

PRODUCTO INTEGRADOR DE APRENDIZAJE

Nombre de la Unidad de Aprendizaje : Estructuras de Datos y su Procesamiento

Nombre del proyecto: Producto Integrador de Aprendizaje

Programa educativo: Licenciado en Tecnologías de Información

Semestre: 3ro Grupo: 34

Nombre del docente: Alvaro Francisco Salazar González.




Nombre de los integrantes del equipo:

Chávez Juárez Alan Alexis	2064674
Garza López Omar de Jesús	2097385
Loera Alvarez Denis Franco	2056827

Contenido mínimo a evaluar	Cumplimiento
Índice	
Introducción .- incluye valores UANL aplicados	
Análisis y emisión de juicio	
Conclusiones individuales	
Conclusión del equipo	
Identificación de sub resultados de aprendizaje ANECA.	
Calificación Evidencia:	
Firma del maestro	

San Nicolás de los Garza, ciudad universitaria a (08-11-2024)

Integrantes del equipo

Nombre	Fotografía
Chávez Juárez Alan Alexis	 A portrait of a young man with dark hair, wearing a dark blue blazer over a dark blue patterned shirt. He is looking directly at the camera against a plain, light-colored background.
Garza López Omar de Jesús	 A portrait of a young man with dark hair, wearing a black t-shirt. He is looking directly at the camera against a background that appears to be a wooden headboard.
Loera Alvarez Denis Franco	 A portrait of a young man with dark hair, wearing a white collared shirt. He is looking directly at the camera against a plain, light-colored background.

Índice

Listado de requerimientos	1
Listado de requerimientos específicos.....	2
Requerimientos en formato APA 7	5
Pseudocódigo.....	68
Diagrama del modelo de datos	85
Listado completo del código de la aplicación.....	86
Reporte de plataforma tecnológica	123
Aportaciones específicas	126

LISTADO DE REQUERIMENTOS

- 1) Unidad
 - a) Clave (Numérico entero mayor a cero, irrepetible y generado automáticamente por el sistema).
 - b) Rodada (Numérico a elegir de los siguientes: 20, 26 o 29)
 - c) Color (Texto, obligatorio, aceptando hasta un máximo de 15 caracteres)
- 2) Cliente
 - a) Clave (Numérico entero mayor a cero, irrepetible y generado automáticamente por el sistema).
 - b) Apellidos (Texto, obligatorio, aceptando hasta un máximo de 40 caracteres).
 - c) Nombres (Texto, obligatorio, aceptando hasta un máximo de 40 caracteres).
 - d) Teléfono (Texto, obligatorio, de 10 caracteres pero que contenga solamente dígitos).
- 3) Préstamo
 - a) Folio (Numérico entero mayor a cero, irrepetible y generado automáticamente por el sistema).
 - b) Clave de la unidad (Numérico entero mayor a cero y obligatorio).
 - c) Clave del Cliente (Numérico entero mayor a cero y obligatorio).
 - d) Fecha del préstamo (Fecha, obligatorio y en formato mm-dd-aaaa).
 - e) Cantidad de días del préstamo (Numérico entero y obligatorio).
 - f) Fecha del retorno de la unidad (Fecha y en formato mm-dd-aaaa; no contendrá datos al momento del préstamo, se llenará al momento de que el cliente retorne la unidad).
- 4) Retorno de unidad:
 - a) El retorno de la unidad se identificará mediante el folio del préstamo (Se presentará un listado de los préstamos pendientes por retornar hasta ese momento ordenado de manera descendente por la fecha de préstamo para que el usuario pueda identificar el préstamo a retornar)
 - b) La fecha efectiva en que la unidad se retornó por parte del cliente no podrá ser anterior a la fecha de préstamo.

Es importante considerar las siguientes restricciones adicionales al momento de registrar un préstamo:

- a) La clave de la unidad debe corresponder a una unidad previamente registrada y que no se encuentre prestada al momento del registro.
- b) La clave del cliente debe corresponder a un cliente previamente registrado.
- c) La fecha del préstamo por default será la fecha del sistema, pero el usuario puede optar por registrar alguna fecha en el futuro no permitiendo el registro de fechas anteriores a la fecha actual.
- d) La cantidad de días a prestar la unidad debe ser por lo menos de un día y hasta un máximo de catorce días.

LISTADO DE REQUERIMIENTOS ESPECÍFICOS

1) El desarrollo debe implementar la estructura de menú presentada en la Figura 1.

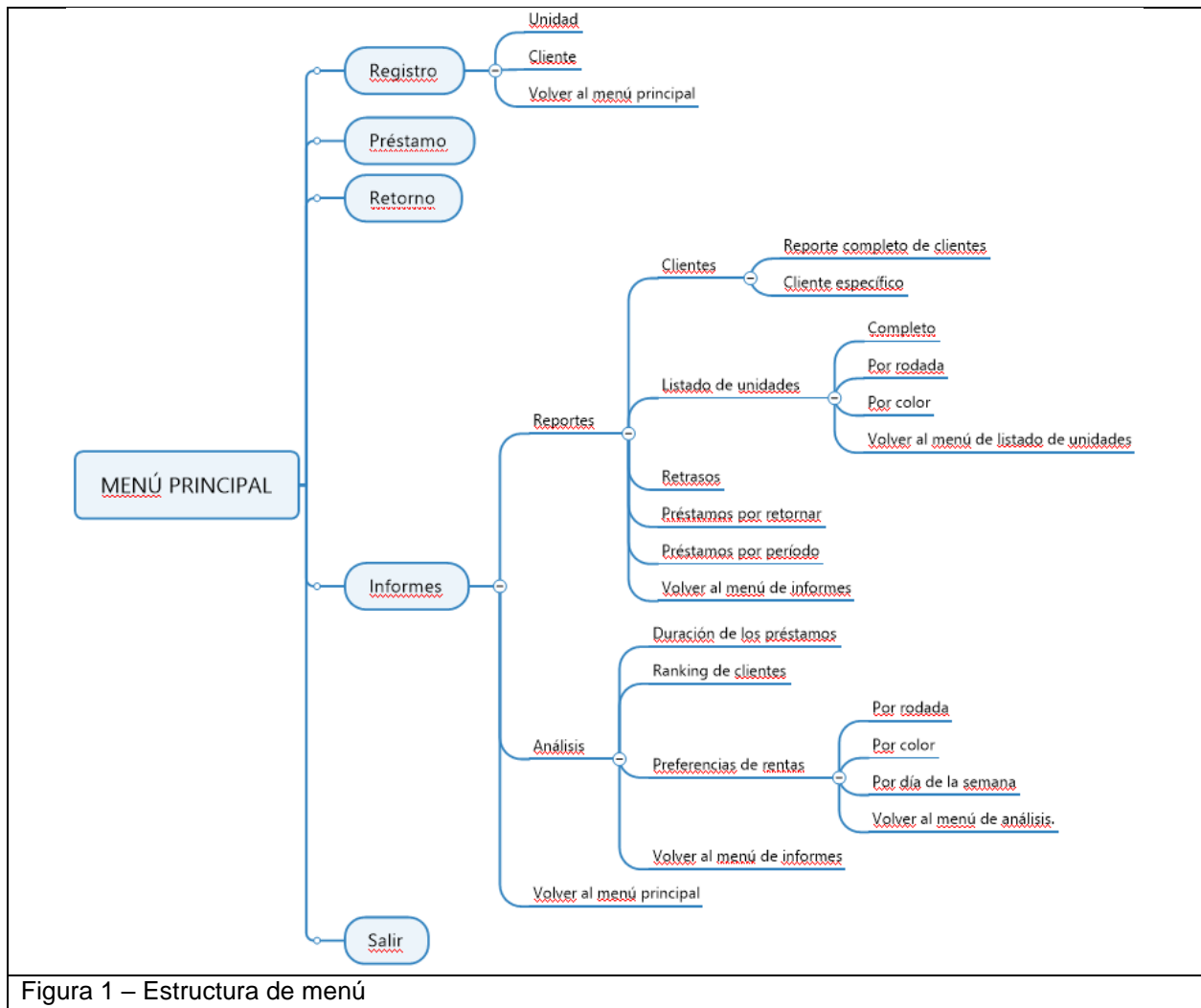


Figura 1 – Estructura de menú

2. La solución debe atender los siguientes requerimientos específicos además de los ya señalados en el escenario y contexto del caso de trabajo:
 - 2.1. Cada vez que se ingrese, o se salga de un menú se deberá presentar una representación de la ruta anidada correspondiente que acompañe a la solicitud de elección del usuario. Ejemplo: Si el usuario se encuentra en el menú en que puede obtener un reporte del listado de unidades se le debe desplegar la siguiente ruta *"Menú Principal > Informes > Reportes"*.
 - 2.2. Cada vez que se ejecute cualquier opción de los menús y submenús el sistema debe volver al nivel de menú en que se presentó la opción recién ejecutada.
 - 2.3. Si alguno de los datos proporcionados por el usuario no cumple con las condiciones especificadas en el contexto del caso, no se debe permitir avanzar hasta que dichas condiciones se satisfagan o bien ofrecerle alguna opción para cancelar el proceso.
 - 2.4. Cada vez que se le solicite al usuario una clave de unidad o cliente se le debe presentar un listado que resuma el actual catálogo de dicho dato.

2.5. Todos los reportes se deben presentar en pantalla en forma tabular y, posteriormente, ofrecerse al usuario la opción de exportar el reporte presentado en formato CSV o MSeExcel

2.6. Especificación de los reportes:

- 2.6.1. Reporte completo de clientes: Un listado de todos los clientes registrados hasta ese momento que incluya cada atributo de ellos.
- 2.6.2. Cliente específico: Se mostrarán los detalles completos de un cliente identificado por su clave y a continuación un reporte tabular con todo su historial de préstamos y devoluciones de unidades ordenado por fecha de préstamo de manera ascendente.
- 2.6.3. Listado de unidades completo: Un listado de todas las unidades registradas hasta ese momento que incluya cada atributo de ellas.
- 2.6.4. Listado de unidades por rodada: Un listado de todas las unidades registradas hasta ese momento que correspondan a la rodada que indique el usuario que presente la clave y color de cada unidad reportada.
- 2.6.5. Listado de unidades por color: Un listado de todas las unidades registradas hasta ese momento que correspondan al color que indique el usuario presentando la clave y rodada de cada unidad reportada
- 2.6.6. Retrasos: Un reporte de todos los préstamos realizados y, que, a la fecha actual, presenten un retraso en su entrega; ordenados de mayor a menor retraso, presentando los datos de: Días de retraso, fecha en que se debió haber retornado la unidad, clave de unidad, rodada, color, nombre completo del cliente y su teléfono de contacto.
- 2.6.7. Préstamos por retornar: Un reporte de todos los préstamos que estén pendientes de retornar y que se hayan prestado durante el período que indique el usuario. Debe presentar la clave de unidad, rodada, fecha del préstamo, nombre completo del cliente y el teléfono del cliente.
- 2.6.8. Préstamos por período: Un reporte de todos los préstamos realizados durante el período que indique el usuario. Debe presentar la clave de unidad, rodada, fecha del préstamo, nombre completo del cliente y el teléfono del cliente.

2.7. Especificación de los análisis:

- 2.7.1. Duración de los préstamos: Debe presentar un reporte de estadísticos descriptivos de la cantidad de días registrados al momento de prestar una unidad que considere todo registro de préstamo realizado; debe incluir: Media, mediana, moda, mínimo, máximo, desviación estándar y cuartiles.
- 2.7.2. Ranking de clientes: Reporte con los siguientes datos de los clientes: Cantidad de rentas acumuladas desde su registro, Clave, nombre completo y teléfono. Ordenado de manera descendente según su acumulado de préstamos a la fecha del reporte.
- 2.7.3. Preferencias de rentas: Submenú que ofrecerá las siguientes opciones:
 - 2.7.3.1. Por rodada: Cantidad acumulada de préstamos por rodada ordenado de manera descendente según su acumulado de préstamos a la fecha. Se debe presentar también una gráfica de pastel que presente la proporción dentro del total de préstamos por cada rodada ofertada.
 - 2.7.3.2. Por color: Cantidad de préstamos por color ordenado de manera descendente según su acumulado de préstamos a la

fecha. Se debe presentar también una gráfica de pastel que presente la proporción dentro del total de cada color ofertado.

2.7.3.3. Por día de la semana: Cantidad de préstamos totales acumulados hasta el momento por día de la semana, ordenado considerando que esta inicia los domingos. Se debe presentar también una gráfica de barras que presente la cantidad por cada día de la semana.

2.7.3.4. Volver al menú de análisis.

2.8. Todos los datos deben almacenarse en una base de datos de SQLite y no debe perderse dato alguno entre ejecuciones del sistema.

2.9. Antes de salir del sistema se debe solicitar la correspondiente confirmación al usuario.

REQUERIMIENTOS EN FORMATO APA 7

1 Los requerimientos de (unidad) son mostrados en la figura 1

-Clave (Numérico entero mayor a cero, irrepitable y generado automáticamente por el sistema).

-Rodada (Numérico a elegir de los siguientes: 20, 26 o 29).

-Color (Texto, obligatorio, aceptando hasta un máximo de 15 caracteres).

Figura 1.0

Ingreso de un número valido para registrar en la entidad unidad

```
Menú de procesos > Menú de registro
MENÚ DE REGISTRO -----
1- Unidad
2- Cliente
3- Regresar
Introduzca la clave de la entidad a registrar: 1

Menú de procesos > Menú de registro > Registro de Unidad
REGISTRO - UNIDAD -----
- Introduzca 'c' para cancelar el proceso -
Introduzca la rodada de la unidad (20, 26, 29): █
AlanBranch* 0 0 0
```

Figura 1.1

Intento de campo vacío en rodada de unidad

```
Menú de procesos > Menú de registro > Registro de Unidad
REGISTRO - UNIDAD -----
- Introduzca 'c' para cancelar el proceso -
Introduzca la rodada de la unidad (20, 26, 29):
Este campo es obligatorio. Por favor, introduzca la rodada de la unidad.

Introduzca la rodada de la unidad (20, 26, 29): █
AlanBranch* 0 0 0
```

Figura 1.2

Intento de ingresar una rodada no disponible "15"

```
Introduzca la rodada de la unidad (20, 26, 29): 15
La rodada 15 no es válida debido a que no es una de las rodadas registradas. Por favor, introduzca alguna de las rodadas válidas.

Introduzca la rodada de la unidad (20, 26, 29): █
AlanBranch* 0 0 0
```


Figura 1.3*Intento de ingresar una cadena de texto*

```

Introduzca la rodada de la unidad (20, 26, 29): veinte
La rodada veinte no es válida debido a que no es un valor numérico. Por favor, introduzca alguna de las rodadas válidas.

Introduzca la rodada de la unidad (20, 26, 29): █
AlanBranch* 0 0 0 0 0

```

Figura 1.4*Intento de valor valido "20"*

```

Introduzca la rodada de la unidad (20, 26, 29): 20
Introduzca el color de la unidad (máximo 15 caracteres): █

```

Nota: Se agregaron 2 unidades más y display de siguiente paso: introducir el color de la unidad a registrar

Figura 1.5*Intento de campo vacío*

```

Introduzca el color de la unidad (máximo 15 caracteres):
Este campo es obligatorio. Por favor, introduzca el color de la unidad.

Introduzca el color de la unidad (máximo 15 caracteres): █
AlanBranch* 0 0 0 0 0

```

Figura 1.6*Intento de ingreso de color con números*

```

Introduzca el color de la unidad (máximo 15 caracteres): 150
El color de la unidad no debe contener números ni caracteres especiales. Por favor, introduzca el color de la unidad únicamente con letras.

Introduzca el color de la unidad (máximo 15 caracteres): █
AlanBranch* 0 0 0 0 0

```

Figura 1.7*Intento de más de 15 caracteres*

```

Introduzca el color de la unidad (máximo 15 caracteres): masdequincecaracteres
El máximo de caracteres permitidos es de 15. Por favor, introduzca el color de la unidad con una longitud de caracteres válida.

Introduzca el color de la unidad (máximo 15 caracteres): █
AlanBranch* 0 0 0 0 0

```

Figura 1.8*Intento de caracteres especiales*

```

Introduzca el color de la unidad (máximo 15 caracteres): masdequincecaracteres
El máximo de caracteres permitidos es de 15. Por favor, introduzca el color de la unidad con una longitud de caracteres válida.

Introduzca el color de la unidad (máximo 15 caracteres): █
AlanBranch* 0 0 0 0

```

Figura 1.9*Inserción de datos correctos "Rojo" y regreso a menú de registro*

```

Introduzca el color de la unidad (máximo 15 caracteres): Rojo

REGISTRO EXITOSO - Se ha registrado una unidad con los datos:

=====
CLAVE    RODADA    COLOR
=====
1        20      Rojo
=====

Menú de procesos > Menú de registro
MENÚ DE REGISTRO -----
1- Unidad
2- Cliente
3- Regresar
Introduzca la clave de la entidad a registrar: █

```

2 Los requerimientos de (clientes) se muestran en la figura 2.

- Clave (Numérico entero mayor a cero, irrepetible y generado automáticamente por el sistema).
- Apellidos (Texto, obligatorio, aceptando hasta un máximo de 40 caracteres).
- Nombres (Texto, obligatorio, aceptando hasta un máximo de 40 caracteres).
- Teléfono (Texto, obligatorio, de 10 caracteres pero que contenga solamente dígitos).

Figura 2.0

Inserción de dato para entidad cliente "2" y siguiente paso: introducir apellidos de cliente

```
Menú de procesos > Menú de registro
MENÚ DE REGISTRO -----
1- Unidad
2- Cliente
3- Regresar
Introduzca la clave de la entidad a registrar: 2

Menú de procesos > Menú de registro > Registro de Cliente
REGISTRO - CLIENTE -----
- Introduzca 'c' para cancelar el proceso -
Introduzca los apellidos del cliente (máximo 40 caracteres): █
AlanBranch*  ↩  ⊗ 0 △ 0  🗑 0
```

Figura 2.1

Intento de campo vacío

```
Menú de procesos > Menú de registro > Registro de Cliente
REGISTRO - CLIENTE -----
- Introduzca 'c' para cancelar el proceso -
Introduzca los apellidos del cliente (máximo 40 caracteres):
Este campo es obligatorio. Por favor, introduzca los apellidos del cliente.

Introduzca los apellidos del cliente (máximo 40 caracteres): █
AlanBranch*  ↩  ⊗ 0 △ 0  🗑 0
```

Figura 2.2

Intento de inserción datos inválidos "123" y excepción

```
Introduzca los apellidos del cliente (máximo 40 caracteres): 123
Los apellidos no deben contener números ni caracteres especiales. Por favor, introduzca los apellidos del cliente únicamente con letras.

Introduzca los apellidos del cliente (máximo 40 caracteres): █
AlanBranch*  ↩  ⊗ 0 △ 0  🗑 0
```

Figura 2.3*Intento de inserción más de 40 caracteres*

```

Introduzca los apellidos del cliente (máximo 40 caracteres): más de cuarenta caracteres en el apellido es mucho
El máximo de caracteres permitidos es de 40. Por favor, introduzca los apellidos del cliente con una longitud de caracteres válida.

Introduzca los apellidos del cliente (máximo 40 caracteres): 
AlanBranch* 0 0 0 0

```

Figura 2.4*Inserción de datos correctos y siguiente paso: nombres del cliente*

```

Introduzca los apellidos del cliente (máximo 40 caracteres): Loera alvarez
Introduzca los nombres del cliente (máximo 40 caracteres): 
AlanBranch* 0 0 0 0

```

Figura 2.5*Intento campo vacío*

```

Introduzca los apellidos del cliente (máximo 40 caracteres): Loera alvarez
Introduzca los nombres del cliente (máximo 40 caracteres): 
Este campo es obligatorio. Por favor, introduzca los nombres del cliente.

Introduzca los nombres del cliente (máximo 40 caracteres): 
AlanBranch* 0 0 0 0

```

Figura 2.6*Intento de inserción más de 40 caracteres*

```

Introduzca los nombres del cliente (máximo 40 caracteres): mas de cuarenta caracteres en el nombre es mucho
El máximo de caracteres permitidos es de 40. Por favor, introduzca los nombres del cliente con una longitud de caracteres válida.

Introduzca los nombres del cliente (máximo 40 caracteres): 
AlanBranch* 0 0 0 0

```

Figura 2.7*Intento inserción caracteres especiales*

```

Introduzca los nombres del cliente (máximo 40 caracteres): !"#%&'()*=?;
Los nombres no deben contener números ni caracteres especiales. Por favor, introduzca los nombres del cliente únicamente con letras.

Introduzca los nombres del cliente (máximo 40 caracteres): 
AlanBranch* 0 0 0 0

```

Figura 2.8*Intento de inserción de números*

```

Introduzca los nombres del cliente (máximo 40 caracteres): 2+2
Los nombres no deben contener números ni caracteres especiales. Por favor, introduzca los nombres del cliente únicamente con letras.

Introduzca los nombres del cliente (máximo 40 caracteres): 
AlanBranch*  ↩  ⊗ 0 △ 0  🚫 0

```

Figura 2.9*Inserción de datos correctos "Denise Franco" y siguiente paso: Teléfono*

```

Introduzca los nombres del cliente (máximo 40 caracteres): Denise Franco
Introduzca el teléfono del cliente (10 caracteres):

```

Figura 2.10*Inserción de datos correcta y siguiente paso: teléfono del cliente*

```

Introduzca los nombres del cliente (máximo 40 caracteres): Denise Franco
Introduzca el teléfono del cliente (10 caracteres): 
AlanBranch*  ↩  ⊗ 0 △ 0  🚫 0

```

Figura 2.11*Intento de campo vacío*

```

Introduzca el teléfono del cliente (10 caracteres):
Este campo es obligatorio. Por favor, introduzca el teléfono del cliente.

Introduzca el teléfono del cliente (10 caracteres): 
AlanBranch*  ↩  ⊗ 0 △ 0  🚫 0

```

Figura 2.12*Intento inserción de cadena de texto*

```

Introduzca el teléfono del cliente (10 caracteres): Telefono
El teléfono no debe contener letras ni caracteres especiales. Por favor, introduzca el teléfono del cliente únicamente con números.

Introduzca el teléfono del cliente (10 caracteres): 
AlanBranch*  ↩  ⊗ 0 △ 0  🚫 0

```

Figura 2.13*Intento inserción de un solo número*

```

Introduzca el teléfono del cliente (10 caracteres): 1
La longitud de caracteres del teléfono es diferente a 10. Por favor, introduzca el teléfono del cliente con una longitud de caracteres válida.

Introduzca el teléfono del cliente (10 caracteres): 
AlanBranch*  ↩  ⊗ 0 △ 0  🚫 0

```

Figura 2.14*Intento inserción más de 10 dígitos*

```

Introduzca el teléfono del cliente (10 caracteres): 123456789101112
La longitud de caracteres del teléfono es diferente a 10. Por favor, introduzca el teléfono del cliente con una longitud de caracteres válida.

Introduzca el teléfono del cliente (10 caracteres): █
AlanBranch* 0 0 0 0

```

Figura 2.15*Inserción de datos correcta y display de lista de clientes y regreso al menú de registro*

```

Introduzca el teléfono del cliente (10 caracteres): 0123456789

REGISTRO EXITOSO - Se ha registrado un cliente con los datos:

=====
NOMBRE COMPLETO                                TELÉFONO
=====
Denise Franco Loera Alvarez                     0123456789
=====

Menú de procesos > Menú de registro
MENÚ DE REGISTRO -----
1- Unidad
2- Cliente
3- Regresar
Introduzca la clave de la entidad a registrar: █
AlanBranch* 0 0 0 0

```

3 Ahora los siguientes requerimientos que son de (prestamos) se muestran en la figura 3.

- Folio (Numérico entero mayor a cero, irrepetible y generado automáticamente por el sistema).
- Clave de la unidad (Numérico entero mayor a cero y obligatorio).
- Clave del Cliente (Numérico entero mayor a cero y obligatorio).
- Fecha del préstamo (Fecha, obligatorio y en formato mm-dd-aaaa).
- Cantidad de días del préstamo (Numérico entero y obligatorio).
- Fecha del retorno de la unidad (Fecha y en formato mm-dd-aaaa; no contendrá datos al momento del préstamo, se llenará al momento de que el cliente retorne la unidad).

