



Informe del Proyecto 1 - Parte 2: Automóviles

Fabricio Josue Lopez Reyes , 202210190

Isabella Luana Romero Celis , 202210302

Gabriel Eduardo Romero Diez , 202210303

Valeria Celeste Valdez Campoverde , 202210359

Kharol Emilia Villamonte Montesinos , 202210373

Josue Yip Rodriguez , 202120039

Ciencias de la Computación, Universidad de Tecnología

CS 1111 - Programación 1

Jesus Bellido // Carlos Alberto Sotomayor Beltran

2022

1. Resumen:

El siguiente informe tiene como objetivo permitir que cualquiera pueda ser capaz de entender el funcionamiento y emplear el mismo. Para ello se ha dividido en 6 partes estratégicas:

- **Introducción**
- **Definiciones**
- **Funcionamiento del programa**
- **Ejemplos de ejecución**
- **Conclusiones y recomendaciones**
- **Referencias Bibliográficas**

2. Introducción:

Durante los últimos años, la industria automotriz en el Perú no ha hecho más que crecer. Hasta abril del 2022, la información más reciente de la Superintendencia Nacional de Registros Públicos (Sunarp) revelaba que en el segmento de cuatro meses se vendieron 52 904 unidades de vehículos livianos, un incremento del 11,6% comparado al 2021. De este incremento, se estima que 12,913 unidades son automóviles (Gerencia de Estudios Económicos de la AAP, 2022).

Por otro lado, de acuerdo con el Índice Nacional del “Flujo Vehicular por Unidades de Peaje” (INEI, 2022) que registra movimientos de entradas y salidas del tráfico vehicular creció un 4,5% en comparación al 2021. De este porcentaje, un 5,7% eran vehículos ligeros. De la misma manera, la venta e inmatriculación de los mismos ha incrementado en más de 10 000 unidades durante el periodo Enero - Mayo.

De acuerdo a las estadísticas presentadas, se puede inferir la importancia de la industria automotriz en el Perú, así como de su adecuado registro, pues las estadísticas son de utilidad al momento de plantear políticas de transporte o como un indicador de actividad y desarrollo económico en diversas regiones según su tráfico vehicular. De esta manera, se considera que el presente trabajo sería de gran ayuda a este fin.

La decisión de crear este software nació con el fin de poder agilizar y automatizar las ventas del mercado automotriz nacional. El objetivo del proyecto es brindar soluciones tecnológicas de calidad que permitan a las MYPEs poder competir en el mercado y de esta manera potenciar y organizar de manera efectiva su pequeño negocio.

3. Definiciones:

3.1. **menu_opcion():**

Esta función se encarga de mostrar cuatro opciones numeradas (registrar auto, autos disponibles, comprar autos, autos vendidos, cantidad total en S/. de autos vendidos y salir) en un formato de menú. Además permite al usuario seleccionar qué opción quiere implementar a continuación.

3.2. **separar_por():**

Esta función separa los datos del archivo “registro.txt”, donde se almacenan los datos de los carros, para que puedan ser legibles para el programa.

3.3. **registro_auto():**

Esta función se encarga de almacenar características (marca, año de fabricación, color y precio) ingresadas por el usuario, para luego ser mostradas en la relación de autos disponibles en un formato de tabla. Además mediante el uso de la librería os corrobora la existencia del *archivo.txt*, de lo contrario, procede a crearlo.

3.4. **inventario_auto():**

Esta función se encarga de mostrar la relación de los autos disponibles para la venta según el inventario de la tienda.

3.5. **inventario_auto_vendido():**

Esta función a su vez, utiliza la función *cambiar_orden()* para poder mostrar la relación de autos vendidos.

3.6. **filtrar_por():**

Esta función se encarga de filtrar el inventario de auto para luego mostrarnos una matriz (almacenamiento de información ordenada) de todos los autos que coincidan con las características previamente ingresadas en la sección de registro de auto, hasta que el usuario compre el auto que desee.

