|  |  |  |
| --- | --- | --- |
| | GUÍA 2.5.2:Guía Ejercicios Experiencia 2 | | |
| Sigla | Asignatura | Experiencia de Aprendizaje |
| FPY1101 | Fundamentos de Programación | EA2: Programación de aplicaciones en Python |
| Tiempo | Modalidad de Trabajo | Indicadores de logro |
| 3 h | Individual | IL 2.1 al IL 2.5 |

|  |
| --- |
| **Código QR con relleno sólido**  **Antecedentes generales** |

## Esta guía tiene como objetivo conocer los aspectos generales en Python

### Ser una guía paso a paso para la construcción de programas en Python

### Debate con tu docente las respuestas obtenidas, si tienes dudas, recuerda dar aviso y argumentar, los programas no tienen soluciones únicas, teniendo derivaciones o mecanismos distintos de funcionamiento.

### Esta guía puede ser desarrollada en casa, o guiada por el docente, con el fin de seguir un paso a paso y comprender las explicaciones de la o el docente.

|  |
| --- |
| **Lista con relleno sólido Requerimientos para esta actividad** |

## Para el desarrollo de esta actividad deberás disponer de:

## Computador

## Visual Studio Code

|  |  |
| --- | --- |
| **Inteligencia artificial con relleno sólidoActividad** |  |

**EJERCICIO 1**

Ordenar los bloques de código.

print("Seleccione el producto a canjear")

print("1.- Gift Card de $10.000, valor de 10.000 puntos")

print("2.- Secadora de pelo, valor de: 25.000 puntos")

print("3.- Disco duro portátil, valor de: 30.000 puntos")

print("1. Ver mis puntos")

print("2. Gastar mis puntos")

print("3. Salir")

op=int(input("Seleccione una opción: "))

continu = int(input("Seleccione la opción"))

if continu==1:

puntos = puntos-10000

print(f"Canje exitoso, le quedan: ${puntos} puntos")

if continu==2:

puntos = puntos-25000

print(f"Canje exitoso, le quedan: ${puntos} puntos")

if continu==3:

puntos = puntos-30000

print(f"Canje exitoso, le quedan: ${puntos} puntos")

#PuntosAcumulados

sw = 1

puntos = 100000

while sw==1:

|  |  |  |
| --- | --- | --- |
| Pregunta | Acomodar el sistema en el siguiente orden, para que permita:   1. Partir con un saldo de 100000 puntos 2. Sistema debe iterar en sentencia While 3. Mostrar un menú de ver mis puntos, gastar mis puntos y salir 4. Dependiendo de la opción seleccionada, muestre los puntos disponibles, descuente los puntos dependiendo del producto seleccionado, y permita salir 5. Cuando se selecciona la opción de gastar mis puntos, debe mostrar 3 productos | |
|  | |

**EJERCICIO 2**

Construir el siguiente programa con las siguiente reglas de negocio:

* Inicio del Programa:
  + El programa comienza inicializando una variable sw en 1, indicando que el sistema está activo, puede utilizar también “while true:”
* Menú Principal:
  + Se presenta un menú al usuario con las siguientes opciones:
    - Ver mi Saldo
    - Retirar Dinero
    - Salir
* Selección de Opción:
  + Se le pide al usuario que seleccione una opción ingresando un número.
* Validación de Entrada:
  + Se verifica que la opción ingresada esté en el rango de 1 a 3.
* Acciones según la Opción:
* Si la opción es 1:
  1. Se imprime el mensaje "Tiene un saldo de $500000".
  2. Se solicita al usuario que presione 1 para volver atrás o 2 para salir.
  3. Si el usuario presiona 2, se imprime "Cierre de sesión exitoso, adiós" y se sale del sistema.
* Si la opción es 2:
  1. Se imprime "Transferencia realizada".
  2. Se solicita al usuario que presione 1 para volver atrás o 2 para salir.
  3. Si el usuario presiona 2, se imprime "Cierre de sesión exitoso, adiós" y se sale del sistema.
* Si la opción es 3:
  1. Se imprime "Cierre de sesión exitoso, adiós".
  2. Se cambia el valor de sw a 0, saliendo así del bucle principal y terminando el programa, también se puede utilizar el comando “break”
* Manejo de Errores:
  + Se utiliza un bloque try-except para manejar errores durante la ejecución, mostrando un mensaje en caso de ingreso incorrecto.
* Salir del Programa:
  + Cambiando el valor de sw a 0, se sale del bucle principal y termina el programa.
* Confirmación del Usuario:
  + Después de realizar ciertas acciones (ver saldo o transferir dinero), se solicita al usuario que confirme si desea volver atrás o salir.

**EJERCICIO**

Observa el siguiente código y programa las sentencias en Visual Studio Code**.**

#CargaTarjetaBip

sw = 1

saldo = 0

while sw==1:

print("1. Ver mi Saldo")

print("2. Cargar Saldo")

print("3. Salir")

op=int(input("Seleccione una opción: "))

try:

if(op > 0 and op < 4):

if op == 1:

print(f"Tiene un saldo de {saldo}")

continu = int(input("presione 1) para volver atrás, presione 2) para salir "))

if continu==2:

print("Cierre de sesión exitoso, adiós")

sw=0

if op == 2:

print("¿Cuánto desea cargar?")

print("1.- $1.000")

print("2.- $5.000")

continu = int(input("Seleccione la opción"))

if continu==1:

saldo = saldo+1000

print(f"Carga exitosa, su saldo es de: ${saldo}")

if continu==2:

saldo = saldo+5000

print(f"Carga exitosa, su saldo es de: ${saldo}")

if op == 3:

print("Muchas gracias por ocupar el servicio, adiós")

sw=0

else:

print("Selección fuera de rango")

except:

print("Ingreso Erróneo")

|  |  |
| --- | --- |
| Pregunta | Agrega al código anterior sentencias que permitan incorporar en el menú la opción de cargar $20.000 a la tarjeta, y luego se acumule al saldo cargado, al saldo actual |
|  |  |