Figura 3.0

Selección de préstamo y display de menú de préstamo

```

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 2

Menú de procesos > Menú de préstamo
MENÚ DE PRÉSTAMO -----
- Introduzca 'c' para cancelar el proceso -

      UNIDADES DISPONIBLES
=====
UNIDAD  RODADA  COLOR
=====
   1      20     Rojo
   2      26     Azul
   3      29     Amarillo
=====

Introduzca la clave de la unidad: 

```

Figura 3.1

Insertión de dato invalido "4"

```

Introduzca la clave de la unidad: 4
Esta unidad no se encuentra registrada. Por favor, introduzca la clave de alguna unidad ya registrada.

Introduzca la clave de la unidad: 

```

Figura 3.2*Intento de Campo vacío*

```

Introduzca la clave de la unidad:
Este campo es obligatorio. Por favor, introduzca la clave de la unidad.

Introduzca la clave de la unidad: 

```

Figura 3.3*Intento de Inserción de cadena de texto*

```

Introduzca la clave de la unidad: Uno
La clave de la unidad no puede contener letras ni caracteres especiales. Por favor, introduzca la clave de la unidad únicamente con números.

Introduzca la clave de la unidad: 

```

Figura 3.4*Intento de Inserción de numero negativo*

```

Introduzca la clave de la unidad: Uno
La clave de la unidad no puede contener letras ni caracteres especiales. Por favor, introduzca la clave de la unidad únicamente con números.

Introduzca la clave de la unidad: 

```

Figura 3.5*Inserción de dato valido "1" y display de clientes posibles para autorizar préstamo*

```

Introduzca la clave de la unidad: 1

                                CLIENTES
=====
CLAVE      NOMBRE COMPLETO
=====
1          Denise Franco Loera Alvarez
2          Alan Alexis Chavez Juarez
3          Omar De Jesus Garza Lopez
=====

Introduzca la clave del cliente: 

```

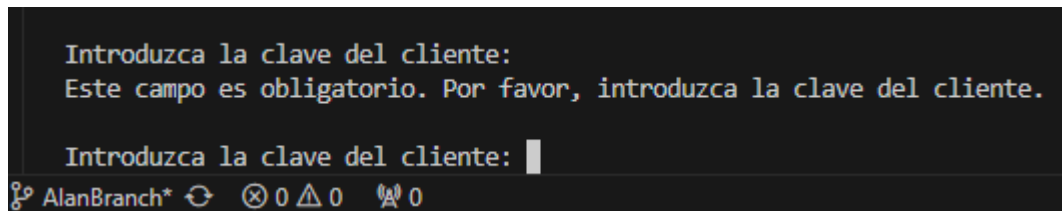
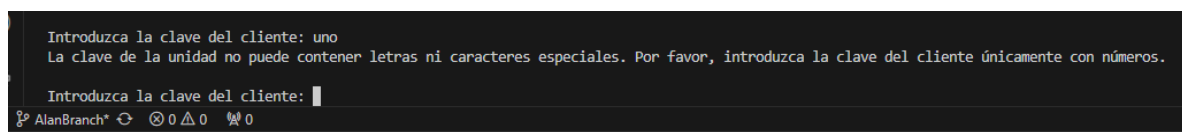
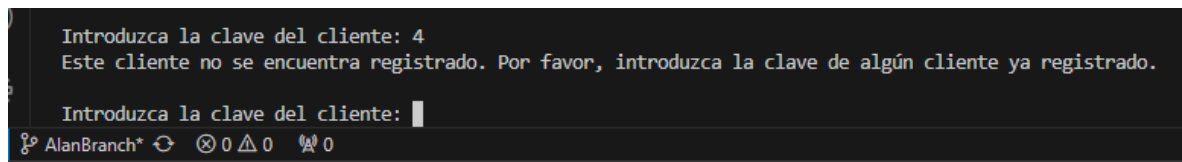
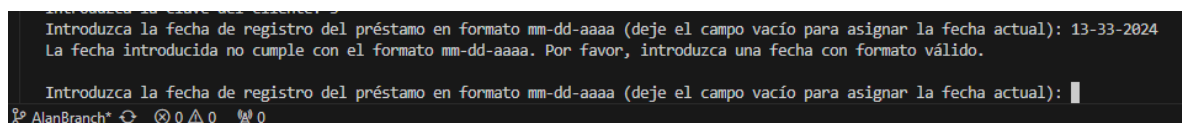
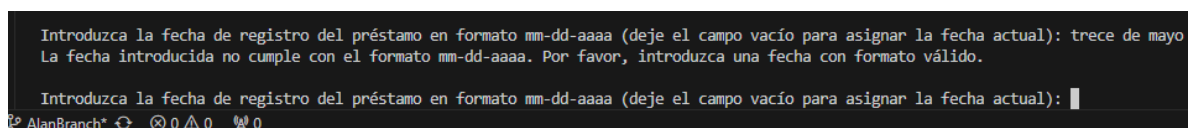

Figura 3.6*Intento de campo vacío***Figura 3.7***Intento de inserción de cadena de texto***Figura 3.8***Intento de inserción de clave no disponible***Figura 3.9***Inserción de dato válido "3" y siguiente paso: fecha de registro de préstamo***Figura 3.10***Inserción de formato de fecha inválida***Figura 3.11***Intento inserción cadena de texto*

Figura 3.12*Dejar campo vacío para tomar fecha actual*

```

Introduzca la fecha de registro del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual):
Introduzca la cantidad de días del préstamo: 
AlanBranch* 0 0 0 0

```

Figura 3.13*Intento inserción cadena de texto*

```

Introduzca la fecha de registro del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual):
Introduzca la cantidad de días del préstamo: doce
La cantidad de días del préstamo no puede contener letras ni caracteres especiales. Por favor, introduzca la cantidad de días del préstamo únicamente con números.
Introduzca la cantidad de días del préstamo: 
AlanBranch* 0 0 0 0

```

Figura 3.14*Intento Campo Vacío*

```

Introduzca la cantidad de días del préstamo:
Este campo es obligatorio. Por favor, introduzca la cantidad de días del préstamo.
Introduzca la cantidad de días del préstamo: 
AlanBranch* 0 0 0 0

```

Figura 3.15*Intento inserción más de 14 días "123"*

```

Introduzca la cantidad de días del préstamo: 123
La cantidad mínima de días del préstamo debe ser de 1 día y máximo 14 días. Por favor, introduzca una cantidad de días del préstamo que se encuentre dentro de ese rango.
Introduzca la cantidad de días del préstamo: 
AlanBranch* 0 0 0 0

```

Figura 3.16*Inserción de dato válido "4" y display de lista de préstamos autorizados y regresa al menú de Procesos*

```

Introduzca la cantidad de días del préstamo: 4

REGISTRO EXITOSO - Se ha registrado un préstamo con los datos:

=====
FOLIO      CLAVE DE UNIDAD  CLAVE DE CLIENTE  FECHA DEL PRÉSTAMO  CANTIDAD DE DÍAS DEL PRÉSTAMO
=====
1          1                3                 10-11-2024         4
=====

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 
AlanBranch* 0 0 0 0

```

4 Los requerimientos que continúan son los de (retorno de unidad) y se muestran en la figura 4.

-El retorno de la unidad se identificará mediante el folio del préstamo (Se presentará un listado de los préstamos pendientes por retornar hasta ese momento ordenado de manera descendente por la fecha de préstamo para que el usuario pueda identificar el préstamo a retornar)

-La fecha efectiva en que la unidad se retornó por parte del cliente no podrá ser anterior a la fecha de préstamo.

Figura 4.0

Inserción de valor para entidad Menú de retorno y display de unidades a Retornar pendientes

```

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 3

Menú de procesos > Menú de retorno
MENÚ DE RETORNO -----
- Introduzca 'c' para cancelar el proceso -

=====
PRÉSTAMOS POR RETORNAR
=====
FECHA DEL PRÉSTAMO    FOLIO    CLAVE DE UNIDAD    CLAVE DE CLIENTE    CANTIDAD DE DÍAS DEL PRÉSTAMO
=====
10-11-2024            1        1                1                    4
10-11-2024            2        2                2                    5
=====

Introduzca el folio del préstamo a retornar: █

```

Figura 4.1

Intento Campo vacío

```

Introduzca el folio del préstamo a retornar:
Este campo es obligatorio. Por favor, introduzca el folio del préstamo.

Introduzca el folio del préstamo a retornar: █

```

Figura 4.2

Intento inserción de cadena de texto

```

Introduzca el folio del préstamo a retornar: uno
El folio no puede contener letras ni caracteres especiales. Por favor, introduzca el folio del préstamo únicamente con números.

Introduzca el folio del préstamo a retornar: █

```

Figura 4.3

Intento inserción valores negativos

```
Introduzca el folio del préstamo a retornar: -15
El folio no puede contener letras ni caracteres especiales. Por favor, introduzca el folio del préstamo únicamente con números.
Introduzca el folio del préstamo a retornar: █
```

Figura 4.4

Intento inserción valor no registrado

```
Introduzca el folio del préstamo a retornar: 3
Este folio no se encuentra registrado. Por favor, introduzca algún folio de préstamo ya registrado.
Introduzca el folio del préstamo a retornar: █
```

Figura 4.5

Inserción valor valido "2" y siguiente paso: Introducir fecha del retorno

```
Introduzca el folio del préstamo a retornar: 2
Introduzca la fecha del retorno del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual): █
```

Figura 4.6

Inserción de fecha anterior a fecha de préstamo

```
Introduzca la fecha del retorno del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual): 09-11-2024
La fecha del retorno no puede ser inferior a la fecha del préstamo. Por favor, introduzca una fecha válida o cancele el proceso con 'c'.
Introduzca la fecha del retorno del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual): █
```

Figura 4.7

Fecha valida y Retorno exitoso

```
Introduzca el folio del préstamo a retornar: 2
Introduzca la fecha del retorno del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual): 10-15-2024

REGISTRO EXITOSO - Se ha registrado el retorno
```

5 Es importante considerar las siguientes restricciones adicionales al momento de registrar un préstamo, y estas se muestran en la figura 5.

- La clave de la unidad debe corresponder a una unidad previamente registrada y que no se encuentre prestada al momento del registro.
- La clave del cliente debe corresponder a un cliente previamente registrado.
- La fecha del préstamo por default será la fecha del sistema, pero el usuario puede optar por registrar alguna fecha en el futuro no permitiendo el registro de fechas anteriores a la fecha actual.
- La cantidad de días a prestar la unidad debe ser por lo menos de un día y hasta un máximo de catorce días.

Figura 5.0

Menú donde no muestra la unidad que ya está con un préstamo activo

```

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 2

Menú de procesos > Menú de préstamo
MENÚ DE PRÉSTAMO -----
- Introduzca 'c' para cancelar el proceso -

      UNIDADES DISPONIBLES
=====
UNIDAD  RODADA    COLOR
=====
    2      26      Azul
    3      29      Amarillo
    4      26      Verde
=====

Introduzca la clave de la unidad: █

```

Figura 5.1

Inserción de clave de unidad prestada anteriormente

```

Introduzca la clave de la unidad: 1
Actualmente esta unidad no se encuentra disponible. Por favor, introduzca la clave de alguna otra unidad.

Introduzca la clave de la unidad: █

```

Figura 5.2*Display de opciones de clientes disponibles para prestamos*

```

Introduzca la clave de la unidad: 2

                                CLIENTES
=====
CLAVE      NOMBRE COMPLETO
=====
1          Denise Franco Loera Alvarez
2          Alan Alexis Chavez Juarez
3          Omar De Jesus Garza Lopez
=====

Introduzca la clave del cliente: █

```

Figura 5.3*Selección de cliente valido previamente mostrado*

```

Introduzca la clave del cliente: 3
Introduzca la fecha de registro del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual): █

```

Figura 5.4*Intento de inserción de fecha anterior a la actual*

```

Introduzca la fecha de registro del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual): 09-11-2024
No se puede registrar una fecha pasada. Por favor, introduzca una fecha válida.

Introduzca la fecha de registro del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual): █

```

Figura 5.5*Campo vacío para que tome por default la fecha actual*

```

Introduzca la fecha de registro del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual):
Introduzca la cantidad de días del préstamo: █

```

Figura 5.6*Inserción de fecha en futuro y siguiente paso: cantidad de días del préstamo*

```

Introduzca la fecha de registro del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual): 10-12-2024
Introduzca la cantidad de días del préstamo: █

```

Figura 5.7*Dejar el Campo Vacío*

```

Introduzca la cantidad de días del préstamo:
Este campo es obligatorio. Por favor, introduzca la cantidad de días del préstamo.

Introduzca la cantidad de días del préstamo: █

```

Figura 5.8*Inserción de dato invalido "15" en días del préstamo*

```

Introduzca la cantidad de días del préstamo: 15
La cantidad mínima de días del préstamo debe ser de 1 día y máximo 14 días. Por favor, introduzca una cantidad de días del préstamo que se encuentre dentro de ese rango.

Introduzca la cantidad de días del préstamo: █

```

Figura 5.9*Unidad de préstamo valido y datos tabulares de la descripción del préstamo*

```

Introduzca la cantidad de días del préstamo: 1

REGISTRO EXITOSO - Se ha registrado un préstamo con los datos:

=====
FOLIO      CLAVE DE UNIDAD  CLAVE DE CLIENTE  FECHA DEL PRÉSTAMO  CANTIDAD DE DÍAS DEL PRÉSTAMO
=====
4          2                3                10-12-2024         1
=====

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: █

```

6 Se expone que el flujo del programa sigue el flujo indicado en la estructura de menú presentada en la figura 1.

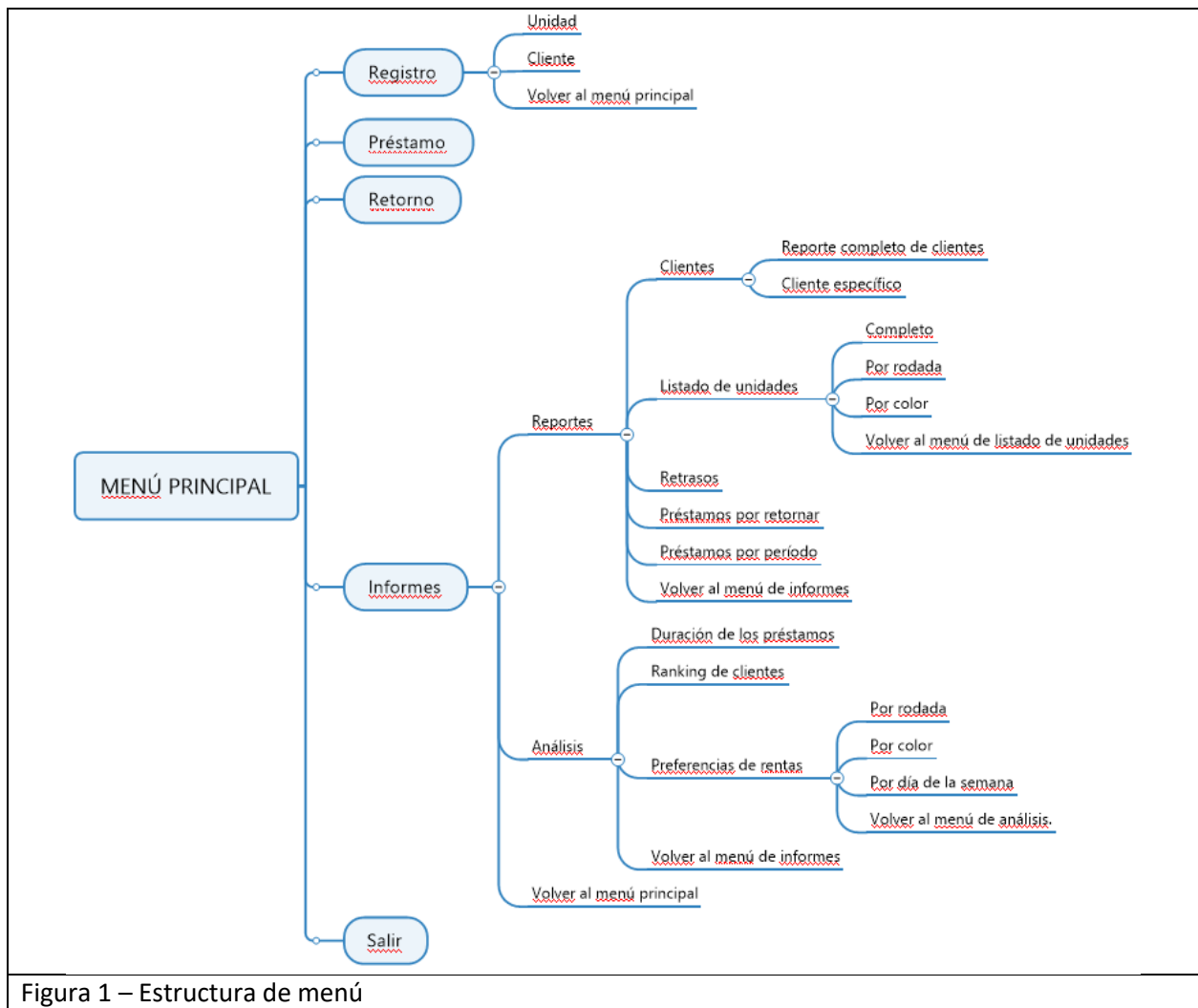


Figura 1 – Estructura de menú

Figura 6.1

Menú de procesos

```

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 
  
```


Figura 6.2

Inserción de cadena de texto en campo

```
La clave Registro no es válida debido a que no es un valor numérico. Por favor, introduzca alguna de las claves de procesos válidas.

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: █
```

Figura 6.3

Intento Inserción clave no valida

```
La clave 6 no es válida debido a que se encuentra fuera del rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: █
```

Figura 6.4

Inserción de clave 1 Registro y display de Menú de registro

```
MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 1

Menú de procesos > Menú de registro
MENÚ DE REGISTRO -----
1- Unidad
2- Cliente
3- Regresar
Introduzca la clave de la entidad a registrar: █
```

Figura 6.5

Inserción de clave 3 para regresar al menú de procesos

```

Menú de procesos > Menú de registro
MENÚ DE REGISTRO -----
1- Unidad
2- Cliente
3- Regresar
Introduzca la clave de la entidad a registrar: 3

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar:

```

Figura 6.6

Inserción de clave 2 para entrar a Menú de préstamo

```

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 2

Menú de procesos > Menú de préstamo
MENÚ DE PRÉSTAMO -----
- Introduzca 'c' para cancelar el proceso -

      UNIDADES DISPONIBLES
=====
UNIDAD  RODADA  COLOR
=====
    3      29    Amarillo
    4      26     Verde
=====

Introduzca la clave de la unidad:

```

Figura 6.7

Inserción de letra "C" para salir de menú de préstamo a menú de procesos

```

Menú de procesos > Menú de préstamo
MENÚ DE PRÉSTAMO -----
- Introduzca 'c' para cancelar el proceso -

      UNIDADES DISPONIBLES
=====
UNIDAD  RODADA  COLOR
=====
      3      29   Amarillo
      4      26    Verde
=====

Introduzca la clave de la unidad: C

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: █

```

Figura 6.8

Inserción de clave 3 para entrar a menú de retorno

```

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 3

Menú de procesos > Menú de retorno
MENÚ DE RETORNO -----
- Introduzca 'c' para cancelar el proceso -

      PRÉSTAMOS POR RETORNAR
=====
FECHA DEL PRÉSTAMO  FOLIO  CLAVE DE UNIDAD  CLAVE DE CLIENTE  CANTIDAD DE DÍAS DEL PRÉSTAMO
=====
      10-12-2024      4      2      3      1
      10-11-2024      3      1      3      5
=====

Introduzca el folio del préstamo a retornar: █

```

Figura 6.9

Inserción de letra "C" para salir de menú de retorno a menú de procesos

```
Menú de procesos > Menú de retorno
MENÚ DE RETORNO -----
- Introduzca 'c' para cancelar el proceso -

=====
PRÉSTAMOS POR RETORNAR
=====
FECHA DEL PRÉSTAMO      FOLIO      CLAVE DE UNIDAD      CLAVE DE CLIENTE      CANTIDAD DE DÍAS DEL PRÉSTAMO
=====
10-12-2024              4          2                    3                      1
10-11-2024              3          1                    3                      5
=====

Introduzca el folio del préstamo a retornar: C

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: █
```

Figura 6.10

Inserción de clave 4 para entrar a menú de informes

```
MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 4

Menú de procesos > Menú de informes
MENÚ DE INFORMES -----
1- Reportes.
2- Análisis.
3- Regresar.
Introduzca la clave la operación a realizar: █
```

Figura 6.11

Inserción de clave 1 para entrar a menú de reportes

```
Menú de procesos > Menú de informes
MENÚ DE INFORMES -----
1- Reportes.
2- Análisis.
3- Regresar.
Introduzca la clave la operación a realizar: 1

Menú de procesos > Menú de reportes
MENÚ DE REPORTES -----
1- Clientes.
2- Listado de Unidades.
3- Retrasos.
4- Préstamos por retornar.
5- Préstamos por periodo.
6- Regresar.
Introduzca la clave la operación a realizar: █
```

Figura 6.12

Inserción de clave 2 para entrar a Listado de unidades

```
Menú de procesos > Menú de reportes
MENÚ DE REPORTES -----
1- Clientes.
2- Listado de Unidades.
3- Retrasos.
4- Préstamos por retornar.
5- Préstamos por periodo.
6- Regresar.
Introduzca la clave la operación a realizar: 2

Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades
LISTADO DE UNIDADES -----
1- Completo.
2- Por rodada.
3- Por color.
4- Regresar.
Introduzca la clave del proceso a realizar: █
```

Figura 6.13

Insertión de clave 4 en listado de unidades para regresar a menú de reportes

```
Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades
LISTADO DE UNIDADES -----
1- Completo.
2- Por rodada.
3- Por color.
4- Regresar.
Introduzca la clave del proceso a realizar: 4

Menú de procesos > Menú de reportes
MENÚ DE REPORTES -----
1- Clientes.
2- Listado de Unidades.
3- Retrasos.
4- Préstamos por retornar.
5- Préstamos por periodo.
6- Regresar.
Introduzca la clave la operación a realizar: █
```

Figuras 6.14

Insertión de clave 6 para regresar de menú de reportes a menú de informes

```
Menú de procesos > Menú de reportes
MENÚ DE REPORTES -----
1- Clientes.
2- Listado de Unidades.
3- Retrasos.
4- Préstamos por retornar.
5- Préstamos por periodo.
6- Regresar.
Introduzca la clave la operación a realizar: 6

Menú de procesos > Menú de informes
MENÚ DE INFORMES -----
1- Reportes.
2- Análisis.
3- Regresar.
Introduzca la clave la operación a realizar: █
```

Figura 6.15

Inserción de clave 2 para entrar a menú de análisis

```
Menú de procesos > Menú de informes
MENÚ DE INFORMES -----
1- Reportes.
2- Análisis.
3- Regresar.
Introduzca la clave la operación a realizar: 2

Menú de procesos > Menú de informes > Menú de análisis
MENÚ DE ANÁLISIS -----
1- Duración de los préstamos.
2- Ranking de clientes.
3- Preferencia de rentas.
4- Regresar.
Introduzca la clave la operación a realizar: █
```

Figura 6.16

Inserción de clave 3 para entrar a menú de preferencias de rentas

```
Menú de procesos > Menú de informes > Menú de análisis > Menú de preferencias de rentas
MENÚ DE PREFERENCIAS DE RENTAS -----
1- Por rodada.
2- Por color.
3- Por día de la semana.
4- Regresar.
Introduzca la clave la operación a realizar: █

master* 0 54 0
```

Figura 6.17

Inserción de clave 4 para regresar a menú de análisis

```
Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades
LISTADO DE UNIDADES -----
1- Completo.
2- Por rodada.
3- Por color.
4- Regresar.
Introduzca la clave del proceso a realizar: 4

Menú de procesos > Menú de reportes
MENÚ DE REPORTES -----
1- Clientes.
2- Listado de Unidades.
3- Retrasos.
4- Préstamos por retornar.
5- Préstamos por periodo.
6- Regresar.
Introduzca la clave la operación a realizar: █
```

Figura 6.18

Inserción de clave 6 para regresar de menú de reportes a Menú de informes

```
Menú de procesos > Menú de reportes
MENÚ DE REPORTES -----
1- Clientes.
2- Listado de Unidades.
3- Retrasos.
4- Préstamos por retornar.
5- Préstamos por periodo.
6- Regresar.
Introduzca la clave la operación a realizar: 6

Menú de procesos > Menú de informes
MENÚ DE INFORMES -----
1- Reportes.
2- Análisis.
3- Regresar.
Introduzca la clave la operación a realizar: █
```


Figura 6.19

Insertión de clave 3 para regresar de menú de informes a menú de procesos

```
Menú de procesos > Menú de informes
MENÚ DE INFORMES -----
1- Reportes.
2- Análisis.
3- Regresar.
Introduzca la clave la operación a realizar: 3

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: █
```

Figura 6.20

Insertión de clave 5 para salir del sistema

```
MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 5

Presione Enter para salir o introduzca cualquier otro caracter para continuar: █
```

Figura 6.21

Dejar campo vacío para salir del sistema

```
Presione Enter para salir o introduzca cualquier otro caracter para continuar:

Saliendo...
PS C:\Evidencia2> █
```

La solución debe atender los siguientes requerimientos específicos además de los ya señalados en el escenario y contexto del caso de trabajo:

7 Cada vez que se ingrese, o se salga de un menú se deberá presentar una representación de la ruta anidada correspondiente que acompañe a la solicitud de elección del usuario. Ejemplo: Si el usuario se encuentra en el menú en que puede obtener un reporte del listado de unidades se le debe desplegar la siguiente ruta “*Menú Principal > Informes > Reportes*”. Esto se ve reflejado en la figura 7

Figura 7.0

Inserción continua

```

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 4

Menú de procesos > Menú de informes
MENÚ DE INFORMES -----
1- Reportes.
2- Análisis.
3- Regresar.
Introduzca la clave la operación a realizar: 1

Menú de procesos > Menú de informes > Menú de reportes
MENÚ DE REPORTES -----
1- Clientes.
2- Préstamos por retornar.
3- Préstamos por periodo.
4- Regresar.
Introduzca la clave la operación a realizar: █

```

Nota: Para llegar a la opción menú de procesos> menú de informes> menú de reportes

8 Cada vez que se ejecute cualquier opción de los menús y submenús el sistema debe volver al nivel de menú en que se presentó la opción recién ejecutada. Es se representa en la figura 8.