3.7. *comprar_auto()*:

Esta función permite al usuario comprar un auto que se encuentre disponible al introducir datos como (marca, año de fabricación, color y precio) para posteriormente mostrar que el carro ha sido vendido satisfactoriamente.

3.8. *reemplazar_vendido()*:

En el caso de que un carro sea vendido, esta función se encarga de reemplazar el estado del carro (“disponible” por “vendido”).

3.9. *eliminar_vendido()*:

Detecta los carros vendidos para que sean eliminados y muestre los nuevos carros disponibles con un nuevo id asignado acorde al total de autos.

3.10. *cambiar_orden()*:

Esta función se encarga de re- asignar ids para ordenar los autos en orden alfabético. En el caso que sean de la misma marca se pondrá primero el menos costoso.

3.11. *si_queda_uno()*:

Busca que la opción de compra de auto sea sea válida para luego ejecutar las funciones *reemplazar_vendido* y *eliminar_vendido*.

3.12. *total_vendidos()*:

Esta función cuenta la cantidad de autos del archivo *vendidos.txt*, luego identifica el dato número 4 (el precio) y los suma para así calcular el dinero total.

→ *librería tabulate*:

Esta librería permite al usuario imprimir o escribir un texto en tablas en distintos formatos.

→ *librería os*:

Esta librería permite interactuar con el sistema operativo de la computadora, desde crear, mover y eliminar archivos.

- *Función limpiar_consola():*

Esta función permite tener una interacción más limpia con el usuario.

Limpiando la consola cada que algunas de las funciones del menú es llamada.

```
def limpiar_consola():  
  
    if os.name == "nt":  
        #Windows  
        os.system("cls")  
    else:  
        #Mac Os  
        os.system("clear")
```

- *Función enterContinuar():*

Esta función hace que el usuario presione la tecla enter para continuar con el programa.

```
def enterContinuar():  
    a = input("Presiona enter para continuar...")  
    a = ""  
    if a == "":  
        limpiar_consola()  
        menu_opcion()
```

- *Función total_vendidos():*

Esta función muestra la cantidad de carros vendidos y el costo total de los autos vendidos.

```
def total_vendidos():
    titulos.total_vendidos()
    if os.path.exists("vendidos.txt"):
        contador = 0
        precios = []
        with open("vendidos.txt", "r") as file:
            for line in file:
                a = line.split(",")
                b = int(a[4])
                precios.append(b)
                contador += 1
        total = sum(precios)
        print("El total de autos vendidos es: {} \n
Cantidad en soles de autos vendidos: S/{} ".format(contador, total))
    else:
        print("No hay autos vendidos")
```

- *Función filtrar_por():*

Esta función filtra los autos dependiendo de sus características; marca, año de fabricación, color y precio.

```
#Filtro de selección para comprar auto
def filtrar_por(dato, matriz_ingresada, posicion):
    limpiar_consola()
    nueva_matriz = []

    #Filtrar por dato
    for i in range(0, len(matriz_ingresada)):
        if matriz_ingresada[i][posicion] == dato :
            nueva_matriz.append(matriz_ingresada[i])

    #imprimo titulo
    titulos.autos_disponibles()

    #Devuelvo la matriz de los autos seleccionados por el usuario
    return nueva_matriz
```

- *Función si_queda_uno():*

Cuando el usuario termina de escoger el auto y sólo queda uno en el registro filtrado, le pregunta al usuario si desea comprar el auto que sobra.

```
def si_queda_uno(auto_a_comprar):
    if len(auto_a_comprar) == 1:
        strauto_a_comprar = ",".join(auto_a_comprar[0])
        strauto_a_comprar = strauto_a_comprar + "\n"

        print("Deseas comprar este auto?: ")
        comprar = input("1.- sí\n2.- Volver al menú\n3.-Salir\nTu respuesta: ")
        if comprar == "1":
            #Codigo reutilizado de "inventario_auto()"
            cambiar_orden("registro.txt")
            reemplazar_vendido("registro.txt", int(auto_a_comprar[0][0]))
            limpiar_consola()
            print("Gracias por su compra")
            inventario_auto()
            eliminar_vendido("registro.txt")
            enterContinuar()
        elif comprar == "2":
            menu_opcion()
        elif comprar == "3":
            exit()
        else:
            print("Introduce una opción válida")
            print("Deseas comprar este auto?: ")
            comprar = input("1.- sí\n2.- volver\n3.-Salir\nTu respuesta: ")
    elif len(auto_a_comprar) == 0:
        print("Lo sentimos no disponenmos el auto que deseas comprar")
        print("Vuelve a intentarlo")
        enterContinuar()
```

