘s’ o ‘S’, el algoritmo continuara repitiéndose.   + Desarrollar un algoritmo que permita la entrada de un numero de un digito y escriba su valor en letras, el algoritmo se debe repetir hasta que se ingrese una letra. Ejemplo: si el usuario digita el número 5, el programa debe mostrar la palabra cinco por pantalla.   + Generar la secuencia de Fibonacci   + **Instrucción for**   **Sintaxis:**  **for** variable in elemento iterable (lista,cadena,range,etc):  Instrucción 1  Instrucción 2  Instrucción n  El ciclo **for** en el lenguaje python, se diferencia de otros lenguajes; en python esta instrucción se repite en una secuencia de objetos como una lista.  **Ejemplo 1:** En el siguiente código, el ciclo for ejecuta 10 iteraciones, mostrando lo números del 0 al 9  0  1  2  3  4  5  6  7  8  9  Salida  for i in range(10):  print (i)    **Ejemplo 2:** En el siguiente código se define un rango de n iteraciones; las cuales inician en 1 y van hasta el 9  for i in range(1,10):  print (i,end=" ")  Salida  1 2 3 4 5 6 7 8 9  Como se puede apreciar solo se muestran los números del 1 al nueve, ya que la función **range** omite el ultimo elemento.  También se ilustra el uso del parámetro **end** de la función **print**; por defecto esta función termina con un salto de línea “\n”, de manera que para cambiar el salto de línea por cualquier otro carácter se utiliza el parámetro **end**  **Ejemplo 3:** En el siguiente código se define un rango de n iteraciones; las cuales inician en 0 y van hasta n, donde n es una variable que se solicita por consola  veces = int(input("Cuantos numeros desea generar: "))  acum = 0  for i in range(veces):  print (i)  acum = acum + i  print("La sumatoria es = ", acum)  **Ejemplo 4:** En el siguiente código se define un rango de n iteraciones; las cuales inician en 2 y van hasta 10, y la variable i (contador) se incrementa de 2 en 2  for i in range(2,10,2):  print (i)  **Ejemplo 5 :** En el siguiente código, el ciclo **for** recorre cada elemento de la [lista](Listas.docx) y los despliega por consola  for color in ["Blanco","Azul","Amarillo"]:  print (color)  **Ejemplo 6:** En el siguiente código, se itera sobre la cadena de texto “SENA REGIONAL TOLIMA”, y se muestran por consola cada una de las letras de esta cadena de texto  for letra in "SENA REGIONAL TOLIMA":  print (letra)  **Ejemplo 7:** En el siguiente código se ejecuta cinco veces, una iteración por cada elemento de la lista.  for i in [0,1,2,3,4,5]:  print ("Hola ",end="") |
| * 1. **Actividades de apropiación del conocimiento.**   **Ejercicios propuestos:**   * Realice los siguientes programas con el ciclo while   + Leer un número y determinar si es par o impar. El programa debe preguntar en cada iteración si desea continuar   + Elaborar un algoritmo repetitivo que permita encontrar el área de un triángulo. Para terminar el algoritmo se le debe preguntar al usuario cuando terminar; es decir, una vez se han pedido la base y la altura al usuario, calculado el área y mostrado el resultado por pantalla, se le debe preguntar al usuario si desea continuar, si el usuario digita ‘s’ o ‘S’, el algoritmo continuara repitiéndose.   + Desarrollar un algoritmo que permita la entrada de un numero de un digito y escriba su valor en letras, el algoritmo se debe repetir hasta que se ingrese una letra. Ejemplo: si el usuario digita el número 5, el programa debe mostrar la palabra cinco por pantalla.   + Generar la secuencia de Fibonacci * Realice los siguientes programas con el ciclo for   + Encontrar el factorial de un numero   + Generar la secuencia de Fibonacci   + Genarar una tabla de multiplicar * Usando ciclos anidados con las instrucciones while y for , implemente un programa que genere varias tablas de multiplicar las cuales deben ser ingresadas por el usuario e irán desde tabla1 hasta tablan, para n líneas |
| * 1. **Actividades de evaluación.**  |  |  |  | | --- | --- | --- | | **Evidencias de Aprendizaje** | **Criterios de Evaluación** | **Técnicas e Instrumentos de Evaluación** | | **Evidencias de Conocimiento :**   * Respuesta a preguntas.   **Evidencias de Desempeño:**   * Desarrollo de programas Java   **Evidencias de Producto:**   * Programas java | * Identifica y describe, en un sistema de información dado, los datos de entrada, * procesamiento de los datos e información generada, según necesidades del cliente. | * Pruebas de conocimiento * Lista de chequeo | |

1. **RECURSOS PARA EL APRENDIZAJE**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ACTIVIDADES DEL PROYECTO** | **DURACIÓN  (Horas)** | **Materiales de formación devolutivos: (Equipos/Herramientas)** | | **Materiales de formación (consumibles)** | | **Talento Humano (Instructores)** | | **AMBIENTES DE  APRENDIZAJE TIPIFICADOS** |
| **Descripción** | **Cantidad** | **Descripción** | **Cantidad** | **Especialidad** | **Cantidad** | **ESCENARIO (Aula, Laboratorio, taller, unidad productiva)** y elementos y condiciones de seguridad industrial, salud ocupacional y medio ambiente |
|  |  |  |  |  |  |  |  |  |

|  |
| --- |
| **JDK**  **J2SE**  **Java** |

1. **GLOSARIO DE TERMINOS**
2. **REFERENTES BIBLIOGRÁFICOS**

|  |
| --- |
| Harvey M. Deitel, Paul J. Deitel (2004). Como Programar en Java, Pearson Education.  [Froufe Quintas, Agustín](http://biblioteca.sena.edu.co/F/8IVEUS4EGGBXJ5SGUXR2QRQLL3TFQ63B6GFACX6M9J3XX3KCS5-24793?func=full-set-set&set_number=009595&set_entry=000006&format=999) (2009). [Java 2 : manual de usuario y tutorial](http://biblioteca.sena.edu.co/F/8IVEUS4EGGBXJ5SGUXR2QRQLL3TFQ63B6GFACX6M9J3XX3KCS5-24794?func=full-set-set&set_number=009595&set_entry=000006&format=999), Alfaomega Ra-Ma  Flórez Fernández, Héctor Arturo (2012). Programación Orientada a Objetos Usando Java,  [Ecoe Ediciones](http://www.digitaliapublishing.com.bdigital.sena.edu.co/s/e/179).  <http://www.digitaliapublishing.com.bdigital.sena.edu.co/a/30030/programacion-orientada-a-objetos-con-java> |

1. **CONTROL DEL DOCUMENTO (ELABORADA POR)**

|  |
| --- |
| **Ing. Luis Fernando Corredor Mora** |