Figura 8.0

Inserción de datos correctos "Rojo" y regreso a menú de registro

```

Introduzca el color de la unidad (máximo 15 caracteres): Rojo

REGISTRO EXITOSO - Se ha registrado una unidad con los datos:

=====
CLAVE      RODADA      COLOR
=====
1          20         Rojo
=====

Menú de procesos > Menú de registro
MENÚ DE REGISTRO -----
1- Unidad
2- Cliente
3- Regresar
Introduzca la clave de la entidad a registrar: █

```

Figura 8.1

Inserción de datos correcta y display de lista de clientes y regreso al menú de registro

```

Introduzca el teléfono del cliente (10 caracteres): 0123456789

REGISTRO EXITOSO - Se ha registrado un cliente con los datos:

=====
NOMBRE COMPLETO                                TELÉFONO
=====
Denise Franco Loera Alvarez                     0123456789
=====

Menú de procesos > Menú de registro
MENÚ DE REGISTRO -----
1- Unidad
2- Cliente
3- Regresar
Introduzca la clave de la entidad a registrar: █

```

Figura 8.2

Inserción de dato valido "4" y display de lista de prestamos autorizados y regresa al menú de Procesos.

```

Introduzca la cantidad de días del préstamo: 4

REGISTRO EXITOSO - Se ha registrado un préstamo con los datos:

=====
FOLIO      CLAVE DE UNIDAD  CLAVE DE CLIENTE  FECHA DEL PRÉSTAMO  CANTIDAD DE DÍAS DEL PRÉSTAMO
=====
1          1          3          10-11-2024          4
=====

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: █
AlanBranch*  ↺  ⊗ 0 △ 0  🗑 0

```

Figura 8.3

Unidad de préstamo valido y datos tabulares de la descripción del préstamo

```

Introduzca la cantidad de días del préstamo: 1

REGISTRO EXITOSO - Se ha registrado un préstamo con los datos:

=====
FOLIO      CLAVE DE UNIDAD  CLAVE DE CLIENTE  FECHA DEL PRÉSTAMO  CANTIDAD DE DÍAS DEL PRÉSTAMO
=====
4          2          3          10-12-2024          1
=====

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: █

```

9 Si alguno de los datos proporcionados por el usuario no cumple con las condiciones especificadas en el contexto del caso, no se debe permitir avanzar hasta que dichas condiciones se satisfagan o bien ofrecerle alguna opción para cancelar el proceso, esto se muestra en la figura 9.

Figura 9.0

Intento de campo vacío en rodada de unidad

```
Menú de procesos > Menú de registro > Registro de Unidad
REGISTRO - UNIDAD -----
- Introduzca 'c' para cancelar el proceso -
Introduzca la rodada de la unidad (20, 26, 29):
Este campo es obligatorio. Por favor, introduzca la rodada de la unidad.

Introduzca la rodada de la unidad (20, 26, 29):
```

Figura 9.1

Intento de ingresar una rodada no disponible "15"

```
Introduzca la rodada de la unidad (20, 26, 29): 15
La rodada 15 no es válida debido a que no es una de las rodadas registradas. Por favor, introduzca alguna de las rodadas válidas.

Introduzca la rodada de la unidad (20, 26, 29):
```

Figura 9.2

Intento de ingresar una cadena de texto

```
Introduzca la rodada de la unidad (20, 26, 29): veinte
La rodada veinte no es válida debido a que no es un valor numérico. Por favor, introduzca alguna de las rodadas válidas.

Introduzca la rodada de la unidad (20, 26, 29):
```

Figura 9.3

Intento de campo vacío

```
Introduzca el color de la unidad (máximo 15 caracteres):
Este campo es obligatorio. Por favor, introduzca el color de la unidad.

Introduzca el color de la unidad (máximo 15 caracteres):
```

Figura 9.4*Intento de ingreso de color con números*

```

Introduzca el color de la unidad (máximo 15 caracteres): 150
El color de la unidad no debe contener números ni caracteres especiales. Por favor, introduzca el color de la unidad únicamente con letras.

Introduzca el color de la unidad (máximo 15 caracteres): █
AlanBranch* 0 0 0 0

```

Figura 9.5*Intento de más de 15 caracteres*

```

Introduzca el color de la unidad (máximo 15 caracteres): masdequincecaracteres
El máximo de caracteres permitidos es de 15. Por favor, introduzca el color de la unidad con una longitud de caracteres válida.

Introduzca el color de la unidad (máximo 15 caracteres): █
AlanBranch* 0 0 0 0

```

Figura 9.6*Intento de caracteres especiales*

```

Introduzca el color de la unidad (máximo 15 caracteres): ?;?!"#%&|,.;
El color de la unidad no debe contener números ni caracteres especiales. Por favor, introduzca el color de la unidad únicamente con letras.

Introduzca el color de la unidad (máximo 15 caracteres): █
AlanBranch* 0 0 0 0

```

Figura 9.7*Intento de ingresar un dato no existente en las opciones disponibles*

```

PS C:\Evidencia2> & C:/Users/Alexis/AppData/Local/Microsoft/WindowsApps/python3.11.exe c:/Evidencia2/main2.py
MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 6

La clave 6 no es válida debido a que se encuentra fuera del rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: █

```

Figura 9.8*Intento de ingresar una palabra en menú principal*

```

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: registro

La clave registro no es válida debido a que no es un valor numérico. Por favor, introduzca alguna de las claves de procesos válidas.

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: █

```

Figura 9.9*Intento de campo vacío*

```

Menú de procesos > Menú de registro > Registro de Cliente
REGISTRO - CLIENTE -----
- Introduzca 'c' para cancelar el proceso -
Introduzca los apellidos del cliente (máximo 40 caracteres):
Este campo es obligatorio. Por favor, introduzca los apellidos del cliente.

Introduzca los apellidos del cliente (máximo 40 caracteres): █

```

Figura 9.10*Intento de inserción datos inválidos "123" y excepción*

```

Introduzca los apellidos del cliente (máximo 40 caracteres): 123
Los apellidos no deben contener números ni caracteres especiales. Por favor, introduzca los apellidos del cliente únicamente con letras.

Introduzca los apellidos del cliente (máximo 40 caracteres): █

```

Figura 9.11*Intento de inserción más de 40 caracteres*

```

Introduzca los apellidos del cliente (máximo 40 caracteres): más de cuarenta caracteres en el apellido es mucho
El máximo de caracteres permitidos es de 40. Por favor, introduzca los apellidos del cliente con una longitud de caracteres válida.

Introduzca los apellidos del cliente (máximo 40 caracteres): █

```

Figura 9.12*Intento campo vacío*

```

Introduzca los apellidos del cliente (máximo 40 caracteres): Loera alvarez
Introduzca los nombres del cliente (máximo 40 caracteres):
Este campo es obligatorio. Por favor, introduzca los nombres del cliente.

Introduzca los nombres del cliente (máximo 40 caracteres): 

```

Figura 9.13*Intento de inserción más de 40 caracteres*

```

Introduzca los nombres del cliente (máximo 40 caracteres): mas de cuarenta caracteres en el nombre es mucho
El máximo de caracteres permitidos es de 40. Por favor, introduzca los nombres del cliente con una longitud de caracteres válida.

Introduzca los nombres del cliente (máximo 40 caracteres): 

```

Figura 9.14*Intento inserción caracteres especiales*

```

Introduzca los nombres del cliente (máximo 40 caracteres): !"#%&'()*=?;
Los nombres no deben contener números ni caracteres especiales. Por favor, introduzca los nombres del cliente únicamente con letras.

Introduzca los nombres del cliente (máximo 40 caracteres): 

```

Figura 9.15*Intento de inserción de números*

```

Introduzca los nombres del cliente (máximo 40 caracteres): 2+2
Los nombres no deben contener números ni caracteres especiales. Por favor, introduzca los nombres del cliente únicamente con letras.

Introduzca los nombres del cliente (máximo 40 caracteres): 

```

Figura 9.16*Intento de campo vacío*

```

Introduzca el teléfono del cliente (10 caracteres):
Este campo es obligatorio. Por favor, introduzca el teléfono del cliente.

Introduzca el teléfono del cliente (10 caracteres): 

```

Figura 9.17*Intento inserción de cadena de texto*

```

Introduzca el teléfono del cliente (10 caracteres): Telefono
El teléfono no debe contener letras ni caracteres especiales. Por favor, introduzca el teléfono del cliente únicamente con números.

Introduzca el teléfono del cliente (10 caracteres): 

```


Figura 9.18*Intento inserción de un solo número*

```

Introduzca el teléfono del cliente (10 caracteres): 1
La longitud de caracteres del teléfono es diferente a 10. Por favor, introduzca el teléfono del cliente con una longitud de caracteres válida.
Introduzca el teléfono del cliente (10 caracteres): █

```

Figura 9.19*Intento inserción más de 10 dígitos*

```

Introduzca el teléfono del cliente (10 caracteres): 123456789101112
La longitud de caracteres del teléfono es diferente a 10. Por favor, introduzca el teléfono del cliente con una longitud de caracteres válida.
Introduzca el teléfono del cliente (10 caracteres): █

```

Figura 9.20*Intento Campo vacío*

```

Introduzca el folio del préstamo a retornar:
Este campo es obligatorio. Por favor, introduzca el folio del préstamo.
Introduzca el folio del préstamo a retornar: █

```

Figura 9.21*Intento inserción de cadena de texto*

```

Introduzca el folio del préstamo a retornar: uno
El folio no puede contener letras ni caracteres especiales. Por favor, introduzca el folio del préstamo únicamente con números.
Introduzca el folio del préstamo a retornar: █

```

Figura 9.22*Intento inserción valores negativos*

```

Introduzca el folio del préstamo a retornar: -15
El folio no puede contener letras ni caracteres especiales. Por favor, introduzca el folio del préstamo únicamente con números.
Introduzca el folio del préstamo a retornar: █

```

Figura 9.23*Intento inserción valor no registrado*

```

Introduzca el folio del préstamo a retornar: 3
Este folio no se encuentra registrado. Por favor, introduzca algún folio de préstamo ya registrado.
Introduzca el folio del préstamo a retornar: █

```

Figura 9.24*Inserción de fecha anterior a fecha de préstamo*

```

Introduzca la fecha del retorno del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual): 09-11-2024
La fecha del retorno no puede ser inferior a la fecha del préstamo. Por favor, introduzca una fecha válida o cancele el proceso con 'c'.
Introduzca la fecha del retorno del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual): █

```

Figura 9.25*Intento Campo vacío*

```

Introduzca el folio del préstamo a retornar:
Este campo es obligatorio. Por favor, introduzca el folio del préstamo.
Introduzca el folio del préstamo a retornar: █

```

Figura 9.26*Intento inserción de cadena de texto*

```

Introduzca el folio del préstamo a retornar: uno
El folio no puede contener letras ni caracteres especiales. Por favor, introduzca el folio del préstamo únicamente con números.
Introduzca el folio del préstamo a retornar: █

```

Figura 9.27*Intento inserción valores negativos*

```

Introduzca el folio del préstamo a retornar: -15
El folio no puede contener letras ni caracteres especiales. Por favor, introduzca el folio del préstamo únicamente con números.
Introduzca el folio del préstamo a retornar: █

```

Figura 9.28*Intento inserción valor no registrado*

```

Introduzca el folio del préstamo a retornar: 3
Este folio no se encuentra registrado. Por favor, introduzca algún folio de préstamo ya registrado.
Introduzca el folio del préstamo a retornar: █

```

Figura 9.29*Inserción de clave de unidad prestada anteriormente*

```

Introduzca la clave de la unidad: 1
Actualmente esta unidad no se encuentra disponible. Por favor, introduzca la clave de alguna otra unidad.
Introduzca la clave de la unidad: █

```

Figura 9.30*Intento de inserción de fecha anterior a la actual*

```

Introduzca la fecha de registro del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual): 09-11-2024
No se puede registrar una fecha pasada. Por favor, introduzca una fecha válida.

Introduzca la fecha de registro del préstamo en formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual): 

```

Figura 9.31*Dejar el Campo Vacío*

```

Introduzca la cantidad de días del préstamo:
Este campo es obligatorio. Por favor, introduzca la cantidad de días del préstamo.

Introduzca la cantidad de días del préstamo: 

```

Figura 9.32*Inserción de dato invalido "15" en días del préstamo*

```

Introduzca la cantidad de días del préstamo: 15
La cantidad mínima de días del préstamo debe ser de 1 día y máximo 14 días. Por favor, introduzca una cantidad de días del préstamo que se encuentre dentro de ese rango.

Introduzca la cantidad de días del préstamo: 

```

Figura 9.33*Inserción de cadena de texto en campo*

```

La clave Registro no es válida debido a que no es un valor numérico. Por favor, introduzca alguna de las claves de procesos válidas.

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 

```

Figura 9.34*Intento Inserción clave no valida*

```

La clave 6 no es válida debido a que se encuentra fuera del rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 

```

10 Cada vez que se le solicite al usuario una clave de unidad o cliente se le debe presentar un listado que resuma el actual catálogo de dicho dato, esto se refleja en la figura 10.

Figura 10.0

Se muestra el listado que resume el actual catalogo del dato

```
Introduzca el color de la unidad (máximo 15 caracteres): Rojo
REGISTRO EXITOSO - Se ha registrado una unidad con los datos:

=====
CLAVE      RODADA      COLOR
=====
  1         20         Rojo
=====
```

11 Todos los reportes se deben presentar en pantalla en forma tabular y, posteriormente, ofrecerse al usuario la opción de exportar el reporte presentado en formato CSV o MSeExcel, y esto se refleja en la figura 11.

Figura 11.0

Se obtiene el reporte completo de clientes y se muestra el reporte tabular y las opciones a csv y Excel

```

Menú de procesos > Menú de informes
MENÚ DE INFORMES -----
1- Reportes.
2- Análisis.
3- Regresar.
Introduzca la clave la operación a realizar: 1

Menú de procesos > Menú de reportes
MENÚ DE REPORTES -----
1- Clientes.
2- Listado de Unidades.
3- Retrasos.
4- Préstamos por retornar.
5- Préstamos por periodo.
6- Regresar.
Introduzca la clave la operación a realizar: 1

Menú de procesos > Menú de informes > Menú de reportes > Reporte de Clientes
-Menú de Clientes
1- Reporte Completo de clientes.
2- Cliente específico.
3- Regresar
Introduzca la clave de la operación a realizar: 1

Menú de procesos > Menú de informes > Menú de reportes > Reporte de Clientes > Reporte Completo de clientes
-Menú de Reportes de Cliente

=====
CLIENTES
=====
CLAVE          NOMBRE          APELLIDOS          TELÉFONO
=====
1              Denis Franco    Loera Alvarez      1231231231
=====

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: █

```

Figura 11.1

Insertión de clave para Excel

```

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: 1

EXPORTACIÓN EXITOSA - El archivo Reporte_Clientes.xlsx se ha generado exitosamente.

```

Figura 11.2*Reporte generado en Excel*

CLAVE	APELLIDOS	NOMBRES	TELÉFONO
1	Denis Franco	Loera Alvarez	1231231231

Figura 11.3*Reporte generado en csv*

```

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: 2

EXPORTACIÓN EXITOSA - El archivo Reporte_Clientes.csv se ha generado exitosamente.

```

Figura 11.4*Archivo csv generado*

CLAVE	APELLIDOS	NOMBRES	TELÉFONO
1	Denis Franco	Loera Alvarez	1231231231
3			

Figura 11.5

Reporte de unidades tabular con opción de csv y Excel

```
Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades
LISTADO DE UNIDADES -----
1- Completo.
2- Por rodada.
3- Por color.
4- Regresar.
Introduzca la clave del proceso a realizar: 1

Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades > Completo

UNIDADES REGISTRADAS
=====
CLAVE    RODADA    COLOR
=====
1        20        Rojo
2        26        Azul
3        29        Amarillo
4        26        Verde
=====

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: █
```

Figura 11.6

Reporte tabular de unidades registrada por rodada y opciones para csv y Excel

```
Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades
LISTADO DE UNIDADES -----
1- Completo.
2- Por rodada.
3- Por color.
4- Regresar.
Introduzca la clave del proceso a realizar: 2

Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades > Rodada
- Introduzca 'c' para cancelar el proceso -
Introduce el número de rodada para mostrar las bicis registradas de es tipo. (20,26,29): 20

UNIDADES REGISTRADAS DE RODADA 20
=====
CLAVE    COLOR
=====
1        Rojo
=====

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: █
```

Figura 11.7

Reporte tabular de unidades registradas por color y opciones para exportar a csv y Excel

```

Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades
LISTADO DE UNIDADES -----
1- Completo.
2- Por rodada.
3- Por color.
4- Regresar.
Introduzca la clave del proceso a realizar: 3

Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades > Color
- Introduzca 'c' para cancelar el proceso -
Introduce el color para mostrar las bicis registradas de ese color: Rojo

UNIDADES REGISTRADAS DE COLOR ROJO
=====
CLAVE          RODADA
=====
1              20
=====

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: █

```

Figura 11.8

Reporte tabular de préstamos con retornos retrasados y opciones para exportar a csv y Excel

```

Menú de procesos > Menú de reportes
MENU DE REPORTES -----
1- Clientes.
2- Listado de Unidades.
3- Retrasos.
4- Préstamos por retornar.
5- Préstamos por periodo.
6- Regresar.
Introduzca la clave la operación a realizar: 3

Menú de procesos > Menú de informes > Menú de reportes > Retrasos

PRESTAMOS CON RETORNOS RETRASADOS
=====
DIAS RETRASO  FECHA LIMITE  CLAVE UNIDAD  RODADA  COLOR  NOMBRE COMPLETO  TELEFONO
=====
1            10-12-2024    5            20      Rosa  Denise Franco Loera Alvarez  0102030405
=====

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: █

```


Figura 11.9

Reporte tabular de préstamos por retornar y opciones para exportar a csv y Excel

Menú de procesos > Menú de informes > Menú de reportes > Reporte de Préstamos por retornar

REPORTE DE PRÉSTAMOS POR RETORNAR-----

- Introduzca 'c' para cancelar el proceso -

Introduzca la fecha inicial con la que se filtrarán los datos en formato mm-dd-yyyy (deje el campo vacío para asignar la fecha actual):

Introduzca la fecha final con la que se filtrarán los datos en formato mm-dd-yyyy (deje el campo vacío para asignar la fecha actual): 10-15-2024

PRESTAMOS POR RETORNAR

FOLIO	CLAVE DE UNIDAD	CLAVE DE CLIENTE	FECHA DEL PRÉSTAMO	CANTIDAD DE DÍAS DEL PRÉSTAMO
3	1	3	10-11-2024	5
4	2	3	10-12-2024	1

1- Guardar archivo Excel.

2- Guardar archivo CSV.

3- Regresar.

Introduzca la clave del proceso a realizar:

Figura 11.10

Reporte tabular de préstamos por periodo y opciones para exportar a csv y Excel

Menú de procesos > Menú de informes > Menú de reportes > Reporte de Préstamos por periodo

REPORTE DE PRÉSTAMOS POR PERIODO-----

- Introduzca 'c' para cancelar el proceso -

Introduzca la fecha inicial con la que se filtrarán los datos en formato mm-dd-yyyy (deje el campo vacío para asignar la fecha actual):

Introduzca la fecha final con la que se filtrarán los datos en formato mm-dd-yyyy (deje el campo vacío para asignar la fecha actual): 11-30-2024

PRÉSTAMOS POR PERIODO

FOLIO	CLAVE DE UNIDAD	CLAVE DE CLIENTE	FECHA DEL PRÉSTAMO	CANTIDAD DE DÍAS DEL PRÉSTAMO	FECHA DE RETORNO DE LA UNIDAD
1	1	1	10-11-2024	4	10-11-2024
2	2	2	10-11-2024	5	10-15-2024
3	1	3	10-11-2024	5	
4	2	3	10-12-2024	1	
5	5	1	10-11-2024	1	10-13-2024

1- Guardar archivo Excel.

2- Guardar archivo CSV.

3- Regresar.

Introduzca la clave del proceso a realizar:

Especificación de los reportes:

12 Reporte completo de clientes: Un listado de todos los clientes registrados hasta ese momento que incluya cada atributo de ellos. Esto se muestra en la figura 12.

Figura 12.0

A partir del menú de reportes, entramos al menú de Reportes de clientes y creamos un Reporte completo de clientes.

```
Menú de procesos > Menú de reportes
MENÚ DE REPORTES -----
1- Clientes.
2- Listado de Unidades.
3- Retrasos.
4- Préstamos por retornar.
5- Préstamos por periodo.
6- Regresar.
Introduzca la clave la operación a realizar: 1

Menú de procesos > Menú de informes > Menú de reportes > Reporte de Clientes
-Menú de Clientes
1- Reporte Completo de clientes.
2- Cliente específico.
3- Regresar
Introduzca la clave de la operación a realizar: 1

Menú de procesos > Menú de informes > Menú de reportes > Reporte de Clientes > Reporte Completo de clientes
-Menú de Reportes de Cliente

=====
CLIENTES
=====
CLAVE          NOMBRE          APELLIDOS          TELÉFONO
=====
1              Denis Franco    Loera Alvarez       1231231231
2              Alan Alexis     Chávez Juárez       1231231234
3              Omar De Jesús   Garza               1231231235
=====
```

Figura 12.1

Generamos el archivo .xlsx.

```
1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: 1

EXPORTACIÓN EXITOSA - El archivo Reporte_Clientes.xlsx se ha generado exitosamente.
```

Figura 12.2

Abrimos el archivo .xlsx.

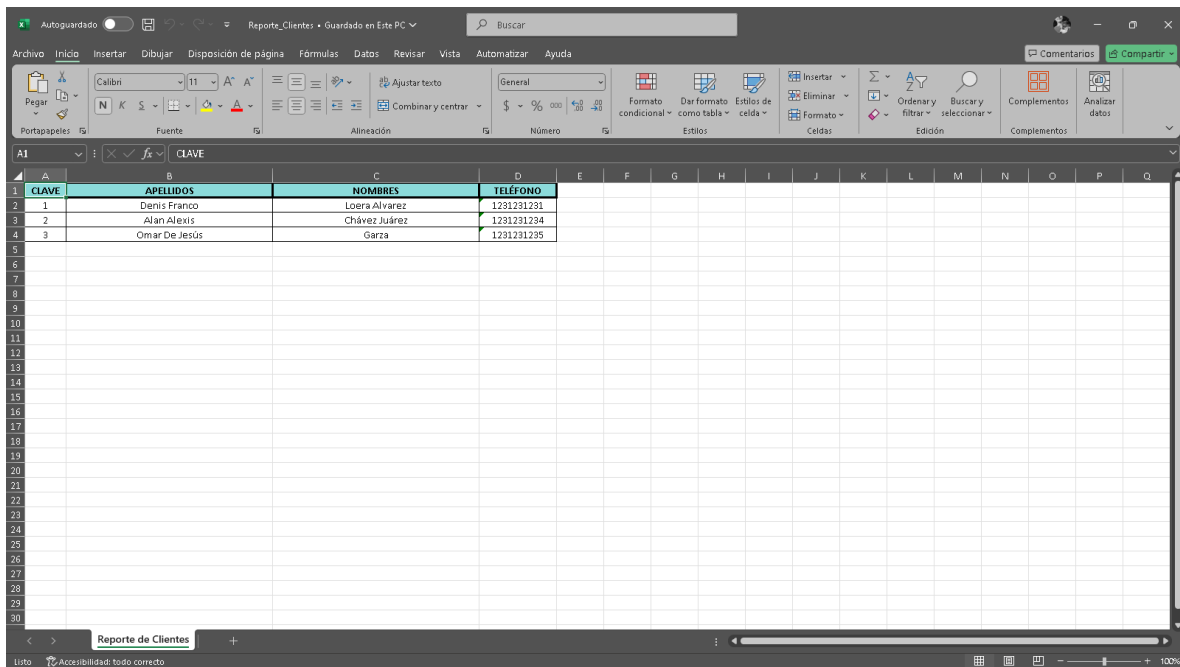


Figura 12.3

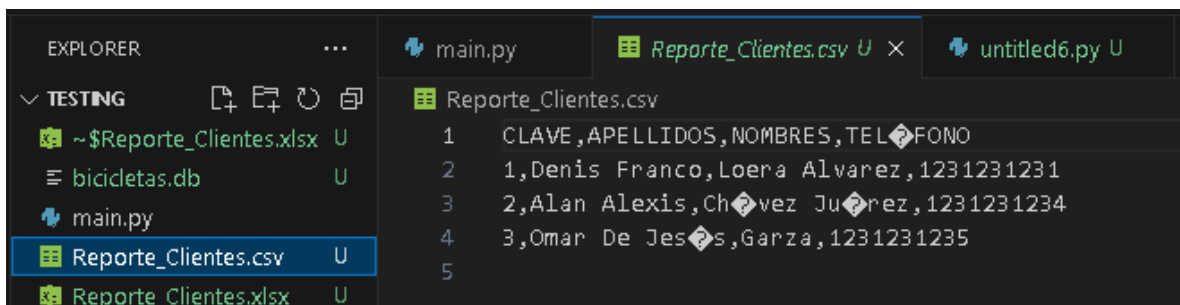
Tras ingresar nuevamente al Reporte de clientes completo, generamos el archivo .csv.

```
1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: 2

EXPORTACIÓN EXITOSA - El archivo Reporte_Clientes.csv se ha generado exitosamente.
```

Figura 12.4

Abrimos el archivo .csv.