- *Función cambiar_orden():*

Esta función ordena de manera alfabética los autos.

```
def cambiar_orden(file):
    listasa = []
    with open(file, "r") as f:
        for line in f:
            apple = line.split(",")
            listasa.append(apple)
    listasa.sort(key = lambda x: x[1] + x[4])
    apa = []
    contador = 1
    for item in listasa:
        b = ",".join(item[1:])
        apa.append("{} , {}".format(contador, b))
        contador += 1
    os.remove(file)
    with open(file, 'a') as f:
        for line in apa:
            f.write(line)
```

- *Función eliminar_vendido():*

Esta función filtra entre los autos que se encuentran vendidos y disponibles, guardando los disponibles una lista y a los vendidos en otra, eliminando a los vendidos a su vez.

```
def eliminar_vendido(file):
    temporal = []
    temporal_vendido = []
    with open(file, 'r') as f:
        contador = 1
        for line in f:
            a = line.split(",")
            if a[-1] == "Vendido\n":
                n = a[1:-1]
                b = ",".join(n)
                z = "{}Vendido\n".format(b)
                temporal_vendido.append(z)
            else:
                n = a[1:-1]
                b = ",".join(n)
                z = "{}{},{}Disponible\n".format(contador, b)
                temporal.append(z)
                contador += 1
    os.remove(file)
```

```
os.remove(file)
with open(file, 'w') as f:
    for line in temporal:
        f.write(line)
a = temporal_vendido[0].split(",")
if os.path.exists("vendidos.txt"):
    with open("vendidos.txt", "r") as file:
        contador = 1
        for line in file:
            contador += 1
    b = ",".join(a)
    registro = "{}{}".format(contador, b)
    with open("vendidos.txt", "a") as file:
        file.write(registro)
else:
    b = ",".join(a)
    registro = "{}{}".format(1, b)
    with open("vendidos.txt", "a") as file:
        file.write(registro)
```

- Función *reemplazar vendido*

Esta función reemplaza el estado "disponible" del auto al estado "vendido"

```

def reemplazar_vendido(file, x):
    temporal = []
    with open(file, 'r') as f:
        contador = 1
        for line in f:
            if contador == x:
                a = line.split(",")
                a[-1] = "Vendido\n"
                b = ",".join(a)
                temporal.append(b)
                contador += 1
            else:
                temporal.append(line)
                contador += 1
    os.remove(file)
    with open(file, 'w') as f:
        for line in temporal:
            f.write(line)

```

- *Función comprar_auto():*

Esta función permite comprar un auto de la lista de disponibles.

La función hace uso de las siguientes funciones:

-filtrar_por():

-si_queda_uno():

```
def comprar_auto():
    if os.path.exists("registro.txt"):
        #Mostrar título de compra de autos
        titulos.compra_de_autos()

    #Creación de la nueva matriz que se encargará de los otros datos
    with open("registro.txt", "r") as lineas:
        listaLineas = []
        listaLineas = (lineas.readlines())
        matrisita = []
        for i in listaLineas:
            lineax = i.split(",")
            matrisita.append(lineax)
```