13 Reporte de cliente específico: Se mostrarán los detalles completos de un cliente identificado por su clave y a continuación un reporte tabular con todo su historial de préstamos y devoluciones de unidades ordenado por fecha de préstamo de manera ascendente. Esto se muestra en la figura 12

Figura 13.0

Tras seleccionar el Reporte de cliente por clave, dentro del menú de Reporte de clientes, en el Menú de Reportes, introducimos una de las claves de clientes que se nos presentan.

```
Menú de procesos > Menú de reportes
MENÚ DE REPORTES -----
1- Clientes.
2- Listado de Unidades.
3- Retrasos.
4- Préstamos por retornar.
5- Préstamos por periodo.
6- Regresar.
Introduzca la clave la operación a realizar: 1

Menú de procesos > Menú de informes > Menú de reportes > Reporte de Clientes
-Menú de Clientes
1- Reporte Completo de clientes.
2- Cliente específico.
3- Regresar
Introduzca la clave de la operación a realizar: 2

Menú de procesos > Menú de informes > Menú de reportes > Reporte de Clientes > Reporte Cliente por clave
-Menú de Reportes de Cliente
- Introduzca 'c' para cancelar el proceso -
```

CLIENTES			
CLAVE	NOMBRES	APELLIDOS	TELÉFONO
1	Denís Franco	Loera Alvarez	1231231231
2	Alan Alexis	Chávez Juárez	1231231234
3	Omar De Jesús	Garza	1231231235

```
Introduzca la clave del cliente específico:
```

Figura 13.1

Validación de contenido tipo cadena

```
Introduzca la clave del cliente específico: a
La clave del cliente no puede contener Letras ni caracteres especiales

Menú de procesos > Menú de informes > Menú de reportes > Reporte de Clientes > Reporte Cliente por clave
-Menú de Reportes de Cliente
- Introduzca 'c' para cancelar el proceso -
```

CLIENTES			
CLAVE	NOMBRES	APELLIDOS	TELÉFONO
1	Denís Franco	Loera Alvarez	1231231231
2	Alan Alexis	Chávez Juárez	1231231234
3	Omar De Jesús	Garza	1231231235

```
Introduzca la clave del cliente específico:
```

Figura 13.2

Validación de cliente inexistente.

```

Introduzca la clave del cliente específico: 12
No se encontró ningún cliente con la clave proporcionada.

Menú de procesos > Menú de informes > Menú de reportes > Reporte de Clientes > Reporte Cliente por clave
-Menú de Reportes de Cliente
- Introduzca 'c' para cancelar el proceso -

                                CLIENTES
=====
CLAVE          NOMBRES          APELLIDOS          TELÉFONO
=====
1              Denis Franco      Loera Alvarez      1231231231
2              Alan Alexis       Chávez Juárez      1231231234
3              Omar De Jesús      Garza              1231231235
=====

Introduzca la clave del cliente específico:

```

Figura 13.3

Se introduce una clave de un cliente existente. Cliente clave 1, Denis Franco.

```

Introduzca la clave del cliente específico:1
Detalles del cliente:
Nombres: Denis Franco
Apellidos: Loera Alvarez
Teléfono: 1231231231

                                HISTORIAL DE PRÉSTAMOS Y RETORNOS
=====
FOLIO PRESTAMO  CLAVE UNIDAD  FECHA PRESTAMO  DIAS PRESTAMO  FECHA RETORNO  FOLIO RETORNO
=====
1              1          10-27-2024      5              10-27-2024      1
2              1          10-27-2024      5              10-27-2024      1
3              2          11-10-2024      3              No Retornado    N/A
5              4          11-10-2024      8              No Retornado    N/A
=====

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar:

```

Figura 13.4

Se genera el archivo .xlsx.

```

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: 1

EXPORTACIÓN EXITOSA - El archivo Historial_1.xlsx se ha generado exitosamente.

```

Figura 13.5

Abrimos el archivo .xlsx.

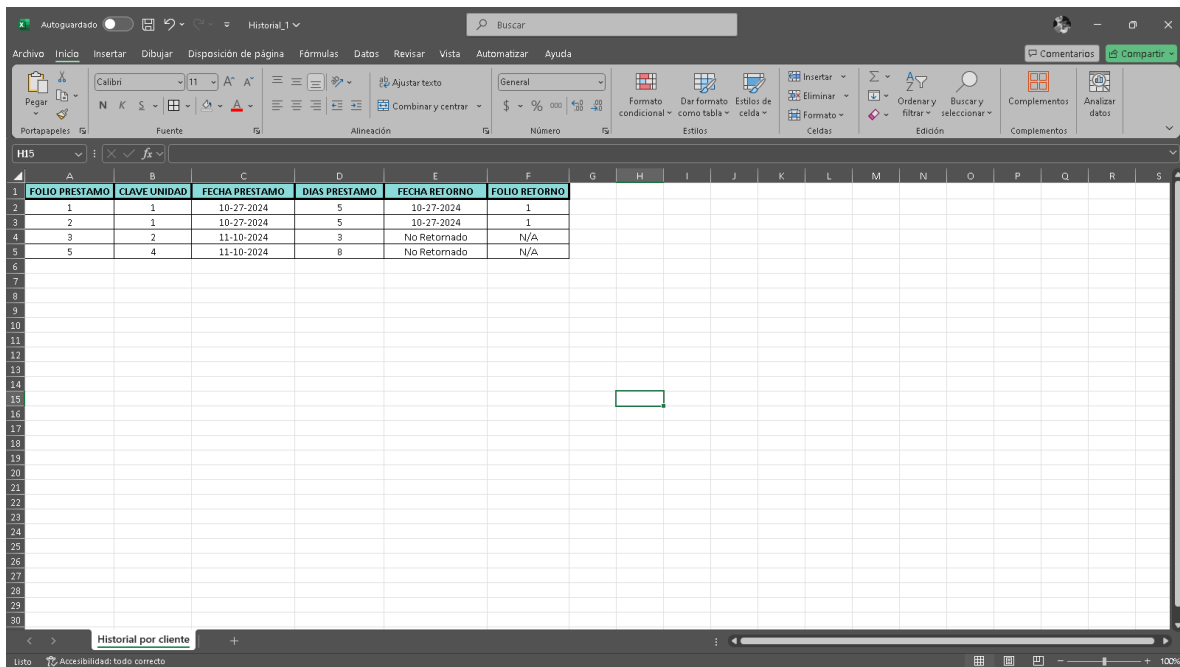


Figura 13.6

Tras ingresar nuevamente al Reporte de cliente por clave, generamos el archivo .csv.

```

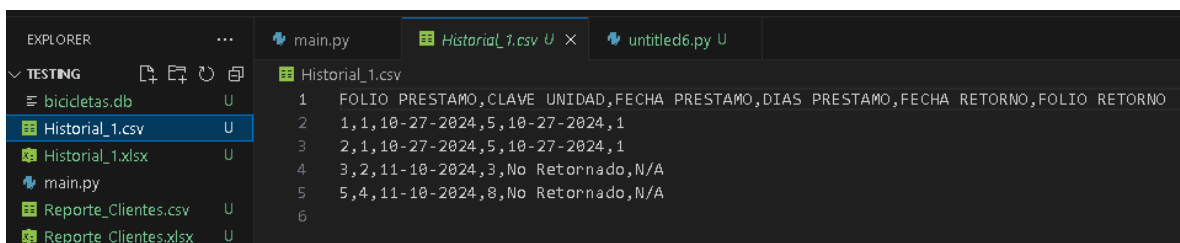
1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: 2

EXPORTACIÓN EXITOSA - El archivo Historial_1.csv se ha generado exitosamente.

```

Figura 13.7

Abrimos el archivo .csv.



14 Listado de unidades completo: Un listado de todas las unidades registradas hasta ese momento que incluya cada atributo de ellas. Esto se verifica en la figura 14.

Figura 14.0

Inserción de clave 2 para listado de unidades seguido de inserción de clave 1 para listado completo

```
Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades
LISTADO DE UNIDADES -----
1- Completo.
2- Por rodada.
3- Por color.
4- Regresar.
Introduzca la clave del proceso a realizar: 1

Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades > Completo

UNIDADES REGISTRADAS
=====
CLAVE    RODADA    COLOR
=====
1        20        Rojo
2        26        Azul
3        29        Amarillo
4        26        Verde
=====

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: █
```

15 Listado de unidades por rodada: Un listado de todas las unidades registradas hasta ese momento que correspondan a la rodada que indique el usuario que presente la clave y color de cada unidad reportada. Esto se ve en la figura 15.

Figura 15.0

Inserción de clave 3 para regresar al menú anterior y seleccionar listado de unidades por rodada

```

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: 3

Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades
LISTADO DE UNIDADES -----
1- Completo.
2- Por rodada.
3- Por color.
4- Regresar.
Introduzca la clave del proceso a realizar: █

```

Figura 15.1

Inserción de clave 2 para seleccionar reporte por rodada y después introducir clave de rodada

```

Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades
LISTADO DE UNIDADES -----
1- Completo.
2- Por rodada.
3- Por color.
4- Regresar.
Introduzca la clave del proceso a realizar: 2

Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades > Rodada
- Introduzca 'c' para cancelar el proceso -
Introduce el número de rodada para mostrar las bicis registradas de es tipo. (20,26,29): 20

UNIDADES REGISTRADAS DE RODADA 20
=====
      CLAVE      COLOR
=====
          1      Rojo
=====

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: █

```


16 Listado de unidades por color: Un listado de todas las unidades registradas hasta ese momento que correspondan al color que indique el usuario presentando la clave y rodada de cada unidad reportada. Esto se muestra en la figura 16.

Figura 16.0

Inserción de clave 3 para listado de bicicletas por color después introducir el color para el listado tabular.

```
Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades
LISTADO DE UNIDADES -----
1- Completo.
2- Por rodada.
3- Por color.
4- Regresar.
Introduzca la clave del proceso a realizar: 3

Menú de procesos > Menú de informes > Menú de reportes > Listado de unidades > Color
- Introduzca 'c' para cancelar el proceso -
Introduce el color para mostrar las bicis registradas de ese color: Rojo

UNIDADES REGISTRADAS DE COLOR ROJO
=====
CLAVE          RODADA
=====
1              20
=====

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: █
```

17 Retrasos: Un reporte de todos los préstamos realizados y, que a la fecha actual, presenten un retraso en su entrega; ordenados de mayor a menor retraso, presentando los datos de: Días de retraso, fecha en que se debió haber retornado la unidad, clave de unidad, rodada, color, nombre completo del cliente y su teléfono de contacto. Se muestra en la figura 17.

Figura 17.0

Inserción de clave para retrasos “3” y reporte tabular de retrasos.

```

Menú de procesos > Menú de reportes
MENU DE REPORTES -----
1- Clientes.
2- Listado de Unidades.
3- Retrasos.
4- Préstamos por retornar.
5- Préstamos por periodo.
6- Regresar.
Introduzca la clave la operación a realizar: 3

Menú de procesos > Menú de informes > Menú de reportes > Retrasos

=====
PRESTAMOS CON RETORNIOS RETRASADOS
=====
DIAS RETRASO  FECHA LIMITE  CLAVE UNIDAD  RODADA  COLOR  NOMBRE COMPLETO  TELEFONO
=====
1            10-12-2024      5            20      Rosa  Denise Franco Loera Alvarez  0102030405
=====

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar: 

```

18 Préstamos por retornar: Un reporte de todos los préstamos que estén pendientes de retornar y que se hayan prestado durante el período que indique el usuario. Debe presentar la clave de unidad, rodada, fecha del préstamo, nombre completo del cliente y el teléfono del cliente. Se refleja en la figura 18.

Figura 18.0

Insertión de clave 4 para préstamos por retornar y siguiente paso: fecha inicial con la que se filtraran los datos

```
3- Regresar.
Introduzca la clave del proceso a realizar: 3

Menú de procesos > Menú de reportes
MENÚ DE REPORTES -----
1- Clientes.
2- Listado de Unidades.
3- Retrasos.
4- Préstamos por retornar.
5- Préstamos por periodo.
6- Regresar.
Introduzca la clave la operación a realizar: 4

Menú de procesos > Menú de informes > Menú de reportes > Reporte de Préstamos por retornar
REPORTE DE PRÉSTAMOS POR RETORNAR-----
- Introduzca 'c' para cancelar el proceso -
Introduzca la fecha inicial con la que se filtrarán los datos en formato mm-dd-yyyy (deje el campo va
cío para asignar la fecha actual):
```

Figura 18.1

Insertión de fecha actual como fecha inicio y fecha final 10-15-2024 con reporte tabular de préstamos por retornar

```
Menú de procesos > Menú de informes > Menú de reportes > Reporte de Préstamos por retornar
REPORTE DE PRÉSTAMOS POR RETORNAR-----
- Introduzca 'c' para cancelar el proceso -
Introduzca la fecha inicial con la que se filtrarán los datos en formato mm-dd-yyyy (deje el campo vacío para asignar la fecha actual):
Introduzca la fecha final con la que se filtrarán los datos en formato mm-dd-yyyy (deje el campo vacío para asignar la fecha actual): 10-15-2024

=====
PRESTAMOS POR RETORNAR
=====
FOLIO      CLAVE DE UNIDAD  CLAVE DE CLIENTE  FECHA DEL PRÉSTAMO  CANTIDAD DE DÍAS DEL PRÉSTAMO
=====
3          1                3                10-11-2024         5
4          2                3                10-12-2024         1
=====

1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar:
```

19 Préstamos por período: Un reporte de todos los préstamos realizados durante el período que indique el usuario. Debe presentar la clave de unidad, rodada, fecha del préstamo, nombre completo del cliente y el teléfono del cliente. Esto es presentado en la figura 18.

Figura 19.0

Inserción de clave para préstamos por periodo y siguiente paso: paso inicial con la que se filtraran los datos.

```
Menú de procesos > Menú de reportes
MENÚ DE REPORTES -----
1- Clientes.
2- Listado de Unidades.
3- Retrasos.
4- Préstamos por retornar.
5- Préstamos por periodo.
6- Regresar.
Introduzca la clave la operación a realizar: 5

Menú de procesos > Menú de informes > Menú de reportes > Reporte de Préstamos por periodo
REPORTE DE PRÉSTAMOS POR PERIODO-----
- Introduzca 'c' para cancelar el proceso -
Introduzca la fecha inicial con la que se filtrarán los datos en formato mm-dd-yyyy (deje el campo va
cío para asignar la fecha actual):
```

Figura 19.1

Inserción de fechas para reporte de préstamos por periodo y reporte tabular.

```
Menú de procesos > Menú de informes > Menú de reportes > Reporte de Préstamos por periodo
REPORTE DE PRÉSTAMOS POR PERIODO-----
- Introduzca 'c' para cancelar el proceso -
Introduzca la fecha inicial con la que se filtrarán los datos en formato mm-dd-yyyy (deje el campo vacío para asignar la fecha actual):
Introduzca la fecha final con la que se filtrarán los datos en formato mm-dd-yyyy (deje el campo vacío para asignar la fecha actual): 11-30-2024

PRÉSTAMOS POR PERIODO
=====
```

FOLIO	CLAVE DE UNIDAD	CLAVE DE CLIENTE	FECHA DEL PRÉSTAMO	CANTIDAD DE DÍAS DEL PRÉSTAMO	FECHA DE RETORNO DE LA UNIDAD
1	1	1	10-11-2024	4	10-11-2024
2	2	2	10-11-2024	5	10-15-2024
3	1	3	10-11-2024	5	
4	2	3	10-12-2024	1	
5	5	1	10-11-2024	1	10-13-2024

```
=====
1- Guardar archivo Excel.
2- Guardar archivo CSV.
3- Regresar.
Introduzca la clave del proceso a realizar:
```

Especificación de los análisis, esto se refleja en la figura 20.

Figura 20.0

Inserción de clave para análisis "2"

```
Menú de procesos > Menú de informes
MENÚ DE INFORMES -----
1- Reportes.
2- Análisis.
3- Regresar.
Introduzca la clave la operación a realizar: 2

Menú de procesos > Menú de informes > Menú de análisis
MENÚ DE ANÁLISIS -----
1- Duración de los préstamos.
2- Ranking de clientes.
3- Preferencia de rentas.
4- Regresar.
Introduzca la clave la operación a realizar: █
```

21 Duración de los préstamos: Debe presentar un reporte de estadísticos descriptivos de la cantidad de días registrados al momento de prestar una unidad que considere todo registro de préstamo realizado; debe incluir: Media, mediana, moda, mínimo, máximo, desviación estándar y cuartiles. Y esto se ve en la figura 21.

Figura 21.0

Registros de préstamos actuales.

Table: PRESTAMO

folio	clave_unidad	clave_cliente	fecha_prestamo	dias_prestamo	fecha_retorno
Filter	Filter	Filter	Filter	Filter	Filter
1	1	1	2024-10-27	5	2024-10-27
2	2	1	2024-10-27	5	NULL
3	3	2	2024-11-10	3	NULL
4	4	3	2024-11-11	6	NULL
5	5	4	2024-11-10	8	NULL

Figura 21.1

Insertión de clave 1 para estadístico descriptivo de duración de los préstamos.

```

Menú de procesos > Menú de informes > Menú de análisis
MENÚ DE ANÁLISIS -----
1- Duración de los préstamos.
2- Ranking de clientes.
3- Preferencia de rentas.
4- Regresar.
Introduzca la clave la operación a realizar: 1

Menú de procesos > Menú de informes > Menú de análisis > Duración de los préstamos

ESTADÍSTICOS DESCRIPTIVOS - DURACIÓN DE LOS PRÉSTAMOS
=====
MEDIA      MEDIANA    MODA      MÍNIMO    MÁXIMO    DESVIACIÓN ESTÁNDAR    PRIMER CUARTIL    SEGUNDO CUARTIL    TERCER CUARTIL
=====
5.40      5.00      5         3         8         1.6248      5.00      5.00      6.00
=====

Menú de procesos > Menú de informes > Menú de análisis
MENÚ DE ANÁLISIS -----
1- Duración de los préstamos.
2- Ranking de clientes.
3- Preferencia de rentas.
4- Regresar.
Introduzca la clave la operación a realizar: 2

```

22 Ranking de clientes: Reporte con los siguientes datos de los clientes: Cantidad de rentas acumuladas desde su registro, Clave, nombre completo y teléfono. Ordenado de manera descendente según su acumulado de préstamos a la fecha del reporte. Esto se ve en la figura 22

Figura 21.0

Registros de clientes actuales.

Table: CLIENTE

	clave	apellidos	nombres	telefono
	Filter	Filter	Filter	Filter
1	1	Denis Franco	Loera Alvarez	1231231231
2	2	Alan Alexis	Chávez Juárez	1231231234
3	3	Omar De Jesús	Garza	1231231235

Figura 22.1

Insertión de clave 2 para ranking de clientes

```

Menú de procesos > Menú de informes > Menú de análisis
MENÚ DE ANÁLISIS -----
1- Duración de los préstamos.
2- Ranking de clientes.
3- Preferencia de rentas.
4- Regresar.
Introduzca la clave la operación a realizar: 2

Menú de procesos > Menú de informes > Menú de análisis > Ranking de clientes

=====
RANKING DE CLIENTES
=====
CANTIDAD DE RENTAS ACUMULADAS    CLAVE    NOMBRE COMPLETO    TELÉFONO
=====
4                                1        Loera Alvarez Denis Franco    1231231231
1                                3        Garza Omar De Jesús          1231231235
=====

Menú de procesos > Menú de informes > Menú de análisis
MENÚ DE ANÁLISIS -----
1- Duración de los préstamos.
2- Ranking de clientes.
3- Preferencia de rentas.
4- Regresar.
Introduzca la clave la operación a realizar:

```

23 Preferencias de rentas: Submenú que ofrecerá las opciones que se muestran en la figura 23.

Figura 23.0

Entrada a menú de preferencias de rentas.

```
Menú de procesos > Menú de informes > Menú de análisis > Menú de preferencias de rentas
MENÚ DE PREFERENCIAS DE RENTAS -----
1- Por rodada.
2- Por color.
3- Por día de la semana.
4- Regresar.
Introduzca la clave la operación a realizar: █
```


24 Por rodada: Cantidad acumulada de préstamos por rodada ordenado de manera descendente según su acumulado de préstamos a la fecha. Se debe presentar también una gráfica de pastel que presente la proporción dentro del total de préstamos por cada rodada ofertada.

Figura 24.0

Intento de campo vacío

```
Menú de procesos > Menú de informes > Menú de análisis > Menú de preferencias de rentas
MENÚ DE PREFERENCIAS DE RENTAS -----
1- Por rodada.
2- Por color.
3- Por día de la semana.
4- Regresar.
Introduzca la clave la operación a realizar:
Este campo es obligatorio. Por favor, introduzca alguna de las claves de procesos.

Menú de procesos > Menú de informes > Menú de análisis > Menú de preferencias de rentas
MENÚ DE PREFERENCIAS DE RENTAS -----
1- Por rodada.
2- Por color.
3- Por día de la semana.
4- Regresar.
Introduzca la clave la operación a realizar: 
```

Figura 24.1

Intento de clave fuera de rango

```
Menú de procesos > Menú de informes > Menú de análisis > Menú de preferencias de rentas
MENÚ DE PREFERENCIAS DE RENTAS -----
1- Por rodada.
2- Por color.
3- Por día de la semana.
4- Regresar.
Introduzca la clave la operación a realizar: 5
La clave 5 no es válida debido a que se encuentra fuera del rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.

Menú de procesos > Menú de informes > Menú de análisis > Menú de preferencias de rentas
MENÚ DE PREFERENCIAS DE RENTAS -----
1- Por rodada.
2- Por color.
3- Por día de la semana.
4- Regresar.
Introduzca la clave la operación a realizar: 
```

Figura 24.2

Intento de inserción de letras y excepción

```

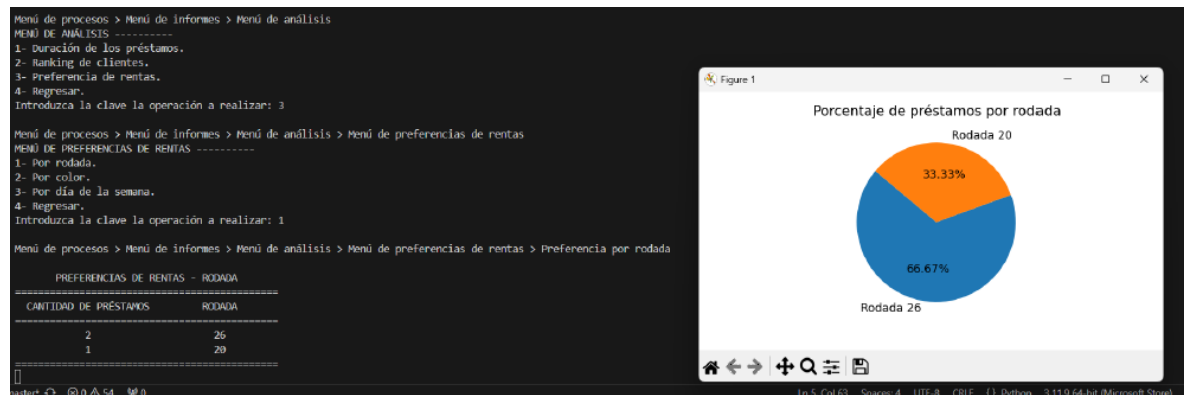
Menú de procesos > Menú de informes > Menú de análisis > Menú de preferencias de rentas
MENÚ DE PREFERENCIAS DE RENTAS -----
1- Por rodada.
2- Por color.
3- Por día de la semana.
4- Regresar.
Introduzca la clave la operación a realizar: dos
La clave dos no es válida debido a que no es un valor numérico. Por favor, introduzca alguna de las claves de procesos válidas.

Menú de procesos > Menú de informes > Menú de análisis > Menú de preferencias de rentas
MENÚ DE PREFERENCIAS DE RENTAS -----
1- Por rodada.
2- Por color.
3- Por día de la semana.
4- Regresar.
Introduzca la clave la operación a realizar:

```

Figura 24.4

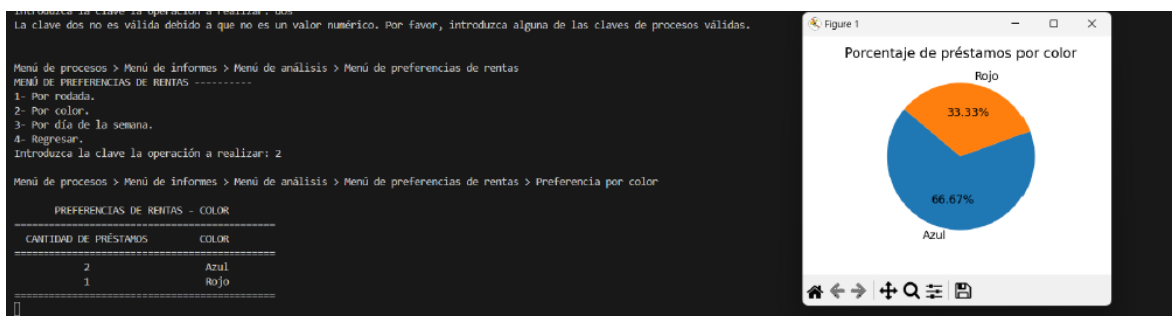
Inserción de clave valida "1" para generación de reporte por rodada. Aquí manejamos datos diferentes a los presentados con anterioridad.



25 Por color: Cantidad de préstamos por color ordenado de manera descendente según su acumulado de préstamos a la fecha. Se debe presentar también una gráfica de pastel que presente la proporción dentro del total de cada color ofertado.

Figura 25.0

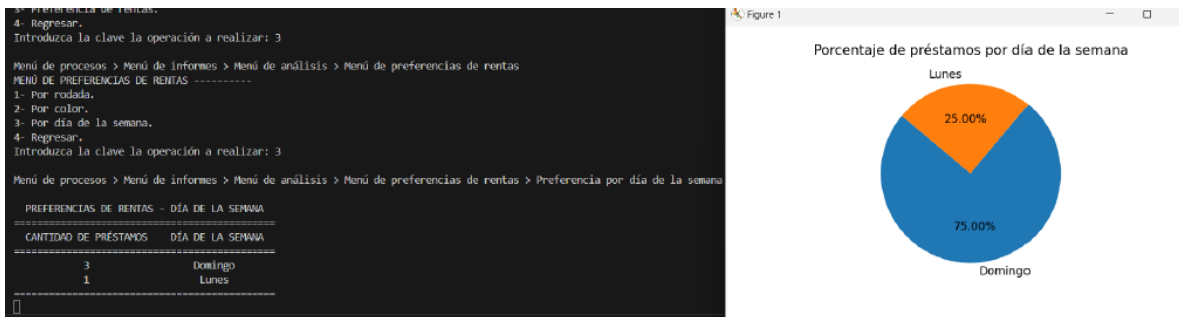
Inserción de clave valida "2" para generación de reporte porcentaje de préstamos por color.