Filtro dependiendo de sus características

```
#Filtrar datos
#marca
marca_seleccionada = input("Introduce la marca del auto que deseas comprar: ")
marca_seleccionada = marca_seleccionada.lower().capitalize()
lista_de_auto_a_comprar = filtrar_por(marca_seleccionada, matrisita, 1)
print(tabulate(lista_de_auto_a_comprar, headers = ["Marca", "Fabricacion", "Color", "Precio", "Estado"],
tablefmt="fancy_grid" ))
si_queda_uno(lista_de_auto_a_comprar)

#Fabricación
fabricacion_seleccionada = input("Introduce el año de fabricación del auto que deseas comprar: ")
fabricacion_seleccionada = fabricacion_seleccionada.lower().capitalize()
lista_de_auto_a_comprar = filtrar_por(fabricacion_seleccionada, lista_de_auto_a_comprar, 2)
print(tabulate(lista_de_auto_a_comprar, headers = ["", "Marca", "Fabricacion", "Color", "Precio",
"Estado"], tablefmt="fancy_grid" ))
si_queda_uno(lista_de_auto_a_comprar)

#Color
color_seleccionado = input("Introduce el color seleccionado del auto que deseas comprar: ")
color_seleccionado = color_seleccionado.lower().capitalize()
lista_de_auto_a_comprar = filtrar_por(color_seleccionado, lista_de_auto_a_comprar, 3)
print(tabulate(lista_de_auto_a_comprar, headers = ["", "Marca", "Fabricacion", "Color", "Precio",
"Estado"], tablefmt="fancy_grid" ))
si_queda_uno(lista_de_auto_a_comprar)

#Precio
precio_seleccionado = input("Introduce el precio seleccionado del auto que deseas comprar: ")
precio_seleccionado = precio_seleccionado.lower().capitalize()
lista_de_auto_a_comprar = filtrar_por(precio_seleccionado, lista_de_auto_a_comprar, 4)
print(tabulate(lista_de_auto_a_comprar, headers = ["", "Marca", "Fabricacion", "Color", "Precio",
"Estado"], tablefmt="fancy_grid" ))
si_queda_uno(lista_de_auto_a_comprar)
```

Si es que no hay autos registrados

```

else:
    print("No hay autos registrados")
    enterContinuar()

```

- *Función inventario_auto_vendido():*

Esta función muestra los autos que se han vendido.

```

def inventario_auto_vendido():
    if os.path.exists("vendidos.txt"):
        cambiar_orden("vendidos.txt")
        with open("vendidos.txt", "r") as lineas:
            listaLineas = []
            listaLineas = (lineas.readlines())
            matrisita = []
            for i in listaLineas:
                lineax = i.split(",")
                matrisita.append(lineax)

            titulos.autos_vendidos()

            #Imprimo la tablita
            print(tabulate(matrisita, headers = ["","Marca", "Fabricacion", "Color", "Precio",
"Estado"], tablefmt="fancy_grid" ))
    else:
        print("No hay autos vendidos")
        enterContinuar()

```

- *Función inventario_auto():*

Esta función muestra los autos disponibles en el sistema.

```
#Mostrar el inventario de la tienda(los autos disponibles)
def inventario_auto():
    if os.path.exists("registro.txt"):
        cambiar_orden("registro.txt")
        with open("registro.txt", "r") as lineas:
            listaLineas = []
            listaLineas = (lineas.readlines())
            matrisita = []
            for i in listaLineas:
                lineax = i.split(",")
                matrisita.append(lineax)

            titulos.autos_disponibles()

            #Imprimo la tablita
            print(tabulate(matrisita, headers = ["", "Marca", "Fabricacion", "Color", "Precio",
"Estado"], tablefmt="fancy_grid" ))
    else:
        print("No hay autos registrados")
        enterContinuar()
```

- ***Función registro_auto():***

Esta función permite que el usuario registre un auto por sus características; marca, año de fabricación, color y precio.

```
#Registro de autos
def registro_auto():
    titulos.registro_auto()
    try:
        marca = input("Introduce la Marca: ")
        fabricacion = int(input("Introduce el año de Fabricación: "))
        if fabricacion < 500:
            print("-----")
            print(
"El año de fabricación no puede ser menor a 500 años, por favor vuelva a intentarlo")
            print("-----")
            registro_auto()
        color = input("Introduce el Color: ")
        precio = int(input("Introduce el Precio: "))
        if precio < 0:
            print("-----")
            print("El precio no puede ser negativo, por favor vuelva a intentarlo")
            print("-----")
            registro_auto()
```

```

except:
    print("-----")
    print("Ingresaste un valor erroneo, por favor vuelve a ingresar los datos: ")
    print("-----")
    registro_auto()
disponibilidad = "Disponible"
if os.path.exists("registro.txt"):
    with open("registro.txt", "r") as file:
        contador = 1
        for line in file:
            contador += 1
        registro = "{},{},{},{},{},{},{}\n".format(contador,marca.lower().capitalize() , fabricacion, color.
lower().capitalize() , precio, disponibilidad.lower().capitalize())
        with open("registro.txt", "a") as file:
            file.write(registro)
else:
    contador = 1
    registro = "{},{},{},{},{},{},{}\n".format(contador,marca.lower().capitalize() , fabricacion, color
.lower().capitalize() , precio, disponibilidad.lower().capitalize())
    with open("registro.txt", "a") as file:
        file.write(registro)
#Fin del programa e interacción con el usuario
limpiar_consola()
titulos.registro_completado()
print("\n Registro exitoso !!!")

```

- ***Función menu_opcion():***

Esta función permite al usuario elegir entre las siguientes opciones:

1. *Registrar auto*
2. *Autos disponibles*
3. *Comprar autos*
4. *Autos vendidos*
5. *Cantidad total en S/ de autos vendidos*
6. *Salir*


```
from tabulate import tabulate
import os, titulos

#Función para mostrar el menú de opciones
def menu_opcion():
    titulos.menu_opcion()
    while True:
        opcion = input("opcion: ")
        #if opcion < 4 and opcion > 0:
        if opcion == "1":
            limpiar_consola()
            registro_auto()
            enterContinuar()
            limpiar_consola()
            break
        elif opcion == "2":
            limpiar_consola()
            inventario_auto()
            enterContinuar()
            limpiar_consola()
            break
        elif opcion == "3":
            limpiar_consola()
            comprar_auto()
            enterContinuar()
            limpiar_consola()
            break
        elif opcion == "4":
            limpiar_consola()
            inventario_auto_vendido()
            enterContinuar()
            limpiar_consola()
            break
        elif opcion == "5":
            limpiar_consola()
            total_vendidos()
            enterContinuar()
            limpiar_consola()
            break
        elif opcion == "6":
            limpiar_consola()
            exit()
        else:
            print("Introduce una opcion valida, por favor vuelva a intentarlo")
```

4.- Ejemplos de ejecución:

Ejecutamos el programa y nos aparece esta interfaz

```
=====
=====
Venta de autos:
=====
=====

1. Registrar auto
2. Autos disponibles
3. Comprar autos
4. Autos vendidos
5. Cantidad total en S/ de autos vendidos
6. Salir

opcion: |
```

El programa nos pide que ingresemos una opción para iniciar con la ejecución de este.

Seleccionemos la opción 1 y rellenamos los datos que el programa nos pide:

```
Registro de autos

Introduce la Marca: Toyota Corola
Introduce el año de Fabricación: 2077
Introduce el Color: Azul
Introduce el Precio: 30000
```

Al terminar nos aparecerá una pantalla de confirmación donde veremos que nuestro auto ya ha sido registrado



Al presionar Enter nos lleva a la pantalla principal nuevamente:



Presionemos 2 en esta ocasión

AUTOVENTA

	Marca	Fabricacion	Color	Precio	Estado
1	Bmw	2000	Azul	40000	Disponible
2	Changan	2016	Verde	25000	Disponible
3	Delorian	1981	Plateado	99999999999	Disponible
4	Kia	2020	Azul	3000	Disponible
5	Kia	2020	Rosa	30000	Disponible
6	Kia	2020	Azul	30000	Disponible
7	Kia	2003	Morado	7000	Disponible
8	Lamborghini veneno	2019	Azul relampago	90000	Disponible
9	Mercedes benz	2020	Azul	9999999	Disponible
10	Mercedes	2019	Rosa	30000	Disponible
11	Mercedes	2019	Rojo	30000	Disponible
12	Mercedes	2022	Rosado	90000	Disponible
13	Mercedes	2022	Rosa	9900000	Disponible
14	Nisan	2022	Plateado	4000000	Disponible
15	Panama_motors	2022	Verde	900	Disponible
16	Tesla	4000	Rojo	4000	Disponible
17	Toyota corola	2077	Azul	30000	Disponible18
19	Toyota	2019	Rosa	30000	Disponible
20	Toyota	2020	Azul	30000	Disponible