26 Por día de la semana: Cantidad de préstamos totales acumulados hasta el momento por día de la semana, ordenado considerando que esta inicia los domingos. Se debe presentar también una gráfica de barras que presente la cantidad por cada día de la semana.

Figura 26.0

Inserción de clave valida “3” para generación de reporte porcentaje de préstamos por día de la Semana.



27 Volver al menú de análisis.

Figura 27.1

Inserción de clave 4 para regresar a menú de análisis.

```
Menú de procesos > Menú de informes > Menú de análisis > Menú de preferencias de renta  
MENÚ DE PREFERENCIAS DE RENTAS -----  
1- Por rodada.  
2- Por color.  
3- Por día de la semana.  
4- Regresar.  
Introduzca la clave la operación a realizar: █
```

28 Antes de salir del sistema se debe solicitar la correspondiente confirmación al usuario. Esto se muestra en la figura 28.

Figura 28.0

Insertión de claves para salir de sistema y confirmación para salir de sistema.

```
Menú de procesos > Menú de informes
MENÚ DE INFORMES -----
1- Reportes.
2- Análisis.
3- Regresar.
Introduzca la clave la operación a realizar: 3

MENÚ DE PROCESOS -----
1- Registro.
2- Préstamo.
3- Retorno.
4- Informes.
5- Salir.
Introduzca la clave del proceso a realizar: 5

Presione Enter para salir o introduza cualquier otro caracter para continuar: |
```

PSEUDOCÓDIGO

INICIO

Intentar conectar con la base de datos SQLite:

Crear conexión con la base de datos utilizando `sqlite3.connect()`

Intentar crear las tablas si no existen en la base de datos:

 Crear tablas UNIDAD, CLIENTE, PRESTAMO y RETORNO

 Ejecutar la consulta para crear cada tabla si no existe

Si ocurre alguna excepción al conectar o crear las tablas:

 Mostrar el error y terminar el programa

Función `main()`

 Conectar a la base de datos SQLite y crear tablas si no existen

 Ejecutar consulta SQL para cada tabla requerida (UNIDAD, CLIENTE, PRESTAMO, RETORNO)

 Si no existe

 Confirmar conexión y creación de tablas

Llamar a la función `menú_principal()`

Función `menú_principal()`

 Mientras verdadero:

 Mostrar menú principal con las opciones:

 1. Registro

 2. Préstamo

 3. Retorno

 4. Reportes

 5. Preferencias de rentas

 6. Salir

 Leer opción seleccionada

 Según opción:

 Caso "1" (Registro):

 Llamar a la función `registro()`

 Regresar al menú principal

 Caso "2" (Préstamo):

Llamar a la función prestamo()

Regresar al menú principal

Caso "3" (Retorno):

Llamar a la función retorno()

Regresar al menú principal

Caso "4" (Reportes):

Llamar a la función reportes()

Regresar al menú principal

Caso "5" (Preferencias de rentas):

Llamar a la función preferencias_rentas()

Regresar al menú principal

Caso "6" (Salir):

Cerrar la conexión con la base de datos

Mostrar mensaje de despedida

Terminar el programa

Opción inválida:

Mostrar mensaje de error y solicitar una opción válida

Función registro()

Mientras verdadero:

Mostrar menú de registro

Leer opción de registro

Si la opción no es válida:

Mostrar mensaje de error

Continuar

Según opción:

Caso "1": (Registro de Unidad)

Mostrar menú de registro de Unidad

Mientras verdadero:

Leer rodada

Si opción de cancelación:

Romper ciclo

Validar rodada

Si válida:

Leer color

Si opción de cancelación:

Romper ciclo

Validar color

Si válido:

Ejecutar consulta INSERT para agregar una nueva unidad a la tabla UNIDAD

Mostrar mensaje de éxito

Romper ciclo

Caso "2": (Registro de Cliente)

Mostrar menú de registro de Cliente

Mientras verdadero:

Leer apellidos

Si opción de cancelación:

Romper ciclo

Validar apellidos

Si válidos:

Leer nombres

Si opción de cancelación:

Romper ciclo

Validar nombres

Si válidos:

Leer teléfono

Si opción de cancelación:

Romper ciclo

Validar teléfono

Si válido:

Ejecutar consulta INSERT para agregar un nuevo cliente a la tabla CLIENTE

Mostrar mensaje de éxito

Romper ciclo

Caso "3": (Regresar)

Romper el ciclo

Función prestamo()

Verificar si existen unidades, clientes y préstamos en las tablas correspondientes:

Ejecutar consultas SELECT COUNT(*) en las tablas UNIDAD, CLIENTE y PRESTAMO

Si no hay unidades registradas en la base de datos:

Mostrar mensaje de error indicando que no hay unidades

Regresar

Si no hay clientes registrados en la base de datos:

Mostrar mensaje de error indicando que no hay clientes

Regresar

Si todas las unidades están ocupadas en préstamos no retornados:

Mostrar mensaje de error indicando que no hay unidades disponibles

Regresar

Mostrar menú de préstamo

Listar las unidades disponibles en la base de datos:

Ejecutar consulta SELECT en la tabla UNIDAD con JOIN a PRESTAMO para verificar disponibilidad

Mostrar los resultados en una tabla con:

- Clave de unidad
- Rodada
- Color

Mientras verdadero:

Leer clave de la unidad

Si opción de cancelación:

Romper el ciclo

Validar la clave de la unidad ingresada

Si no es válida:

Mostrar mensaje de error

Continuar

Listar todos los clientes registrados en la base de datos:

Ejecutar consulta SELECT en la tabla CLIENTE

Mostrar los resultados en una tabla con:

- Clave de cliente
- Nombre completo
- Teléfono

Mientras verdadero:

Leer clave del cliente

Si opción de cancelación:

Romper el ciclo

Validar la clave del cliente ingresada

Si no es válida:

Mostrar mensaje de error

Continuar

Mientras verdadero:

Leer fecha del préstamo

Si opción de cancelación:

Romper el ciclo

Validar la fecha ingresada

Si válida:

Leer cantidad de días del préstamo

Si opción de cancelación:

Romper el ciclo

Validar la cantidad de días ingresada

Si válida:

Ejecutar consulta INSERT en la tabla PRESTAMO
con los datos del préstamo:

- Clave de unidad
- Clave de cliente
- Fecha de préstamo
- Cantidad de días del préstamo

Confirmar los cambios en la base de datos
(commit)

Mostrar mensaje de éxito

Romper ciclo

Continuar

Romper ciclo

Romper ciclo

Romper ciclo

Función retorno()

Verificar si existen unidades, clientes y préstamos en las tablas correspondientes:

Ejecutar consultas SELECT COUNT(*) en las tablas UNIDAD, CLIENTE y PRESTAMO

Si no hay unidades registradas en la base de datos:

Mostrar mensaje indicando que no hay unidades registradas y terminar la función

Si no hay clientes registrados en la base de datos:

Mostrar mensaje indicando que no hay clientes registrados y terminar la función

Si no hay préstamos registrados en la base de datos:

Mostrar mensaje indicando que no hay préstamos registrados y terminar la función

Verificar si existen préstamos no retornados:

Ejecutar consulta SELECT en la tabla PRESTAMO para obtener registros donde fecha de
retorno sea NULL

Si no hay resultados:

Mostrar mensaje indicando que todas las unidades han sido retornadas

Terminar la función

Mostrar encabezado del menú de retorno

Listar préstamos no retornados en la base de datos:

Ejecutar consulta SELECT en la tabla PRESTAMO con JOIN a UNIDAD y CLIENTE,
ordenados por fecha de préstamo

Mostrar los resultados en una tabla con:

- Folio del préstamo
- Clave de unidad
- Clave de cliente
- Fecha del préstamo

- Días del préstamo

Mientras verdadero:

Leer el folio del préstamo a retornar

Si opción de cancelación:

Romper el ciclo

Validar que el folio de préstamo ingresado exista y no haya sido retornado:

Ejecutar consulta SELECT en la tabla PRESTAMO con el folio ingresado y verificar que fecha de retorno es NULL

Si el folio no existe o ya fue retornado:

Mostrar mensaje de error y continuar

Mientras verdadero:

Solicitar fecha de retorno

Si opción de cancelación:

Romper el ciclo

Si la fecha de retorno está vacía:

Asignar la fecha actual como fecha de retorno

Si no, intentar convertir la fecha ingresada al formato correcto

Si la conversión falla:

Mostrar mensaje de error indicando formato incorrecto y continuar

Validar que la fecha de retorno no sea anterior a la fecha del préstamo

Si es anterior:

Mostrar mensaje de error indicando que la fecha de retorno no puede ser anterior a la fecha de préstamo

Continuar

Actualizar la tabla PRESTAMO con la fecha de retorno ingresada:

Ejecutar consulta UPDATE para asignar fecha de retorno en el registro con el folio correspondiente

Confirmar los cambios en la base de datos (commit)

Registrar el retorno en la tabla RETORNO:

Ejecutar consulta INSERT en la tabla RETORNO con:

- Fecha de retorno

- Clave de unidad

- Clave de cliente

Confirmar los cambios en la base de datos (commit)

Mostrar mensaje de éxito indicando el retorno registrado

Romper ciclo interno

Romper ciclo externo

Función reportes()

Verificar si existen registros en las tablas CLIENTE y PRESTAMO:

Ejecutar consultas SELECT COUNT(*) en CLIENTE y PRESTAMO

Si no hay registros en ambas tablas:

Mostrar mensaje indicando que no hay registros suficientes para generar reportes

Terminar la función

Mientras verdadero:

Mostrar menú de reportes

Leer opción seleccionada

Segun opción

Caso "1": (Menú de reportes de clientes)

Mostrar opciones de reportes de cliente

- Reporte Completo de clientes

- Cliente específico

- Regresar.

Leer opción de filtro

Segun opción de filtro:

Caso "1" (Reporte Completo de clientes):

Mostrar encabezado de reporte de clientes

Obtener datos de todos los clientes desde la tabla CLIENTE:

Ejecutar consulta SELECT en CLIENTE

Mostrar los resultados en una tabla con:

- Clave de cliente

- Apellidos
- Nombres
- Teléfono

Mientras verdadero:

Mostrar opciones para exportar reporte de clientes

Leer opción de exportación

Si opción de exportación es inválida:

Mostrar mensaje de error y continuar

Según opción de exportación:

Caso "1" (Exportar a Excel):

Crear archivo Excel

Configurar encabezados y ajustar
columnas

Guardar datos del reporte de
clientes en el archivo

Guardar archivo como
"Reporte_Clientes.xlsx"

Mostrar mensaje de éxito y romper
ciclo

Caso "2" (Exportar a CSV):

Crear archivo CSV

Configurar encabezados

Guardar datos del reporte de
clientes en el archivo

Guardar archivo como
"Reporte_Clientes.csv"

Mostrar mensaje de éxito y romper
ciclo

Caso "3" (Regresar):

Romper ciclo

Caso "2":

Mostrar opciones de reportes de cliente por clave

Mostrar todos los clientes

Según opción

Leer opción

Si opción es invalida

Mostrar mensaje de error y continuar

Si opción no cumple las requirements

Mostrar mensaje de error y regresar

Si opción es válida

Ejecutar consulta SQL para obtener datos del cliente por clave

Si cliente existe:

Mostrar detalles del cliente

Ejecutar consulta SQL para obtener historial de préstamos y retornos

Para cada registro en historial:

Mostrar historial de cliente en formato tabular

Según opción de exportación:

Caso "1" (Exportar a Excel):

Crear archivo Excel

Configurar encabezados y ajustar columnas

Guardar datos del reporte de clientes en el archivo

Guardar archivo como "Reporte_Clientes.xlsx"

Mostrar mensaje de éxito y romper ciclo

Caso "2" (Exportar a CSV):

Crear archivo CSV

Configurar encabezados

Guardar datos del reporte de clientes en el archivo

Guardar archivo como "Reporte_Clientes.csv"

Mostrar mensaje de éxito y romper ciclo

Caso "3" (Regresar):

Romper ciclo

Si no:

Mostrar "No se encontró ningún cliente con la clave proporcionada."

Continuar

Caso "3":

Salir de bucle de menú de reportes de clientes

Caso "2" (Listado de Unidades):

Mostrar opciones de filtro para el listado:

- Completo
- Por rodada
- Por color

Leer opción de filtro

Según opción de filtro:

Caso "1" (Listado completo de unidades):

Obtener todos los registros de unidades desde la tabla UNIDAD

Mostrar los resultados en una tabla con:

- Clave de unidad
- Rodada
- Color

Función exportar_reporte(tipo_reporte, formato):

Si formato es "Excel":

Crear archivo Excel

Configurar encabezados y ajustar columnas de acuerdo con el tipo de reporte

Escribir datos en el archivo

Guardar archivo con el nombre correspondiente a tipo_reporte

Mostrar mensaje de éxito indicando la exportación

Si formato es "CSV":

Crear archivo CSV

Configurar encabezados de acuerdo con el tipo de reporte

Escribir datos en el archivo

Guardar archivo con el nombre correspondiente a tipo_reporte

Mostrar mensaje de éxito indicando la exportación

Caso "2" (Listado por rodada):

Solicitar rodada deseada al usuario

Obtener todas las unidades de la rodada especificada desde UNIDAD

Mostrar los resultados con:

- Clave de unidad

- Color

Función exportar_reporte(tipo_reporte, formato):

Si formato es "Excel":

Crear archivo Excel

Configurar encabezados y ajustar columnas de acuerdo con el tipo de reporte

Escribir datos en el archivo

Guardar archivo con el nombre correspondiente a tipo_reporte

Mostrar mensaje de éxito indicando la exportación

Si formato es "CSV":

Crear archivo CSV

Configurar encabezados de acuerdo con el tipo de reporte

Escribir datos en el archivo

Guardar archivo con el nombre correspondiente a tipo_reporte

Mostrar mensaje de éxito indicando la exportación

Caso "3" (Listado por color):

Solicitar color deseado al usuario

Obtener todas las unidades del color especificado desde UNIDAD

Mostrar los resultados con:

- Clave de unidad

- Rodada

Función exportar_reporte(tipo_reporte, formato):

Si formato es "Excel":

Crear archivo Excel

Configurar encabezados y ajustar columnas de acuerdo con el tipo de reporte

Escribir datos en el archivo

Guardar archivo con el nombre correspondiente a tipo_reporte

Mostrar mensaje de éxito indicando la exportación

Si formato es "CSV":

Crear archivo CSV

Configurar encabezados de acuerdo con el tipo de reporte

Escribir datos en el archivo

Guardar archivo con el nombre correspondiente a tipo_reporte

Mostrar mensaje de éxito indicando la exportación

Caso "4" (Regresar al menú de reportes):

Romper ciclo

Caso "3" (Reporte de Retrasos):

Mostrar encabezado de reporte de retrasos

Calcular retrasos en base a las fechas de préstamo y días otorgados

Obtener todos los préstamos con retrasos:

Ejecutar consulta SELECT en PRESTAMO donde fecha de retorno sea NULL y fecha límite de retorno vencida

Calcular días de retraso y preparar la lista de retrasos

Imprimir reporte tabular con:

- Días de retraso
- Fecha límite de retorno
- Clave de unidad
- Rodada
- Color
- Nombre completo del Cliente
- Teléfono del Cliente

Mostrar opciones de exportación y seguir lógica similar a "Reporte de Clientes"

Caso "4" (Reporte de Préstamos por Retornar):

Filtrar préstamos en la tabla PRESTAMO donde fecha de retorno sea NULL

Mostrar reporte en formato tabular con:

- Folio
- Clave de unidad
- Clave de cliente
- Fecha del préstamo
- Días del préstamo

M Función exportar_reporte(tipo_reporte, formato):

Si formato es "Excel":

Crear archivo Excel

Configurar encabezados y ajustar columnas de acuerdo con el tipo de reporte

Escribir datos en el archivo

Guardar archivo con el nombre correspondiente a tipo_reporte

Mostrar mensaje de éxito indicando la exportación

Si formato es "CSV":

Crear archivo CSV

Configurar encabezados de acuerdo con el tipo de reporte

Escribir datos en el archivo

Guardar archivo con el nombre correspondiente a tipo_reporte

Mostrar mensaje de éxito indicando la exportación

Caso "5" (Reporte de Préstamos por Período):

Solicitar fecha de inicio y fin del período

Validar las fechas ingresadas y filtrar préstamos en PRESTAMO en ese rango

Imprimir reporte tabular con:

- Folio
- Clave de unidad
- Clave de cliente
- Fecha del préstamo
- Días del préstamo
- Fecha de retorno (si existe)

Función exportar_reporte(tipo_reporte, formato):

Si formato es "Excel":

Crear archivo Excel

Configurar encabezados y ajustar columnas de acuerdo con el tipo de reporte

Escribir datos en el archivo

Guardar archivo con el nombre correspondiente a tipo_reporte

Mostrar mensaje de éxito indicando la exportación

Si formato es "CSV":

Crear archivo CSV

Configurar encabezados de acuerdo con el tipo de reporte

Escribir datos en el archivo

Guardar archivo con el nombre correspondiente a tipo_reporte

Mostrar mensaje de éxito indicando la exportación

Caso "6" (Regresar al menú principal):

Romper el ciclo y regresar al menú principal

Función exportar_reporte(tipo_reporte, formato):

Si formato es "Excel":

Crear archivo Excel

Configurar encabezados y ajustar columnas de acuerdo con el tipo de reporte

Escribir datos en el archivo

Guardar archivo con el nombre correspondiente a tipo_reporte

Mostrar mensaje de éxito indicando la exportación

Si formato es "CSV":

Crear archivo CSV

Configurar encabezados de acuerdo con el tipo de reporte

Escribir datos en el archivo

Guardar archivo con el nombre correspondiente a tipo_reporte

Mostrar mensaje de éxito indicando la exportación

Función preferencias_rentas()

Mostrar submenú de preferencias de rentas:

1. Por rodada
2. Por color
3. Por día de la semana
4. Volver al menú de análisis

Leer opción del submenú

Según opción:

Caso "1" (Por rodada):

Ejecutar consulta SELECT en PRESTAMO con JOIN en UNIDAD para contar préstamos por rodada

Ordenar resultados por cantidad en orden descendente

Mostrar cantidad de préstamos por rodada en formato tabular

Crear gráfica de pastel con proporción de cada rodada en el total de préstamos

Mostrar gráfica al usuario

Esperar que el usuario cierre la grafica para continuar

Caso "2" (Por color):

Ejecutar consulta SELECT en PRESTAMO con JOIN en UNIDAD para contar préstamos

por color

Ordenar resultados por cantidad en orden descendente

Mostrar cantidad de préstamos por color en formato tabular

Crear gráfica de pastel con proporción de cada color en el total de préstamos

Mostrar gráfica al usuario

Esperar que el usuario cierre la grafica para continuar

Caso "3" (Por día de la semana):

Ejecutar consulta SELECT en PRESTAMO para contar préstamos por día de la semana

Ordenar resultados de acuerdo a días de la semana (domingo a sábado)

Mostrar cantidad de préstamos por día en formato tabular

Crear gráfica de barras con cantidad de préstamos por cada día de la semana

Mostrar gráfica al usuario

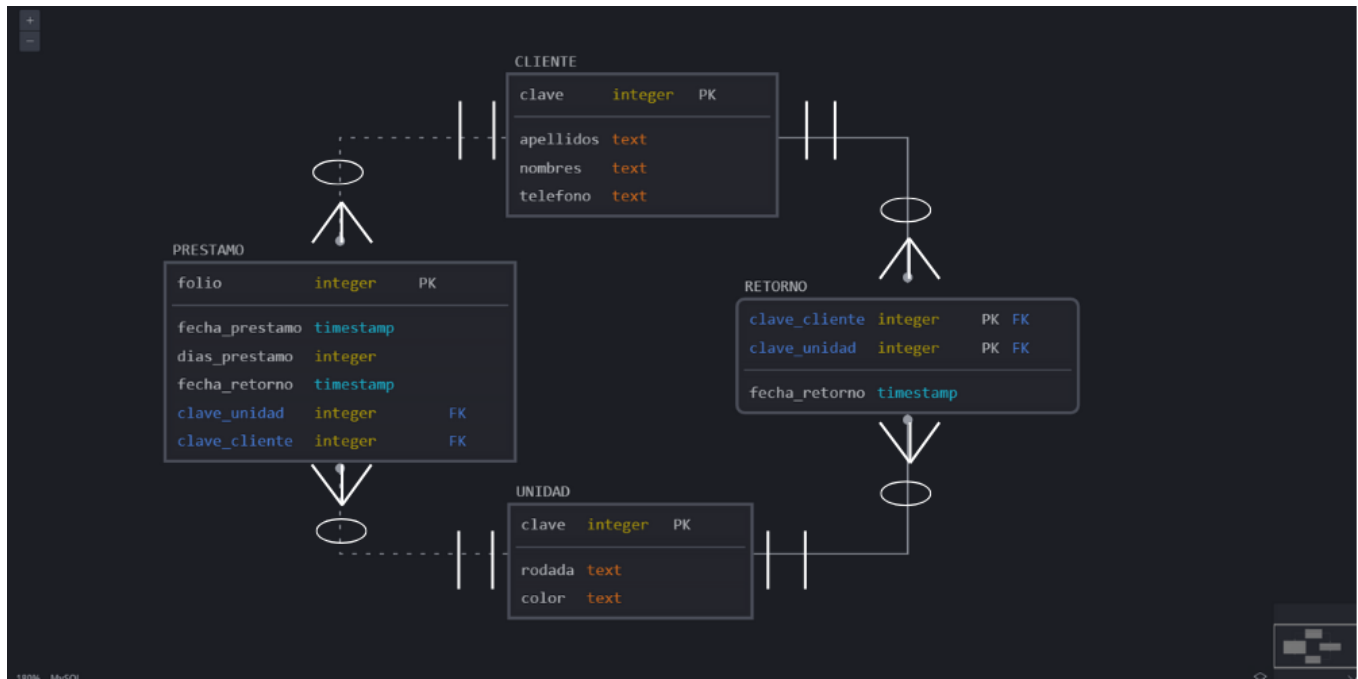
Esperar que el usuario cierre la gráfica para continuar

Caso "4" (Volver al menú de análisis):

Romper el ciclo y regresar al menú de análisis

FIN

DIAGRAMA DEL MODELO DE DATOS



LISTADO COMPLETO DEL CÓDIGO DE LA APLICACIÓN

```

import csv, sys, datetime, openpyxl, numpy as np, sqlite3, matplotlib.pyplot as plt
from scipy import stats
from openpyxl.styles import Font, Side, Border, Alignment, PatternFill
from sqlite3 import Error

border_title = Border(top = Side(border_style = "thick", color = "000000"),
                      right = Side(border_style = "thick", color = "000000"),
                      bottom = Side(border_style = "thick", color = "000000"),
                      left = Side(border_style = "thick", color = "000000"))

border_content = Border(right = Side(border_style = "thin", color = "000000"),
                       bottom = Side(border_style = "thin", color = "000000"),
                       left = Side(border_style = "thin", color = "000000"))

fill_title = PatternFill(start_color='8bdada', end_color='8bdada', fill_type= 'solid')

negritas = Font(bold=True)

alignment_center = Alignment(horizontal = "center", vertical = "center")

with sqlite3.connect('bicicletas.db', detect_types = sqlite3.PARSE_DECLTYPES |
sqlite3.PARSE_COLNAMES) as conexion:
    cursor = conexion.cursor()
    cursor.execute("PRAGMA foreign_keys = ON;")
    cursor.execute("""
        CREATE TABLE IF NOT EXISTS UNIDAD(
            clave INTEGER,
            rodada TEXT NOT NULL,
            color TEXT NOT NULL,
            CONSTRAINT PK_UNIDAD PRIMARY KEY(clave)
        );
    """)
    cursor.execute("""
        CREATE TABLE IF NOT EXISTS CLIENTE(
            clave INTEGER,
            apellidos TEXT NOT NULL,
            nombres TEXT NOT NULL,
            telefono TEXT NOT NULL,
            CONSTRAINT PK_CLIENTE PRIMARY KEY(clave)
        );
    """)
    cursor.execute("""
        CREATE TABLE IF NOT EXISTS PRESTAMO(
            folio INTEGER,
            clave_unidad INTEGER NOT NULL,
            clave_cliente INTEGER NOT NULL,
            fecha_prestamo TIMESTAMP NOT NULL,
            dias_prestamo INTEGER NOT NULL,

```

```

        fecha_retorno TIMESTAMP,
        CONSTRAINT PK_PRESTAMO PRIMARY KEY(folio),
        CONSTRAINT FK_clave_unidad_PRESTAMO FOREIGN KEY(clave_unidad)
REFERENCES UNIDAD(clave) ON DELETE CASCADE,
        CONSTRAINT FK_clave_cliente_PRESTAMO FOREIGN KEY(clave_cliente)
REFERENCES CLIENTE(clave) ON DELETE CASCADE
    );
    """)
cursor.execute("""
    CREATE TABLE IF NOT EXISTS RETORNO(
        fecha_retorno TIMESTAMP NOT NULL,
        clave_unidad INTEGER NOT NULL,
        clave_cliente INTEGER NOT NULL,
        CONSTRAINT FK_clave_unidad_RETORNO FOREIGN KEY(clave_unidad)
REFERENCES UNIDAD(clave),
        CONSTRAINT FK_clave_cliente_RETORNO FOREIGN KEY(clave_cliente)
REFERENCES CLIENTE(clave),
        CONSTRAINT PK_RETORNO PRIMARY KEY(clave_unidad, clave_cliente)
    );
    """)

def registro():
    while True:
        print("Menú de procesos > Menú de registro")
        print(f"MENÚ DE REGISTRO {'-' * 10}")
        print("1- Unidad")
        print("2- Cliente")
        print("3- Regresar")

        opcion_Submenu = input("Introduzca la clave de la entidad a registrar: ").strip()

        if not opcion_Submenu:
            print("Este campo es obligatorio. Por favor, introduzca alguna de las claves de procesos.\n")
            continue

        elif not(opcion_Submenu.isdigit()):
            print(f"La clave {opcion_Submenu} no es válida debido a que no es un valor numérico. Por favor, introduzca alguna de las claves de registro válidas.\n")
            continue

        elif not(int(opcion_Submenu) in range(1, 4)):
            print(f"La clave {opcion_Submenu} no es válida debido a que se encuentra fuera del rango de opciones. Por favor, introduzca alguna de las claves de registro válidas.\n")
            continue

        match opcion_Submenu:

            case "1":
                print("\nMenú de procesos > Menú de registro > Registro de Unidad")
                print(f"REGISTRO - UNIDAD {'-' * 10}")
                print("- Introduzca 'c' para cancelar el proceso -")

                while True:
                    rodada = input("Introduzca la rodada de la unidad (20, 26, 29): ").strip()

```

```

if (rodada.upper() == "C"): print(); break

elif not rodada:
    print("Este campo es obligatorio. Por favor, introduzca la rodada de la unidad.\n")
    continue

elif not(rodada.isdigit()):
    print(f"La rodada {rodada} no es válida debido a que no es un valor numérico. Por
favor, introduzca alguna de las rodadas válidas.\n")
    continue

elif not(rodada in ("20", "26", "29")):
    print(f"La rodada {rodada} no es válida debido a que no es una de las rodadas
registradas. Por favor, introduzca alguna de las rodadas válidas.\n")
    continue

while True:
    color = input("Introduzca el color de la unidad (máximo 14 caracteres):
").strip().title()

    if (color.upper() == "C"): print(); break

    elif not color:
        print("Este campo es obligatorio. Por favor, introduzca el color de la unidad.\n")
        continue

    elif not(color.replace(" ", "").isalpha()):
        print(f"El color de la unidad no debe contener números ni caracteres
especiales. Por favor, introduzca el color de la unidad únicamente con letras.\n")
        continue

    elif (len(color) > 15):
        print(f"El máximo de caracteres permitidos es de 14. Por favor, introduzca el
color de la unidad con una longitud de caracteres válida.\n")
        continue

    cursor.execute("INSERT INTO UNIDAD(rodada, color) VALUES(?, ?)", (rodada,
color))

    conexion.commit()

    print(f"\nREGISTRO EXITOSO - Se ha registrado una unidad con los datos:")
    print(f"\n{'='*33}")
    print(f"{'CLAVE':^8}{ 'RODADA':^10}{ 'COLOR':^15}")
    print(f"{'='*33}")
    print(f"{'cursor.lastrowid':^8}{rodada:^10}{color:^15}")
    print(f"{'='*33}\n")

    break
break

case "2":
    print("\nMenú de procesos > Menú de registro > Registro de Cliente")
    print(f"REGISTRO - CLIENTE {'-' * 10}")
    print("- Introduzca 'c' para cancelar el proceso -")

```

```

while True:

    apellidos = input("Introduzca los apellidos del cliente (máximo 40 caracteres):
).strip().title()

    if (apellidos.upper() == "C"): print(); break

    elif not apellidos:
        print("Este campo es obligatorio. Por favor, introduzca los apellidos del cliente.\n")
        continue

    elif not(apellidos.replace(" ", "").isalpha()):
        print(f"Los apellidos no deben contener números ni caracteres especiales. Por
favor, introduzca los apellidos del cliente únicamente con letras.\n")
        continue

    elif (len(apellidos) > 40):
        print(f"El máximo de caracteres permitidos es de 40. Por favor, introduzca los
apellidos del cliente con una longitud de caracteres válida.\n")
        continue

    while True:

        nombres = input("Introduzca los nombres del cliente (máximo 40 caracteres):
).strip().title()

        if (nombres.upper() == "C"): print(); break

        elif not nombres:
            print("Este campo es obligatorio. Por favor, introduzca los nombres del
cliente.\n")
            continue

        elif not(nombres.replace(" ", "").isalpha()):
            print(f"Los nombres no deben contener números ni caracteres especiales. Por
favor, introduzca los nombres del cliente únicamente con letras.\n")
            continue

        elif (len(nombres) > 40):
            print(f"El máximo de caracteres permitidos es de 40. Por favor, introduzca los
nombres del cliente con una longitud de caracteres válida.\n")
            continue

        while True:
            telefono = input("Introduzca el teléfono del cliente (10 caracteres): ").strip()

            if (telefono.upper() == "C"): print(); break

            elif not telefono:
                print("Este campo es obligatorio. Por favor, introduzca el teléfono del
cliente.\n")
                continue

            elif not(telefono.isdigit()):

```

```

        print(f"El teléfono no debe contener letras ni caracteres especiales. Por
favor, introduzca el teléfono del cliente únicamente con números.\n")
        continue

    elif not(len(telefono) == 10):
        print(f"La longitud de caracteres del teléfono es diferente a 10. Por favor,
introduzca el teléfono del cliente con una longitud de caracteres válida.\n")
        continue

    cursor.execute("INSERT INTO CLIENTE(apellidos, nombres, telefono)
VALUES (?, ?, ?)", (nombres, apellidos, telefono))
    conexion.commit()

    print(f"\nREGISTRO EXITOSO - Se ha registrado un cliente con los datos:")
    print(f"\n{'='*100}")
    print(f"{'NOMBRE COMPLETO':^80}{ 'TELÉFONO':^20}")
    print(f"{'='*100}")
    print(f"{nombres + ' ' + apellidos:^80}{telefono:^20}")
    print(f"{'='*100}\n")

    break
break
break

case "3":
    break

def prestamo():
    cursor.execute("SELECT COUNT(*) FROM UNIDAD;")
    unidades_existentes = cursor.fetchone()[0]

    cursor.execute("SELECT COUNT(*) FROM CLIENTE;")
    clientes_existentes = cursor.fetchone()[0]

    if clientes_existentes == 0:
        print(f"Actualmente no se encuentra registrado ningún cliente.")
        return

    elif unidades_existentes == 0:
        print(f"Actualmente no se encuentra registrada ninguna unidad.")
        return

    else:
        cursor.execute("SELECT COUNT(*) FROM UNIDAD u JOIN PRESTAMO p ON u.clave =
p.clave_unidad WHERE p.fecha_retorno IS NULL")
        unidades_ocupadas = cursor.fetchone()[0]

        if unidades_existentes == unidades_ocupadas:
            print(f"Actualmente no hay ninguna unidad disponible.")
            return

    print("Menú de procesos > Menú de préstamo")
    print(f"MENÚ DE PRÉSTAMO {'-' * 18}")
    print("- Introduzca 'c' para cancelar el proceso -\n")

```

```

cursor.execute("""SELECT u.clave, u.rodada, u.color
                FROM UNIDAD u
                LEFT JOIN PRESTAMO p ON u.clave = p.clave_unidad
                GROUP BY u.clave, u.rodada, u.color
                HAVING SUM(CASE WHEN p.fecha_retorno IS NULL THEN 1 ELSE 0 END) = 0
                OR COUNT(p.clave_unidad) = 0;
                """)
unidades = cursor.fetchall()

print(f"{'UNIDADES DISPONIBLES':^33}")
print("=" * 33)
print(f"{'UNIDAD':^8}{'^10' * 2}{'^15' * 2}")
print("=" * 33)
[print(f"{'unidad[0]:^8}{unidad[1]:^10}{unidad[2]:^15}" for unidad in unidades)]
print(f"{'=' * 33}\n")

while True:

    unidad = input("Introduzca la clave de la unidad: ").strip()

    cursor.execute("SELECT clave FROM UNIDAD WHERE clave = ?", (unidad,))
    unidad_encontrada = cursor.fetchone()

    cursor.execute("SELECT u.clave FROM UNIDAD u LEFT JOIN PRESTAMO p ON u.clave =
p.clave_unidad WHERE u.clave = ? GROUP BY u.clave HAVING SUM(CASE WHEN
p.fecha_retorno IS NULL THEN 1 ELSE 0 END) = 0 OR COUNT(p.clave_unidad) = 0; ", (unidad,))
    unidad_disponible = cursor.fetchone()

    if (unidad.upper() == "C"): break

    elif not unidad:
        print("Este campo es obligatorio. Por favor, introduzca la clave de la unidad.\n")
        continue

    elif not(unidad.isdigit()):
        print("La clave de la unidad no puede contener letras ni caracteres especiales. Por favor,
introduzca la clave de la unidad únicamente con números.\n")
        continue

    elif not unidad_encontrada:
        print("Esta unidad no se encuentra registrada. Por favor, introduzca la clave de alguna
unidad ya registrada.\n")
        continue

    elif not unidad_disponible:
        print("Actualmente esta unidad no se encuentra disponible. Por favor, introduzca la clave
de alguna otra unidad.\n")
        continue

    cursor.execute("SELECT * FROM CLIENTE;")
    clientes = cursor.fetchall()

    print(f"\n{'CLIENTES':^90}")
    print("=" * 90)
    print(f"{'CLAVE':^10}{'^10' * 2}{'^80' * 2}")

```

```

print("=" * 90)
[print(f"{cliente[0]:^10}{cliente[2] + ' ' + cliente[1]:^80}") for cliente in clientes]
print(f"{'=' * 90}\n")

while True:

    cliente = input("Introduzca la clave del cliente: ").strip()

    cursor.execute("SELECT clave FROM CLIENTE WHERE clave = ?", (cliente,))
    cliente_encontrado = cursor.fetchone()

    if (cliente.upper() == "C"): break

    elif not cliente:
        print("Este campo es obligatorio. Por favor, introduzca la clave del cliente.\n")
        continue

    elif not(cliente.isdigit()):
        print("La clave de la unidad no puede contener letras ni caracteres especiales. Por
favor, introduzca la clave del cliente únicamente con números.\n")
        continue

    elif not cliente_encontrado:
        print("Este cliente no se encuentra registrado. Por favor, introduzca la clave de algún
cliente ya registrado.\n")
        continue

    while True:

        fecha_Prestamo_Capturada = input("Introduzca la fecha de registro del préstamo en
formato mm-dd-aaaa (deje el campo vacío para asignar la fecha actual): ").strip()

        if (fecha_Prestamo_Capturada.upper() == "C"): break

        elif not fecha_Prestamo_Capturada:

            fecha_Prestamo_Procesada = datetime.datetime.now().date()

        else:
            try:
                fecha_Prestamo_Procesada =
datetime.datetime.strptime(fecha_Prestamo_Capturada, "%m-%d-%Y").date()

            except ValueError:
                print("La fecha introducida no cumple con el formato mm-dd-aaaa. Por favor,
introduzca una fecha con formato válido.\n")
                continue

        fecha_Actual = datetime.date.today().strftime("%m-%d-%Y")
        fecha_Actual = datetime.datetime.strptime(fecha_Actual, "%m-%d-%Y").date()

        if (fecha_Prestamo_Procesada < fecha_Actual):
            print("No se puede registrar una fecha pasada. Por favor, introduzca una fecha
válida. \n")
            continue

```

```

while True:

    cantidad_Dias = input("Introduzca la cantidad de días del préstamo: ").strip()

    if (cantidad_Dias.upper() == "C"): break

    elif not cantidad_Dias:
        print("Este campo es obligatorio. Por favor, introduzca la cantidad de días del
préstamo.\n")
        continue

    elif not(cantidad_Dias.isdigit()):
        print("La cantidad de días del préstamo no puede contener letras ni caracteres
especiales. Por favor, introduzca la cantidad de días del préstamo únicamente con números.\n")
        continue

    elif not(int(cantidad_Dias) in range(1, 15)):
        print("La cantidad mínima de días del préstamo debe ser de 1 día y máximo 14
días. Por favor, introduzca una cantidad de días del préstamo que se encuentre dentro de ese
rango.\n")
        continue

    cursor.execute("""
        INSERT INTO PRESTAMO(clave_unidad, clave_cliente, fecha_prestamo,
dias_prestamo)
        VALUES(?, ?, ?, ?)
        """, (unidad, cliente, fecha_Prestamo_Procesada, cantidad_Dias))
    conexion.commit()

    print("\nREGISTRO EXITOSO - Se ha registrado un préstamo con los datos:\n")
    print("=" * 105)
    print(f"{'FOLIO':^10}{ 'CLAVE DE UNIDAD':^20}{ 'CLAVE DE CLIENTE':^20}{ 'FECHA
DEL PRÉSTAMO':^25}{ 'CANTIDAD DE DÍAS DEL PRÉSTAMO':^30}")
    print("=" * 105)

    print(f"{'cursor.lastrowid':^10}{unidad:^20}{cliente:^20}{fecha_Prestamo_Procesada.strftime('%m-
%d-%Y'):^25}{cantidad_Dias:^30}")
    print("=" * 105)
    break
    break
    break
    break

def retorno():
    cursor.execute("SELECT COUNT(*) FROM UNIDAD")
    unidades_existentes = cursor.fetchone()[0]

    cursor.execute("SELECT COUNT(*) FROM CLIENTE")
    clientes_existentes = cursor.fetchone()[0]

    cursor.execute("SELECT COUNT(*) FROM PRESTAMO")
    prestamos_existentes = cursor.fetchone()[0]

    if unidades_existentes == 0:

```



```

        print("Actualmente no se encuentra registrada ninguna unidad. Por favor, registre alguna
unidad antes de realizar un retorno.")
        return
    elif clientes_existentes == 0:
        print("Actualmente no se encuentra registrado ningún cliente. Por favor, registre algún
cliente antes de realizar un retorno.")
        return
    elif prestamos_existentes == 0:
        print("Actualmente no se encuentra registrado ningún préstamo. Por favor, registre algún
préstamo antes de realizar un retorno.")
        return

    cursor.execute("""
        SELECT folio, clave_unidad, clave_cliente, strftime('%m-%d-%Y', fecha_prestamo) as
fecha_prestamo, dias_prestamo
        FROM PRESTAMO
        WHERE fecha_retorno IS NULL
        ORDER BY fecha_prestamo DESC;
    """)

    registros_no_retorno = cursor.fetchall()

    if not registros_no_retorno:
        print("Actualmente todas las unidades han sido retornadas.")
        return

    print(f"\n{'PRÉSTAMOS NO RETORNADOS':^105}")
    print("=" * 105)
    print(f"{'FECHA DEL PRÉSTAMO':^25}{FOLIO':^10}{CLAVE DE UNIDAD':^20}{CLAVE DE
CLIENTE':^20}{CANTIDAD DE DÍAS DEL PRÉSTAMO':^30}")
    print("=" * 105)
    for registro in registros_no_retorno:
        folio, clave_unidad, clave_cliente, fecha_prestamo, dias_prestamo = registro

    print(f"{'fecha_prestamo':^25}{folio:^10}{clave_unidad:^20}{clave_cliente:^20}{dias_prestamo:^30}")
    print("=" * 105)

    while True:
        folio = input("Introduzca el folio del préstamo a retornar: ").strip()

        if folio.upper() == "C":
            break

        elif not folio:
            print("Este campo es obligatorio. Por favor, introduzca la cantidad de días del
préstamo.\n")
            continue

        elif not(folio.isdigit()):
            print("El folio del préstamo no puede contener letras ni caracteres especiales. Por
favor, introduzca el folio del préstamo únicamente con números.\n")
            continue

        cursor.execute("SELECT folio, strftime('%m-%d-%Y', fecha_prestamo) as fecha_prestamo
FROM PRESTAMO WHERE folio = ? AND fecha_retorno IS NULL", (folio,))

```

```

prestamo_no_retorno = cursor.fetchone()

if not prestamo_no_retorno:
    print("No se encontró el préstamo con el folio especificado o ya ha sido retornado.")
    continue

folio, fecha_prestamo = prestamo_no_retorno
fecha_prestamo = datetime.datetime.strptime(fecha_prestamo, "%m-%d-%Y").date()

while True:
    fecha_Retorno_Capturada = input("Introduzca la fecha del retorno en formato mm-dd-aaaa (deje en blanco para usar la fecha actual): ").strip()

    if fecha_Retorno_Capturada.upper() == "C":
        break

    if not fecha_Retorno_Capturada:
        fecha_retorno = datetime.datetime.now().date()
    else:
        try:
            fecha_retorno = datetime.datetime.strptime(fecha_Retorno_Capturada, "%m-%d-%Y").date()
        except ValueError:
            print("Formato incorrecto. Use mm-dd-aaaa.")
            continue

    if fecha_retorno < fecha_prestamo:
        print("La fecha de retorno no puede ser anterior a la fecha del préstamo.")
        continue

    cursor.execute("UPDATE PRESTAMO SET fecha_retorno = ? WHERE folio = ?",
(fecha_retorno, folio))

    cursor.execute("SELECT clave_unidad, clave_cliente FROM PRESTAMO WHERE folio =
?", (folio,))
    resultado = cursor.fetchone()

    if resultado:
        clave_unidad, clave_cliente = resultado

        cursor.execute("INSERT INTO RETORNO (fecha_retorno, clave_unidad,
clave_cliente) VALUES (?, ?, ?)",
(fecha_retorno, clave_unidad, clave_cliente))

        conexion.commit()
        print(f"\nREGISTRO EXITOSO - Retorno registrado para el préstamo con folio {folio}.")
        break
    break

def informe():
    cursor.execute("SELECT COUNT(*) FROM UNIDAD")
    unidades_existentes = cursor.fetchone()[0]

    cursor.execute("SELECT COUNT(*) FROM CLIENTE")
    clientes_existentes = cursor.fetchone()[0]

```

```

if (not unidades_existentes) and (not clientes_existentes):
    print(f"Actualmente no se cuenta con los registros necesarios para generar informes. Por favor, registre datos antes de solicitar informes.")
    return

while True:
    print("Menú de procesos > Menú de informes")
    print(f"MENÚ DE INFORMES {'-' * 10}")
    print("1- Reportes.")
    print("2- Análisis.")
    print("3- Regresar.")

    opcion_Submenu = input("Introduzca la clave la operación a realizar: ").strip()

    if not(opcion_Submenu.isdigit()):
        print(f"La clave {opcion_Submenu} no es válida debido a que no es un valor numérico. Por favor, introduzca alguna de las claves de procesos válidas.\n")
        continue

    elif not(int(opcion_Submenu) in range(1, 4)):
        print(f"La clave {opcion_Submenu} no es válida debido a que se encuentra fuera del rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
        continue

    elif not opcion_Submenu:
        print("Este campo es obligatorio. Por favor, introduzca alguna de las claves de procesos.\n")
        continue

    match opcion_Submenu:

        case "1":
            print()
            reporte()

        case "2":
            print()
            analisis()

        case "3":
            break

def reporte():

    while True:
        print("Menú de procesos > Menú de reportes")
        print(f"MENÚ DE REPORTES {'-' * 10}")
        print("1- Clientes.")
        print("2- Listado de Unidades.")
        print("3- Retrasos.")
        print("4- Préstamos por retornar.")
        print("5- Préstamos por periodo.")
        print("6- Regresar.")

```

```

opcion_Submenu = input("Introduzca la clave la operación a realizar: ").strip()

if not opcion_Submenu:
    print("Este campo es obligatorio. Por favor, introduzca alguna de las claves de
procesos.\n")
    continue

elif not(opcion_Submenu.isdigit()):
    print(f"La clave {opcion_Submenu} no es válida debido a que no es un valor numérico.
Por favor, introduzca alguna de las claves de procesos válidas.\n")
    continue

elif not(int(opcion_Submenu) in range(1, 7)):
    print(f"La clave {opcion_Submenu} no es válida debido a que se encuentra fuera del
rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
    continue

match opcion_Submenu:
    case "1":
        while True:
            print("\nMenú de procesos > Menú de informes > Menú de reportes > Reporte de
Clientes")
            print("-Menú de Clientes")
            print("1- Reporte Completo de clientes.")
            print("2- Cliente especifico.")
            print("3- Regresar")

            opcion_submenu_cliente = input("Introduzca la clave de la operación a realizar:
").strip()

            if not opcion_submenu_cliente:
                print("Este campo es obligatorio. Por favor, introduzca alguna de las claves de
procesos.\n")
                continue

            elif not(opcion_submenu_cliente.isdigit()):
                print(f"La clave{opcion_submenu_cliente} no es valida ya que no es un valor
numerico. Por favor introduzca alguna de las claves de procesos válidas.\n")
                continue

            elif not (int(opcion_submenu_cliente) in range(1,4)):
                print(f"La clave {opcion_submenu_cliente} no es válida debido a que se encuentra
fuera del rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
                continue

            match opcion_submenu_cliente:
                case "1":
                    print("\nMenú de procesos > Menú de informes > Menú de reportes > Reporte
de Clientes > Reporte Completo de clientes")
                    print("-Menú de Reportes de Cliente")
                    print(f"\n{'CLIENTES':^120}")
                    print("=" * 120)
                    print(f"{'CLAVE':^10}{ 'NOMBRE':^45}{ 'APELLIDOS':^45}{ 'TELÉFONO':^20}")

```

```

print("=" * 120)

cursor.execute("SELECT * FROM CLIENTE")
registros_cliente = cursor.fetchall()

[print(f"{clave:^10}{apellidos:^45}{nombre:^45}{tel:^20}") for clave, apellidos,
nombre, tel in registros_cliente]
print(f"{'=' * 120}\n")

while True:

    print("1- Guardar archivo Excel.")
    print("2- Guardar archivo CSV.")
    print("3- Regresar.")

    opcion = input("Introduzca la clave del proceso a realizar: ").strip()

    if not opcion:
        print("Este campo es obligatorio. Por favor, introduzca alguna de las
claves de procesos.\n")
        continue

    elif not(opcion.isdigit()):
        print(f"La clave {opcion} no es válida debido a que no es un valor
numérico. Por favor, introduzca alguna de las claves de procesos válidas.\n")
        continue

    elif not(int(opcion) in range(1, 4)):
        print(f"La clave {opcion} no es válida debido a que se encuentra fuera del
rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
        continue

    match opcion:

        case "1":
            libro = openpyxl.Workbook()
            hoja = libro["Sheet"]
            hoja.title = "Reporte de Clientes"

            hoja["A1"].value = "CLAVE"
            hoja["B1"].value = "APELLIDOS"
            hoja["C1"].value = "NOMBRES"
            hoja["D1"].value = "TELÉFONO"

            hoja.column_dimensions["A"].width = 8
            hoja.column_dimensions["B"].width = 40
            hoja.column_dimensions["C"].width = 40
            hoja.column_dimensions["D"].width = 15

            for head in range(1, 5):
                hoja.cell(row = 1, column = head).border = border_title
                hoja.cell(row = 1, column = head).alignment = alignment_center
                hoja.cell(row = 1, column = head).fill = fill_title
                hoja.cell(row = 1, column = head).font = negritas

```

```

i = 2
for clave, apellidos, nombre, tel in registros_cliente:

    for j, elemento in enumerate((clave, apellidos, nombre, tel), start=1):
        hoja.cell(row=i, column=j).value = elemento
        hoja.cell(row=i, column=j).border = border_content
        hoja.cell(row=i, column=j).alignment = alignment_center
    i += 1

    libro.save("Reporte_Clientes.xlsx")
    print("\nEXPORTACIÓN EXITOSA - El archivo Reporte_Clientes.xlsx
se ha generado exitosamente.\n")
    break

case "2":
    with open("Reporte_Clientes.csv", "w", encoding = "latin1", newline="")
as archivo:

        escritor = csv.writer(archivo)
        escritor.writerow(("CLAVE", "APELLIDOS", "NOMBRES",
"TELÉFONO"))
        escritor.writerows([(clave, apellidos, nombre, tel) for clave, apellidos,
nombre, tel in registros_cliente])

        print("\nEXPORTACIÓN EXITOSA - El archivo Reporte_Clientes.csv
se ha generado exitosamente.\n")
        break

case "3":
    print()
    break

case "2":
    while True:
        print("\nMenú de procesos > Menú de informes > Menú de reportes >
Reporte de Clientes > Reporte Cliente por clave")
        print("-Menú de Reportes de Cliente")
        print("- Introduzca 'c' para cancelar el proceso -")
        print(f"\n{'CLIENTES':^120}")
        print("=" * 120)

        print(f"{'CLAVE':^10}{{'NOMBRES':^45}{{'APELLIDOS':^45}{{'TELÉFONO':^20}}}")
        print("=" * 120)

        cursor.execute("SELECT * FROM CLIENTE")
        registros_cliente = cursor.fetchall()

        [print(f"{'clave':^10}{{'nombre':^45}{{'apellidos':^45}{{'tel':^20}}}") for clave, nombre,
apellidos, tel in registros_cliente]
        print(f"{'=' * 120}\n")

        seleccion_cliente = input("Introduzca la clave del cliente específico:").strip()

        if seleccion_cliente.upper() == "C":
            break

```

```

        if not(seleccion_cliente.isdigit()):
            print("La clave del cliente no puede contener Letras ni caracteres
especiales")
            continue

        cursor.execute("SELECT * FROM CLIENTE WHERE clave = ?",
(seleccion_cliente,))
        cliente_info = cursor.fetchall()

        if cliente_info:
            [print(f"\nDetalles del Cliente: \nNombres: {nombre} \nApellidos: {apellidos}
\nTeléfono: {telefono}") for clave, nombre, apellidos, telefono in cliente_info]
            print(f"\n{'HISTORIAL DE PRÉSTAMOS Y RETORNOS':^150}")
            print("=" * 150)
            print(f"{'FOLIO PRESTAMO':^20}{'CLAVE UNIDAD':^20}{'FECHA
PRESTAMO':^25}{'DIAS PRESTAMO':^20}{'FECHA RETORNO':^25}{'FOLIO RETORNO':^20}")
            print("=" * 150)

            cursor.execute("""SELECT p.folio, p.clave_unidad, strftime('%m-%d-%Y',
p.fecha_prestamo), p.dias_prestamo, strftime('%m-%d-%Y', r.fecha_retorno), r.clave_unidad
FROM PRESTAMO p
LEFT JOIN RETORNO r ON p.clave_unidad = r.clave_unidad AND
p.clave_cliente = r.clave_cliente
WHERE p.clave_cliente = ?
ORDER BY p.fecha_prestamo ASC;""", (seleccion_cliente,))
            historial = cursor.fetchall()

            for folio_prestamo, clave_unidad, fecha_prestamo, dias_prestamo,
fecha_retorno, folio_retorno in historial:
                print(f"{'folio_prestamo':^20}{clave_unidad:^20}{fecha_prestamo:^25}{dias_prestamo:^20}{fecha_retorno or 'No Retornado':^25}{folio_retorno or 'N/A':^20}")
                print("=" * 150)

            while True:
                print("1- Guardar archivo Excel.")
                print("2- Guardar archivo CSV.")
                print("3- Regresar.")

                opcion = input("Introduzca la clave del proceso a realizar: ").strip()

                if not opcion:
                    print("Este campo es obligatorio. Por favor, introduzca alguna de las
claves de procesos.\n")
                    continue

                elif not(opcion.isdigit()):
                    print(f"La clave {opcion} no es válida debido a que no es un valor
numérico. Por favor, introduzca alguna de las claves de procesos válidas.\n")
                    continue

                elif not(int(opcion) in range(1, 4)):

```

```

        print(f"La clave {opcion} no es válida debido a que se encuentra fuera
del rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
        continue

```

```

if opcion.upper() == "C":
    break

```

```

match opcion:

```

```

    case "1":

```

```

        libro = openpyxl.Workbook()
        hoja = libro["Sheet"]
        hoja.title = "Historial por cliente"

```

```

        hoja["A1"].value = "FOLIO PRESTAMO"
        hoja["B1"].value = "CLAVE UNIDAD"
        hoja["C1"].value = "FECHA PRESTAMO"
        hoja["D1"].value = "DIAS PRESTAMO"
        hoja["E1"].value = "FECHA RETORNO"
        hoja["F1"].value = "FOLIO RETORNO"

```

```

        hoja.column_dimensions["A"].width = 20
        hoja.column_dimensions["B"].width = 15
        hoja.column_dimensions["C"].width = 20
        hoja.column_dimensions["D"].width = 15
        hoja.column_dimensions["E"].width = 20
        hoja.column_dimensions["F"].width = 15
        hoja.column_dimensions["A"].width = 8

```

```

        for head in range(1, 7):

```

```

            hoja.cell(row = 1, column = head).border = border_title
            hoja.cell(row = 1, column = head).alignment =

```

alignment_center

```

            hoja.cell(row = 1, column = head).fill = fill_title
            hoja.cell(row = 1, column = head).font = negritas

```

```

        i = 2

```

```

        for folio_prestamo, clave_unidad, fecha_prestamo,

```

dias_prestamo, fecha_retorno, folio_retorno in historial:

```

            hoja.cell(row=i, column=1).value = folio_prestamo
            hoja.cell(row=i, column=2).value = clave_unidad
            hoja.cell(row=i, column=3).value = fecha_prestamo
            hoja.cell(row=i, column=4).value = dias_prestamo
            hoja.cell(row=i, column=5).value = fecha_retorno or "No

```

Retornado"

```

            hoja.cell(row=i, column=6).value = folio_retorno or "N/A"
            hoja.cell(row=i, column=7).border = border_content
            hoja.cell(row=i, column=8).alignment = alignment_center
            i += 1

```

```

        nombre_excel = f"Historial_{seleccion_cliente}.xlsx"

```

```

        libro.save(nombre_excel)

```

```

        print(f"\nEXPORTACIÓN EXITOSA - El archivo {nombre_excel}

```

se ha generado exitosamente.\n")


```

        case "2":
            nombre_csv = f"Historial_{seleccion_cliente}.csv"
            with open(nombre_csv, "w", encoding="latin1", newline="") as
archivo:
                escritor = csv.writer(archivo)
                escritor.writerow(("FOLIO PRESTAMO", "CLAVE UNIDAD",
"FECHA PRESTAMO", "DIAS PRESTAMO", "FECHA RETORNO", "FOLIO RETORNO"))
                escritor.writerows([(folio_prestamo, clave_unidad,
fecha_prestamo, dias_prestamo, fecha_retorno or "No Retornado", folio_retorno or "N/A") for
folio_prestamo, clave_unidad, fecha_prestamo, dias_prestamo, fecha_retorno, folio_retorno in
historial])

                print(f"\nEXPORTACIÓN EXITOSA - El archivo
Historial_{seleccion_cliente}.csv se ha generado exitosamente.\n")
                break

        case "3":
            print()
            break
        break
    break
else:
    print("No se encontró ningún cliente con la clave proporcionada.\n")
    continue

case "3" :
    print(); break

case "4":

    cursor.execute("""
        SELECT COUNT(*)
        FROM PRESTAMO
        WHERE fecha_retorno IS NULL;
    """)
    prestamos_xretornar = cursor.fetchall()

    cursor.execute("SELECT folio FROM PRESTAMO")
    prestamos_existentes = cursor.fetchone()

    if prestamos_existentes is None:
        print(f"\nAún no se ha realizado ningún préstamo.\n")
        continue

    elif prestamos_xretornar == 0:
        print(f"\nActualmente todas las unidades han sido retornadas.\n")
        continue

    print("\nMenú de procesos > Menú de informes > Menú de reportes > Reporte de
Préstamos por retornar")
    print(f'REPORTE DE PRÉSTAMOS POR RETORNAR{'-' * 10}')
    print("- Introduzca 'c' para cancelar el proceso -")

    while True:

```

```

        fecha_inicial = input("Introduzca la fecha inicial con la que se filtrarán los datos en
formato mm-dd-yyyy (deje el campo vacío para asignar la fecha actual): ").strip()

        if (fecha_inicial.upper() == "C"): break

        elif not fecha_inicial:
            fecha_inicial_procesada = datetime.date.today().strftime("%m-%d-%Y")
            fecha_inicial_procesada = datetime.datetime.strptime(fecha_inicial_procesada,
"%m-%d-%Y").date()

        else:
            try:
                fecha_inicial_procesada = datetime.datetime.strptime(fecha_inicial, "%m-%d-
%Y").date()

            except ValueError:
                print("La fecha introducida no cumple con el formato mm-dd-aaaa. Por favor,
introduzca una fecha con formato válido.\n")
                continue

        while True:
            fecha_final = input("Introduzca la fecha final con la que se filtrarán los datos en
formato mm-dd-yyyy (deje el campo vacío para asignar la fecha actual): ").strip()

            if (fecha_final.upper() == "C"): break

            elif not fecha_final:
                fecha_final_procesada = datetime.date.today().strftime("%m-%d-%Y")
                fecha_final_procesada = datetime.datetime.strptime(fecha_final_procesada,
"%m-%d-%Y").date()

            else:
                try:
                    fecha_final_procesada = datetime.datetime.strptime(fecha_final, "%m-%d-
%Y").date()

                except ValueError:
                    print("La fecha introducida no cumple con el formato mm-dd-aaaa. Por favor,
introduzca una fecha con formato válido.\n")
                    continue

            if (fecha_final_procesada < fecha_inicial_procesada):
                print("La fecha final introducida no puede ser inferior a la fecha inicial. Por
favor, introduzca una fecha final válida.\n")
                continue

        cursor.execute("""
            SELECT p.folio, p.clave_unidad, u.rodada, strftime('%m-%d-%Y',
p.fecha_prestamo), (c.nombres || ' ' || c.apellidos) AS nombre_completo, c.telefono
            FROM PRESTAMO p LEFT JOIN UNIDAD u ON p.clave_unidad = u.clave
            LEFT JOIN CLIENTE c ON p.clave_cliente = c.clave
            WHERE fecha_retorno IS NULL AND (p.fecha_prestamo >= ? AND
p.fecha_prestamo <= ?);""", (fecha_inicial_procesada, fecha_final_procesada))
        prestamos = cursor.fetchall()

```

```

        if prestamos:
            print(f"\n{'PRESTAMOS POR RETORNAR':^175}")
            print("=" * 175)
            print(f"{'FOLIO':^10}{{'CLAVE DE UNIDAD':^20}{{'RODADA':^20}{{'FECHA DEL
PRÉSTAMO':^25}{{'NOMBRE COMPLETO':^80}{{'TELEFONO':^20}}}}")
            print("=" * 175)

[print(f"{'folio':^10}{clave_unidad:^20}{rodada:^20}{fecha_prestamo:^25}{nombre_completo:^80}{telef
ono:^20}") for folio, clave_unidad, rodada, fecha_prestamo, nombre_completo, telefono in
prestamos]

            print(f"{'=' * 175}\n")

        else:
            print("\nNo se cuenta con préstamos por retornar dentro de las fechas
indicadas.\n")
            break

    while True:

        print("1- Guardar archivo Excel.")
        print("2- Guardar archivo CSV.")
        print("3- Regresar.")

        opcion = input("Introduzca la clave del proceso a realizar: ")

        if not opcion:
            print("Este campo es obligatorio. Por favor, introduzca alguna de las claves
de procesos.\n")
            continue

        elif not(opcion.isdigit()):
            print(f"La clave {opcion} no es válida debido a que no es un valor numérico.
Por favor, introduzca alguna de las claves de procesos válidas.\n")
            continue

        elif not(int(opcion) in range(1, 4)):
            print(f"La clave {opcion} no es válida debido a que se encuentra fuera del
rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
            continue

        match opcion:

            case "1":
                libro = openpyxl.Workbook()
                hoja = libro["Sheet"]
                hoja.title = "Prestamos por retornar"

                hoja["A1"].value = "FOLIO"
                hoja["B1"].value = "CLAVE DE UNIDAD"
                hoja["C1"].value = "RODADA"
                hoja["D1"].value = "FECHA DEL PRÉSTAMO"
                hoja["E1"].value = "NOMBRE COMPLETO"
                hoja["F1"].value = "TELEFONO"

```

```

hoja.column_dimensions["A"].width = 10
hoja.column_dimensions["B"].width = 20
hoja.column_dimensions["C"].width = 20
hoja.column_dimensions["D"].width = 25
hoja.column_dimensions["E"].width = 80
hoja.column_dimensions["F"].width = 20

for head in range(1, 7):
    hoja.cell(row = 1, column = head).border = border_title
    hoja.cell(row = 1, column = head).alignment = alignment_center
    hoja.cell(row = 1, column = head).fill = fill_title
    hoja.cell(row = 1, column = head).font = negritas

i = 2
for folio, clave_unidad, rodada, fecha_prestamo, nombre_completo,
telefono in prestamos:
    for j, elemento in enumerate((folio, clave_unidad, rodada,
fecha_prestamo, nombre_completo, telefono), start = 1):
        hoja.cell(row = i, column = j).value = elemento
        hoja.cell(row = i, column = j).border = border_content
        hoja.cell(row = i, column = j).alignment = alignment_center
        i += 1

    libro.save("Prestamos_por_Retornar.xlsx")
    print("\nEXPORTACIÓN EXITOSA - El archivo
Prestamos_por_Retornar.xlsx se ha generado exitosamente.\n")
    break

case "2":
    with open("Prestamos_por_Retornar.csv", "w", encoding = "latin1",
newline="") as archivo:
        escritor = csv.writer(archivo)
        escritor.writerow(("FOLIO", "CLAVE DE UNIDAD", "RODADA", "FECHA
DEL PRÉSTAMO", "NOMBRE COMPLETO", "TELÉFONO"))

        for folio, clave_unidad, rodada, fecha_prestamo, nombre_completo,
telefono in prestamos:
            escritor.writerow([folio, clave_unidad, rodada, fecha_prestamo,
nombre_completo, telefono])

        print("\nEXPORTACIÓN EXITOSA - El archivo
Prestamos_por_Retornar.csv se ha generado exitosamente.\n")
        break

case "3":
    print()
    break
break
break
break

case "5":

    print("\nMenú de procesos > Menú de informes > Menú de reportes > Reporte de
Préstamos por periodo")

```

```

print(f"REPORTE DE PRÉSTAMOS POR PERIODO{'-' * 10}")
print("- Introduzca 'c' para cancelar el proceso -")

while True:

    fecha_inicial = input("Introduzca la fecha inicial con la que se filtrarán los datos en
formato mm-dd-yyyy (deje el campo vacío para asignar la fecha actual): ").strip()

    if (fecha_inicial.upper() == "C"): print(); break

    elif not fecha_inicial:
        fecha_inicial_procesada = datetime.date.today().strftime("%m-%d-%Y")
        fecha_inicial_procesada = datetime.datetime.strptime(fecha_inicial_procesada,
"%m-%d-%Y").date()

    else:
        try:
            fecha_inicial_procesada = datetime.datetime.strptime(fecha_inicial, "%m-%d-
%Y").date()

        except ValueError:
            print("La fecha introducida no cumple con el formato mm-dd-aaaa. Por favor,
introduzca una fecha con formato válido.\n")
            continue

    while True:

        fecha_final = input("Introduzca la fecha final con la que se filtrarán los datos en
formato mm-dd-yyyy (deje el campo vacío para asignar la fecha actual): ").strip()

        if (fecha_final.upper() == "C"): print(); break

        elif not fecha_final:
            fecha_final_procesada = datetime.date.today().strftime("%m-%d-%Y")
            fecha_final_procesada = datetime.datetime.strptime(fecha_final_procesada,
"%m-%d-%Y").date()

        else:
            try:
                fecha_final_procesada = datetime.datetime.strptime(fecha_final, "%m-%d-
%Y").date()

            except ValueError:
                print("La fecha introducida no cumple con el formato mm-dd-aaaa. Por favor,
introduzca una fecha con formato válido.\n")
                continue

        if (fecha_final_procesada < fecha_inicial_procesada):
            print("La fecha final introducida no puede ser inferior a la fecha inicial. Por
favor, introduzca una fecha final válida.\n")
            continue

        cursor.execute("""
            SELECT p.folio, p.clave_unidad, u.rodada, strftime('%m-%d-%Y',
p.fecha_prestamo), (c.nombres || ' ' || c.apellidos) AS nombre_completo, c.telefono
            FROM PRESTAMO p LEFT JOIN UNIDAD u ON p.clave_unidad = u.clave

```

```

LEFT JOIN CLIENTE c ON p.clave_cliente = c.clave
WHERE p.fecha_prestamo >= ? AND p.fecha_prestamo <= ?;',
(fecha_inicial_procesada, fecha_final_procesada))
prestamos = cursor.fetchall()

if prestamos:
    print(f"\n{'PRESTAMOS POR PERIODO':^175}")
    print("=" * 175)
    print(f"{'FOLIO':^10}{ 'CLAVE DE UNIDAD':^20}{ 'RODADA':^20}{ 'FECHA DEL
PRÉSTAMO':^25}{ 'NOMBRE COMPLETO':^80}{ 'TELEFONO':^20}")
    print("=" * 175)

[print(f"{'folio':^10}{ 'clave_unidad':^20}{ 'rodada':^20}{ 'fecha_prestamo':^25}{ 'nombre_completo':^80}{ 'telefono':^20}") for folio, clave_unidad, rodada, fecha_prestamo, nombre_completo, telefono in
prestamos]
    print(f"{'=' * 175}\n")

else:
    print("\nNo se cuenta con préstamos registrados dentro de las fechas
indicadas.\n")
    break

while True:
    print("1- Guardar archivo Excel.")
    print("2- Guardar archivo CSV.")
    print("3- Regresar.")

    opcion = input("Introduzca la clave del proceso a realizar: ").strip()

    if not opcion:
        print("Este campo es obligatorio. Por favor, introduzca alguna de las claves
de procesos.\n")
        continue

    elif not(opcion.isdigit()):
        print(f"La clave {opcion} no es válida debido a que no es un valor numérico.
Por favor, introduzca alguna de las claves de procesos válidas.\n")
        continue

    elif not(int(opcion) in range(1, 4)):
        print(f"La clave {opcion} no es válida debido a que se encuentra fuera del
rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
        continue

    match opcion:
        case "1":
            libro = openpyxl.Workbook()
            hoja = libro["Sheet"]
            hoja.title = "Prestamos por periodo"

            hoja["A1"].value = "FOLIO"
            hoja["B1"].value = "CLAVE DE UNIDAD"
            hoja["C1"].value = "RODADA"
            hoja["D1"].value = "FECHA DEL PRÉSTAMO"

```

```

hoja["E1"].value = "NOMBRE COMPLETO"
hoja["F1"].value = "TELEFONO"

hoja.column_dimensions["A"].width = 10
hoja.column_dimensions["B"].width = 20
hoja.column_dimensions["C"].width = 20
hoja.column_dimensions["D"].width = 25
hoja.column_dimensions["E"].width = 80
hoja.column_dimensions["F"].width = 20

for head in range (1, 7):
    hoja.cell(row = 1, column = head).border = border_title
    hoja.cell(row = 1, column = head).alignment = alignment_center
    hoja.cell(row = 1, column = head).fill = fill_title
    hoja.cell(row = 1, column = head).font = negritas

i = 2
for folio, clave_unidad, rodada, fecha_prestamo, nombre_completo,
telefono in prestamos:
    for j, elemento in enumerate((folio, clave_unidad, rodada,
fecha_prestamo, nombre_completo, telefono), start = 1):
        hoja.cell(row = i, column = j).value = elemento
        hoja.cell(row = i, column = j).border = border_content
        hoja.cell(row = i, column = j).alignment = alignment_center
        i += 1

    libro.save("Prestamos_por_Periodo.xlsx")
    print("\nEXPORTACIÓN EXITOSA - El archivo
Prestamos_por_Periodo.xlsx se ha generado exitosamente.\n")
    break

case "2":
    with open("Prestamos_por_Periodo.csv", "w", encoding = "latin1",
newline="") as archivo:
        escritor = csv.writer(archivo)
        escritor.writerow(("FOLIO", "CLAVE DE UNIDAD", "RODADA", "FECHA
DEL PRÉSTAMO", "NOMBRE COMPLETO", "TELEFONO"))

        for folio, clave_unidad, rodada, fecha_prestamo, nombre_completo,
telefono in prestamos:
            escritor.writerow([folio, clave_unidad, rodada, fecha_prestamo,
nombre_completo, telefono])

        print("\nEXPORTACIÓN EXITOSA - El archivo
Prestamos_por_Periodo.csv se ha generado exitosamente.\n")
        break

case "3":
    print()
    break
break
break

case "2":
    while True:

```

```

unidades")    print("\nMenú de procesos > Menú de informes > Menú de reportes > Listado de
               print(f"LISTADO DE UNIDADES {'-' * 10}")
               print("1- Completo.")
               print("2- Por rodada.")
               print("3- Por color.")
               print("4- Regresar.")

               opcion = input("Introduzca la clave del proceso a realizar: ").strip()

               if not opcion:
procesos.\n")    print("Este campo es obligatorio. Por favor, introduzca alguna de las claves de
                   continue

               elif not(opcion.isdigit()):
favor, introduzca alguna de las claves de procesos válidas.\n")
                   print(f"La clave {opcion} no es válida debido a que no es un valor numérico. Por
                   continue

               elif not(int(opcion) in range(1, 5)):
de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
                   print(f"La clave {opcion} no es válida debido a que se encuentra fuera del rango
                   continue

               match opcion:
                   case "1":
de unidades > Completo\n")
                       print("\nMenú de procesos > Menú de informes > Menú de reportes > Listado
                       print(f"UNIDADES REGISTRADAS: ^33}")
                       print("=" * 33)
                       print(f"CLAVE: ^8}{RODADA: ^10}{COLOR: ^15}")
                       print("=" * 33)

                       cursor.execute("SELECT * FROM UNIDAD")
                       registros_unidad = cursor.fetchall()

registros_unidad] [print(f"{clave: ^8}{rodada: ^10}{color: ^15}") for clave, rodada, color in
                    print(f"{'=' * 33}\n")

                    while True:
                        print("1- Guardar archivo Excel.")
                        print("2- Guardar archivo CSV.")
                        print("3- Regresar.")

                        opcion = input("Introduzca la clave del proceso a realizar: ").strip()

                        if not opcion:
claves de procesos.\n")
                            print("Este campo es obligatorio. Por favor, introduzca alguna de las
                            continue

                        elif not(opcion.isdigit()):

```



```

        print(f"La clave {opcion} no es válida debido a que no es un valor
numérico. Por favor, introduzca alguna de las claves de procesos válidas.\n")
        continue

    elif not(int(opcion) in range(1, 4)):
        print(f"La clave {opcion} no es válida debido a que se encuentra fuera del
rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
        continue

    match opcion:

        case "1":
            libro = openpyxl.Workbook()
            hoja = libro["Sheet"]
            hoja.title = "Completo"

            hoja["A1"].value = "CLAVE"
            hoja["B1"].value = "RODADA"
            hoja["C1"].value = "COLOR"

            hoja.column_dimensions["A"].width = 10
            hoja.column_dimensions["B"].width = 15
            hoja.column_dimensions["C"].width = 20

            for head in range(1, 4):
                hoja.cell(row = 1, column = head).border = border_title
                hoja.cell(row = 1, column = head).alignment = alignment_center
                hoja.cell(row = 1, column = head).fill = fill_title
                hoja.cell(row = 1, column = head).font = negritas

            i = 2
            for clave, rodada, color in registros_unidad:
                tupla_unidades = clave, rodada, color
                j = 1
                for dato in tupla_unidades:
                    hoja.cell(row = i, column = j).value = dato
                    hoja.cell(row = i, column = j).border = border_content
                    hoja.cell(row = i, column = j).alignment = alignment_center
                    j += 1
                i += 1

            libro.save("Reporte_Unidades_Completo.xlsx")
            print("\nEXPORTACIÓN EXITOSA - El archivo
Reporte_Unidades_Completo.xlsx se ha generado exitosamente.")
            break

        case "2":
            with open("Reporte_Unidades_Completo.csv", "w", encoding = "latin1",
newline="") as archivo:

                escritor = csv.writer(archivo)
                escritor.writerow(("CLAVE", "RODADA", "COLOR"))

                for clave, rodada, color in registros_unidad:
                    escritor.writerow([clave, rodada, color])

```

```

        print("\nEXPORTACIÓN EXITOSA - El archivo
Reporte_Unidades_Completo.csv se ha generado exitosamente.")
        break

    case "3":
        break

    case "2":
        print("\nMenú de procesos > Menú de informes > Menú de reportes > Listado
de unidades > Rodada")
        print("- Introduzca 'c' para cancelar el proceso -")

        while True:
            rodada_elegida = input("Introduzca la rodada con la cual se filtrarán los
datos (20, 26, 29): ").strip()

            if (rodada_elegida.upper() == "C"):
                break

            elif not rodada_elegida:
                print("Este campo es obligatorio. Por favor, introduzca la rodada de la
unidad.\n")
                continue

            elif not(rodada_elegida.isdigit()):
                print(f"La rodada {rodada_elegida} no es válida debido a que no es un
valor numérico. Por favor, introduzca alguna de las rodadas válidas.\n")
                continue

            elif not(rodada_elegida in ("20", "26", "29")):
                print(f"La rodada {rodada_elegida} no es válida debido a que no es una de
las rodadas registradas. Por favor, introduzca alguna de las rodadas válidas.\n")
                continue

            cursor.execute("SELECT clave, color FROM UNIDAD WHERE rodada = ?",
(rodada_elegida,))
            registros_rodada = cursor.fetchall()

            if registros_rodada:
                print(f"\n{f'UNIDADES REGISTRADAS DE RODADA
{rodada_elegida}':^34}")
                print("=" * 34)
                print(f"{'CLAVE':^17}{'COLOR':^17}")
                print("=" * 34)

                [print(f"{'clave':^17}{'color':^17}") for clave, color in registros_rodada]
                print(f"{'=' * 34}\n")

            else:
                print(f"Por el momento no hay unidades con rodada {rodada_elegida}
registradas. Por favor, introduzca alguna otra de las rodadas válidas.\n")
                continue

        while True:
            print("1- Guardar archivo Excel.")

```

```

print("2- Guardar archivo CSV.")
print("3- Regresar.")

opcion = input("Introduzca la clave del proceso a realizar: ").strip()

if not opcion:
    print("Este campo es obligatorio. Por favor, introduzca alguna de las
claves de procesos.\n")
    continue

elif not(opcion.isdigit()):
    print(f"La clave {opcion} no es válida debido a que no es un valor
numérico. Por favor, introduzca alguna de las claves de procesos válidas.\n")
    continue

elif not(int(opcion) in range(1, 4)):
    print(f"La clave {opcion} no es válida debido a que se encuentra fuera
del rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
    continue

match opcion:
    case "1":
        libro = openpyxl.Workbook()
        hoja = libro["Sheet"]
        hoja.title = "Rodada"

        hoja["A1"].value = "CLAVE"
        hoja["B1"].value = "COLOR"

        hoja.column_dimensions["A"].width = 10
        hoja.column_dimensions["B"].width = 20

        for head in range(1, 3):
            hoja.cell(row = 1, column = head).border = border_title
            hoja.cell(row = 1, column = head).alignment = alignment_center
            hoja.cell(row = 1, column = head).fill = fill_title
            hoja.cell(row = 1, column = head).font = negritas

        i = 2
        for clave, color in registros_rodada:
            for j, elemento in enumerate((clave,color), start = 1):
                hoja.cell(row = i, column = j).value = elemento
                hoja.cell(row = i, column = j).border = border_content
                hoja.cell(row = i, column = j).alignment = alignment_center
            i += 1

        libro.save("Reporte_Rodadas.xlsx")
        print("\nEXPORTACIÓN EXITOSA - El archivo
Reporte_Rodadas.xlsx se ha generado exitosamente.")
        break

    case "2":
        with open("Reporte_Rodadas.csv", "w", encoding = "latin1",
newline="") as archivo:

```

```

        escritor = csv.writer(archivo)
        escritor.writerow(("CLAVE", "COLOR"))

        for clave, color in registros_rodada:
            escritor.writerow((clave, color))

        print("\nEXPORTACIÓN EXITOSA - El archivo
Reporte_Rodada.csv se ha generado exitosamente.")
        break

    case "3":
        break
    break
    break

case "3":
    print("\nMenú de procesos > Menú de informes > Menú de reportes > Listado
de unidades > Color")
    print("- Introduzca 'c' para cancelar el proceso -")

    while True:
        color_elegido = input("Introduzca el color con el cual se filtrarán los datos:
").strip().capitalize()

        if (color_elegido.upper() == "C"):
            break

        elif not color_elegido:
            print("Este campo es obligatorio. Por favor, introduzca el color de la
unidad.\n")

            continue

        elif not(color_elegido.replace(" ", "").isalpha()):
            print(f"El color de la unidad no debe contener números ni caracteres
especiales. Por favor, introduzca el color de la unidad únicamente con letras.\n")
            continue

        elif (len(color_elegido) > 15):
            print(f"El máximo de caracteres permitidos es de 14. Por favor, introduzca
el color de la unidad con una longitud de caracteres válida.\n")
            continue

        cursor.execute("SELECT clave, rodada FROM UNIDAD WHERE color = ?",
(color_elegido,))

        registros_color = cursor.fetchall()

        if registros_color:
            color_elegido = color_elegido.upper()
            print(f"\n{f'UNIDADES REGISTRADAS DE COLOR {color_elegido}':^45}")
            print("=" * 45)
            print(f"{'CLAVE':^20}{f'RODADA':^25}")
            print("=" * 45)

            [print(f"{clave:^20}{rodada:^25}") for clave, rodada in registros_color]
            print(f"{'=' * 45}\n")

```

```

else:
    print(f"Por el momento no hay unidades con color {color_elegido}
registradas. Por favor, introduzca algún otro de los colores válidos.\n")
    continue

while True:
    print("1- Guardar archivo Excel.")
    print("2- Guardar archivo CSV.")
    print("3- Regresar.")

    opcion = input("Introduzca la clave del proceso a realizar: ").strip()

    if not opcion:
        print("Este campo es obligatorio. Por favor, introduzca alguna de las
claves de procesos.\n")
        continue

    elif not(opcion.isdigit()):
        print(f"La clave {opcion} no es válida debido a que no es un valor
numérico. Por favor, introduzca alguna de las claves de procesos válidas.\n")
        continue

    elif not(int(opcion) in range(1, 4)):
        print(f"La clave {opcion} no es válida debido a que se encuentra fuera
del rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
        continue

    match opcion:

        case "1":
            libro = openpyxl.Workbook()
            hoja = libro["Sheet"]
            hoja.title = "Color"

            hoja["A1"].value = "CLAVE"
            hoja["B1"].value = "RODADA"

            hoja.column_dimensions["A"].width = 10
            hoja.column_dimensions["B"].width = 15

            for head in range(1, 3):
                hoja.cell(row = 1, column = head).border = border_title
                hoja.cell(row = 1, column = head).alignment = alignment_center
                hoja.cell(row = 1, column = head).fill = fill_title
                hoja.cell(row = 1, column = head).font = negritas

            i = 2
            for clave, rodada in registros_color:
                for j, elemento in enumerate((clave, rodada), start=1):
                    hoja.cell(row = i, column = j).value = elemento
                    hoja.cell(row = i, column = j).border = border_content
                    hoja.cell(row = i, column = j).alignment = alignment_center
                i += 1

```

```

        libro.save("Reporte_Colores.xlsx")
        print("\nEXPORTACIÓN EXITOSA - El archivo Reporte_Colores.xlsx
se ha generado exitosamente.")
        break

    case "2":
        with open("Reporte_Colores.csv", "w", encoding = "latin1",
newline="") as archivo:
            escritor = csv.writer(archivo)
            escritor.writerow(("CLAVE", "RODADA"))

            for clave, rodada in registros_color:
                escritor.writerow((clave, rodada))

            print("\nEXPORTACIÓN EXITOSA - El archivo
Reporte_Colores.csv se ha generado exitosamente.")
            break

    case "3":
        break
    break
    break

    case "4":
        print()
        break

    case "3":
        cursor.execute("""
            SELECT
            CAST(CASE
                WHEN p.fecha_retorno IS NOT NULL
                THEN julianday(p.fecha_retorno) - julianday(date(p.fecha_prestamo, '+'
|| p.dias_prestamo || ' days'))
            ELSE
                julianday('now') - julianday(date(p.fecha_prestamo, '+' ||
p.dias_prestamo || ' days'))
            END AS INTEGER) AS dias_retraso,
            strftime('%m-%d-%Y', date(p.fecha_prestamo, '+' || p.dias_prestamo || '
days')) AS fecha_limite,
            p.clave_unidad,
            u.rodada,
            u.color,
            (c.nombres || ' ' || c.apellidos) AS nombre_completo,
            c.telefono
            FROM PRESTAMO p LEFT JOIN UNIDAD u ON p.clave_unidad = u.clave
            LEFT JOIN CLIENTE c ON p.clave_cliente = c.clave
            WHERE (p.fecha_retorno IS NOT NULL AND date(p.fecha_prestamo, '+' ||
p.dias_prestamo || ' days') < p.fecha_retorno)
            OR (p.fecha_retorno IS NULL AND date(p.fecha_prestamo, '+' ||
p.dias_prestamo || ' days') < date('now'))
            ORDER BY 1 DESC;
        """)
        retrasos = cursor.fetchall()

```

```

if retrasos:
    print("\nMenú de procesos > Menú de informes > Menú de reportes > Retrasos\n")

    print(f'{"PRESTAMOS CON RETORNOS RETRASADOS":^170}')
    print("=" * 170)
    print(f'{"DIAS RETRASO":^14}{{"FECHA LIMITE":^16}{{"CLAVE
UNIDAD":^14}{{"RODADA":^14}{{"COLOR":^16}{{"NOMBRE COMPLETO":^82}{{"TELEFONO":^14}}}')
    print("=" * 170)

[print(f'{"dias_retraso":^14}{{"fecha_limite":^16}{{"clave_unidad":^14}{{"rodada":^14}{{"color":^16}{{"nombre_com
pleto":^82}{{"tel":^14}}}') for dias_retraso, fecha_limite, clave_unidad, rodada, color, nombre_completo,
tel in retrasos]
    print("=" * 170, "\n")

while True:
    print("1- Guardar archivo Excel.")
    print("2- Guardar archivo CSV.")
    print("3- Regresar.")

    opcion = input("Introduzca la clave del proceso a realizar: ").strip()

    if not opcion:
        print("Este campo es obligatorio. Por favor, introduzca alguna de las claves de
procesos.\n")
        continue

    elif not(opcion.isdigit()):
        print(f"La clave {opcion} no es válida debido a que no es un valor numérico. Por
favor, introduzca alguna de las claves de procesos válidas.\n")
        continue

    elif not(int(opcion) in range(1, 4)):
        print(f"La clave {opcion} no es válida debido a que se encuentra fuera del rango
de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
        continue

    match opcion:
        case "1":
            libro = openpyxl.Workbook()
            hoja = libro["Sheet"]
            hoja.title = "Retrasos"

            hoja["A1"].value = "DIAS RETRASO"
            hoja["B1"].value = "FECHA LIMITE"
            hoja["C1"].value = "CLAVE UNIDAD"
            hoja["D1"].value = "RODADA"
            hoja["E1"].value = "COLOR"
            hoja["F1"].value = "NOMBRE COMPLETO"
            hoja["G1"].value = "TELEFONO"

            hoja.column_dimensions["A"].width = 15
            hoja.column_dimensions["B"].width = 20
            hoja.column_dimensions["C"].width = 20

```

```

hoja.column_dimensions["D"].width = 15
hoja.column_dimensions["E"].width = 20
hoja.column_dimensions["F"].width = 85
hoja.column_dimensions["G"].width = 20

for head in range(1, 8):
    hoja.cell(row = 1, column = head).border = border_title
    hoja.cell(row = 1, column = head).alignment = alignment_center
    hoja.cell(row = 1, column = head).fill = fill_title
    hoja.cell(row = 1, column = head).font = negritas

i = 2
for dias_retraso, fecha_limite, clave_unidad, rodada, color,
nombre_completo, tel in retrasos:
    for j, elemento in enumerate((dias_retraso, fecha_limite, clave_unidad,
rodada, color, nombre_completo, tel), start=1):
        hoja.cell(row = i, column = j).value = elemento
        hoja.cell(row = i, column = j).border = border_content
        hoja.cell(row = i, column = j).alignment = alignment_center
    i += 1

libro.save("Reporte_Retrasos.xlsx")
print("\nEXPORTACIÓN EXITOSA - El archivo Reporte_Retrasos.xlsx se ha
generado exitosamente.\n")
break

case "2":
    with open("Reporte_Retrasos.csv", "w", encoding = "latin1", newline="") as
archivo:
        escritor = csv.writer(archivo)
        escritor.writerow(("DIAS RETRASO", "FECHA LIMITE", "CLAVE
UNIDAD", "RODADA", "COLOR", "NOMBRE COMPLETO", "TELEFONO"))
        for dias_retraso, fecha_limite, clave_unidad, rodada, color,
nombre_completo, tel in retrasos:
            escritor.writerow((dias_retraso, fecha_limite, clave_unidad, rodada,
color, nombre_completo, tel))

        print("\nEXPORTACIÓN EXITOSA - El archivo Reporte_Retrasos.csv se
ha generado exitosamente.\n")
        break

case "3":
    print(); break

else:
    print("\nActualmente no se cuenta con retornos atrasados.\n")

case "6":
    print(); break

def analisis():

    while True:
        cursor.execute("SELECT COUNT(*) FROM PRESTAMO")
        registros = cursor.fetchone()[0]

```



```

if registros == 0:
    print("\nActualmente no se cuenta con préstamos registrados. Por favor, registre
préstamos antes de solicitar análisis.\n")
    return

print("Menú de procesos > Menú de informes > Menú de análisis")
print(f"MENÚ DE ANÁLISIS {'-' * 10}")
print("1- Duración de los préstamos.")
print("2- Ranking de clientes.")
print("3- Preferencia de rentas.")
print("4- Regresar.")

opcion_Submenu = input("Introduzca la clave la operación a realizar: ").strip()

if not opcion_Submenu:
    print("Este campo es obligatorio. Por favor, introduzca alguna de las claves de
procesos.\n")
    continue

elif not(opcion_Submenu.isdigit()):
    print(f"La clave {opcion_Submenu} no es válida debido a que no es un valor numérico.
Por favor, introduzca alguna de las claves de procesos válidas.\n")
    continue

elif not(int(opcion_Submenu) in range(1, 5)):
    print(f"La clave {opcion_Submenu} no es válida debido a que se encuentra fuera del
rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
    continue

match opcion_Submenu:

    case "1":
        cursor.execute("SELECT dias_prestamo FROM PRESTAMO")
        registro_dias = [i[0] for i in cursor.fetchall()]
        array_dias = np.array(registro_dias)

        moda = stats.mode(array_dias)

        print("\nMenú de procesos > Menú de informes > Menú de análisis > Duración de los
préstamos")
        print(f"\n{'ESTADÍSTICOS DESCRIPTIVOS - DURACIÓN DE LOS
PRÉSTAMOS':^135}")
        print("=" * 135)

        print(f"{'MEDIA':^10}{ 'MEDIANA':^10}{ 'MODA':^10}{ 'MÍNIMO':^10}{ 'MÁXIMO':^10}{ 'DESVIACIÓN
ESTÁNDAR':^25}{ 'PRIMER CUARTIL':^20}{ 'SEGUNDO CUARTIL':^20}{ 'TERCER CUARTIL':^20}")
        print("=" * 135)

        print(f"{array_dias.mean():^10.2f}{np.median(array_dias):^10.2f}{moda[0]:^10}{array_dias.min():^10
}{array_dias.max():^10}{array_dias.std():^25.4f}{np.percentile(array_dias,
25):^20.2f}{np.percentile(array_dias, 50):^20.2f}{np.percentile(array_dias, 75):^20.2f}")
        print(f"{'=' * 135}\n")

    case "2":

```

```

cursor.execute("""
    SELECT
    COUNT(p.clave_cliente) AS cantidad_de_rentas_acumuladas,
    p.clave_cliente,
    (c.nombres || ' ' || c.apellidos) AS nombre_completo,
    c.telefono
    FROM PRESTAMO p LEFT JOIN CLIENTE c ON p.clave_cliente = c.clave
    WHERE p.fecha_prestamo <= date('now')
    GROUP BY p.clave_cliente, nombre_completo, c.telefono
    ORDER BY 1 DESC;
""")
ranking = cursor.fetchall()

print("\nMenú de procesos > Menú de informes > Menú de análisis > Ranking de
clientes")
print(f"\n{'RANKING DE CLIENTES':^150}")
print("=" * 150)
print(f"{'CANTIDAD DE RENTAS ACUMULADAS':^35}{{'CLAVE':^15}{{'NOMBRE
COMPLETO':^80}{{'TELÉFONO':^20}}}")
print("=" * 150)
[print(f"{'acum_rentas':^35}{{'clave':^15}{{'usuario':^80}{{'tel':^20}}}") for acum_rentas, clave,
usuario, tel in ranking]
print(f"{'=' * 150}\n")

case "3":
    while True:
        print("\nMenú de procesos > Menú de informes > Menú de análisis > Menú de
preferencias de rentas")
        print(f"{'MENÚ DE PREFERENCIAS DE RENTAS {'-' * 10}}")
        print("1- Por rodada.")
        print("2- Por color.")
        print("3- Por día de la semana.")
        print("4- Regresar.")

        opcion_Submenu = input("Introduzca la clave la operación a realizar: ").strip()

        if not opcion_Submenu:
            print("Este campo es obligatorio. Por favor, introduzca alguna de las claves de
procesos.\n")
            continue

        elif not(opcion_Submenu.isdigit()):
            print(f"La clave {opcion_Submenu} no es válida debido a que no es un valor
numérico. Por favor, introduzca alguna de las claves de procesos válidas.\n")
            continue

        elif not(int(opcion_Submenu) in range(1, 5)):
            print(f"La clave {opcion_Submenu} no es válida debido a que se encuentra fuera
del rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
            continue

        match opcion_Submenu:
            case "1":
                cursor.execute("SELECT COUNT(*) FROM PRESTAMO")
                total_registros = cursor.fetchone()[0]

```

```

        cursor.execute("""
            SELECT COUNT(*) AS cantidad_rentas, u.rodada FROM
PRESTAMO p
            LEFT JOIN UNIDAD u ON p.clave_unidad = u.clave
            GROUP BY u.rodada ORDER BY 1 DESC;
        """)

registros_filtrados = cursor.fetchall()

print("\nMenú de procesos > Menú de informes > Menú de análisis > Menú de
preferencias de rentas > Preferencia por rodada")
print(f"\n{'PREFERENCIAS DE RENTAS - RODADA':^45}")
print("=" * 45)
print(f"{'CANTIDAD DE PRÉSTAMOS':^25}{{'RODADA':^20}}")
print("=" * 45)
[print(f"{'cantidad':^25}{rodada:^20}") for cantidad, rodada in registros_filtrados]
print(f"{'=' * 45}")

rodadas = [f"Rodada {registro[1]}" for registro in registros_filtrados]
porcentajes = [(cantidad[0] / total_registros) * 100 for cantidad in
registros_filtrados]

plt.figure(figsize=(8, 8))
plt.pie(porcentajes, labels = rodadas, autopct = "%1.2f%%", startangle = 140)
plt.title("Porcentaje de préstamos por rodada")
plt.show()

case "2":
    cursor.execute("SELECT COUNT(*) FROM PRESTAMO")
    total_registros = cursor.fetchone()[0]

    cursor.execute("""
        SELECT COUNT(*) AS cantidad_rentas, u.color FROM PRESTAMO
p
        LEFT JOIN UNIDAD u ON p.clave_unidad = u.clave
        GROUP BY u.color ORDER BY 1 DESC;
    """)

registros_filtrados = cursor.fetchall()

print("\nMenú de procesos > Menú de informes > Menú de análisis > Menú de
preferencias de rentas > Preferencia por color")
print(f"\n{'PREFERENCIAS DE RENTAS - COLOR':^45}")
print("=" * 45)
print(f"{'CANTIDAD DE PRÉSTAMOS':^25}{{'COLOR':^20}}")
print("=" * 45)
[print(f"{'cantidad':^25}{color:^20}") for cantidad, color in registros_filtrados]
print(f"{'=' * 45}")

colores = [f"{'registro[1]'}" for registro in registros_filtrados]
porcentajes = [(cantidad[0] / total_registros) * 100 for cantidad in
registros_filtrados]

plt.figure(figsize=(8, 8))

```

```

plt.pie(porcentajes, labels = colores, autopct = "%1.2f%%", startangle = 140)
plt.title("Porcentaje de préstamos por color")
plt.show()

case "3":
    cursor.execute("SELECT COUNT(*) FROM PRESTAMO")
    total_registros = cursor.fetchone()[0]

    cursor.execute("""
        SELECT CASE strftime('%w', fecha_prestamo)
            WHEN '0' THEN 'Domingo'
            WHEN '1' THEN 'Lunes'
            WHEN '2' THEN 'Martes'
            WHEN '3' THEN 'Miércoles'
            WHEN '4' THEN 'Jueves'
            WHEN '5' THEN 'Viernes'
            WHEN '6' THEN 'Sábado'
        END AS dia_semana,
        COUNT(*) AS total_prestamos
        FROM PRESTAMO GROUP BY 1 ORDER BY 1;
    """)
    registros_filtrados = cursor.fetchall()

    print("\nMenú de procesos > Menú de informes > Menú de análisis > Menú de
preferencias de rentas > Preferencia por día de la semana")
    print(f"\n{'PREFERENCIAS DE RENTAS - DÍA DE LA SEMANA':^45}")
    print("=" * 45)
    print(f"{'CANTIDAD DE PRÉSTAMOS':^25}{'DÍA DE LA SEMANA':^20}")
    print("=" * 45)
    [print(f"{'cantidad':^25}{'dia':^20}") for dia, cantidad in registros_filtrados]
    print(f"{'=' * 45}")

    dias = [f"{'registro[0]'}" for registro in registros_filtrados]
    porcentajes = [(cantidad[1] / total_registros) * 100 for cantidad in
registros_filtrados]

    plt.figure(figsize=(8, 8))
    plt.pie(porcentajes, labels = dias, autopct = "%1.2f%%", startangle = 140)
    plt.title("Porcentaje de préstamos por día de la semana")
    plt.show()

    case "4":
        print()
        break

case "4":
    print()
    break

while True:
    try:
        print(f"MENÚ DE PROCESOS {'-' * 10}")
        print("1- Registro.")
        print("2- Préstamo.")
        print("3- Retorno.")
        print("4- Informes.")

```

```

print("5- Salir.")

opcion_Menu = input("Introduzca la clave del proceso a realizar: ").strip()
print()

if not opcion_Menu:
    salida = input("Presione Enter para salir o introduzca cualquier otro caracter para continuar: ")
    if (salida == ""): break
    else: continue

elif not(opcion_Menu.isdigit()):
    print(f"La clave {opcion_Menu} no es válida debido a que no es un valor numérico. Por favor, introduzca alguna de las claves de procesos válidas.\n")
    continue

elif not(int(opcion_Menu) in range(1, 6)):
    print(f"La clave {opcion_Menu} no es válida debido a que se encuentra fuera del rango de opciones. Por favor, introduzca alguna de las claves de procesos válidas.\n")
    continue

match(opcion_Menu):
    case "1":
        registro()
        print()
    case "2":
        prestamo()
        print()
    case "3":
        retorno()
        print()
    case "4":
        informe()
        print()
    case "5":
        confirmacion = input("Presione Enter para salir o introduzca cualquier otro caracter para continuar: ").strip()
        if (confirmacion == ""):
            print("\nSaliendo...")
            break
        else:
            print()
            continue

except Exception:
    excepcion = sys.exc_info()
    print(f"Ha ocurrido una excepción de tipo {excepcion[0]}. Mensaje de la excepción: {excepcion[1]}")

```

REPORTE DE PLATAFORMA TECNOLÓGICA

1. Hardware: Especificaciones Técnicas

Para ejecutar el entorno de desarrollo Visual Studio Code (VS Code), es necesario cumplir con ciertos requisitos de hardware que dependen del sistema operativo en el cual se va a utilizar. Según Microsoft, los requerimientos generales para usar Visual Studio Code son los siguientes:

Especificaciones mínimas de hardware:

- Procesador: 1.6 GHz o superior (preferiblemente de múltiples núcleos para un mejor rendimiento).
- Memoria RAM: 2 GB de RAM o más (recomendado 4 GB o más, especialmente al trabajar con múltiples extensiones y proyectos grandes).
- Espacio en disco: Al menos 200 MB de espacio libre para la instalación de Visual Studio Code. Sin embargo, se recomienda tener varios GB libres, ya que los proyectos y las dependencias pueden consumir más espacio.
- Resolución de pantalla: 1024 x 768 píxeles o mayor.
- Gráficos: Un chip gráfico estándar, aunque VS Code no es particularmente exigente en cuanto a gráficos.

Sistemas operativos compatibles:

- Windows: Desde Windows 7 en adelante (32 o 64 bits).
- macOS: Soportado desde macOS 10.11 (El Capitan) o versiones posteriores.
- Linux: Distribuciones más comunes como Ubuntu, Debian, Fedora y Red Hat.

2. Software: Versiones, incluyendo librerías

Versión de la plataforma de desarrollo (Visual Studio Code):

- Visual Studio Code: La versión más reciente disponible es la recomendada. A partir de la fecha del reporte, la última versión estable de Visual Studio Code es la versión 1.79.2.

Versión de Python:

Para el desarrollo en Python, es fundamental tener una versión compatible y actualizada de Python.

- Python: La última versión estable de Python es Python 3.12.0, aunque para algunos proyectos puede ser más estable usar la versión 3.11.x si las librerías no están actualizadas.

- Se recomienda mantener el entorno actualizado utilizando herramientas como pyenv o venv para gestionar múltiples versiones de Python en un mismo sistema.

Base de datos SQLite:

Para el almacenamiento y la manipulación de datos, se utilizó SQLite, una base de datos ligera y de fácil integración en Python, ideal para el desarrollo de prototipos o aplicaciones pequeñas.

- Versión de SQLite: Es recomendable utilizar la última versión de SQLite compatible con Python. Al momento del reporte, SQLite 3.42.0 es una versión estable comúnmente integrada en Python.
- Configuración de SQLite: La base de datos SQLite es independiente, lo que significa que no requiere configuración de servidor ni credenciales adicionales. La estructura de datos se almacena en un archivo .sqlite, que puede manipularse directamente mediante Python o mediante herramientas de administración compatibles.

Librerías importadas:

1. csv: Librería estándar de Python para manipulación de archivos CSV.
2. sys: Librería estándar de Python que proporciona acceso a variables y funciones relacionadas con el sistema.
3. datetime: Librería estándar de Python para trabajar con fechas y horas.
4. openpyxl: Librería externa para manipular archivos de Excel en formato .xlsx.
5. numpy: Librería popular en Python para computación numérica eficiente y manipulación de matrices.
6. scipy.stats: Módulo de la librería SciPy, utilizado para funciones estadísticas y análisis de datos.
7. openpyxl.styles: Parte de la librería openpyxl, que permite estilizar celdas en archivos de Excel, como cambiar fuentes, bordes, alineación, etc.
8. sqlite3: Módulo estándar de Python para interactuar con bases de datos SQLite. Permite crear, leer, actualizar y eliminar datos en una base de datos SQLite.
9. Matplotlib: Librería de Python para la creación de gráficos y visualizaciones. Es muy utilizada para representar datos de manera visual, incluyendo gráficos de líneas, barras, pastel, entre otros.

Versión de las librerías:

Para asegurar la compatibilidad y correcto funcionamiento del proyecto, asegúrate de tener

instaladas las siguientes versiones o versiones compatibles de las librerías:

- openpyxl: La versión estable más reciente es openpyxl 3.1.2.
- numpy: La versión estable es numpy 1.23.4.
- scipy: La versión estable más reciente es scipy 1.10.0.
- matplotlib: La versión estable más reciente es matplotlib 3.6.2.

APORTACIONES ESPECIFICAS

- Loera Alvarez Denis Franco

Documento PIA.

Líneas 1-64 (contextualizador, base de datos y librerías)

Líneas 1412-1716 (retrasos y función completa de análisis)

- Chávez Juárez Alan Alexis

Líneas 65-215 (Función de registros de unidades y clientes)

Líneas 1718-1774 (Menú principal y manejador de excepciones)

Líneas 478-956 (Función de menú informe, Menú de reporte, Función de reportes de clientes y reporte de préstamos por retornar)

- Garza López Omar de Jesús.

Líneas 216-477 (Función de préstamo de unidades y Función de retorno de préstamos)

Líneas 957-1411 (funciones y opciones de submenú Listado de unidades, Reporte de préstamos por periodo)