Presiona enter para continuar...

como podemos ver, la opción 2 nos muestra de una manera ordenada los autos disponibles incluyendo nuestra maravillosa función: “enter_continuar” la cual nos lleva al menú de opciones nuevamente:

```
=====
=====
Venta de autos:
=====
=====

1. Registrar auto
2. Autos disponibles
3. Comprar autos
4. Autos vendidos
5. Cantidad total en S/ de autos vendidos
6. Salir

opcion: |
```

Presionamos la opción 3:

```
=====
=====
Compra de autos:
=====
=====
Introduce la marca del auto que desees comprar: Toyota|
```

	Marca	Fabricacion	Color	Precio	Estado
19	Toyota	2019	Rosa	30000	Disponible
20	Toyota	2020	Azul	30000	Disponible

Introduce el año de fabricación del auto que deseas comprar: 2019

si presionamos esta opción nos aparecerá una lista actualizada de los autos disponibles cambiando el que compramos por “vendido”

	Marca	Fabricacion	Color	Precio	Estado
1	Bmw	2000	Azul	40000	Disponible
2	Changan	2016	Verde	25000	Disponible
3	Delorian	1981	Plateado	9999999999	Disponible
4	Kia	2020	Azul	3000	Disponible
5	Kia	2020	Rosa	30000	Disponible
6	Kia	2020	Azul	30000	Disponible
7	Kia	2003	Morado	7000	Disponible
8	Lamborghini veneno	2019	Azul relampago	90000	Disponible
9	Mercedes benz	2020	Azul	9999999	Disponible
10	Mercedes	2019	Rosa	30000	Disponible
11	Mercedes	2019	Rojo	30000	Disponible
12	Mercedes	2022	Rosado	90000	Disponible
13	Mercedes	2022	Rosa	9900000	Disponible
14	Nisan	2022	Plateado	4000000	Disponible
15	Panama_motors	2022	Verde	900	Disponible
16	Tesla	4000	Rojo	4000	Disponible
17	Toyota corola	2077	Azul	30000	Disponible18
18	Toyota	2019	Rosa	30000	Disponible
19	Toyota	2020	Azul	30000	Vendido

Presiona enter para continuar...|

“enter_continuar” nuevamente nos lleva a nuestro menú de opciones:

```

=====
=====
Venta de autos:
=====
=====

1. Registrar auto
2. Autos disponibles
3. Comprar autos
4. Autos vendidos
5. Cantidad total en S/ de autos vendidos
6. Salir

opcion: |

```

Si presionamos la opción 4 el programa nos mostrará un historial de los automóviles vendidos

```

Automoviles vendidos:

+----+-----+-----+-----+-----+-----+
|    | Marca | Fabricacion | Color | Precio | Estado |
+----+-----+-----+-----+-----+-----+
| 1  | Toyota | 2020       | Azul  | 30000  | Vendido |
+----+-----+-----+-----+-----+-----+

Presiona enter para continuar...

```

“enter_continuar” nuevamente nos lleva a nuestro menú de opciones:

```
=====
=====
Venta de autos:
=====
=====

1. Registrar auto
2. Autos disponibles
3. Comprar autos
4. Autos vendidos
5. Cantidad total en S/ de autos vendidos
6. Salir

opcion: |
```

Presionamos la opción 5:

```
Total en S/ de autos vendidos
El total de autos vendidos es: 1
Cantidad en soles de autos vendidos: S/30000
Presiona enter para continuar...|
```

Esta opción nos muestra la cantidad en soles de los autos vendidos y la cantidad de ellos

“enter_continuar” nuevamente nos lleva a nuestro menú de opciones:


```
=====
=====
Venda de autos:
=====
=====

1. Registrar auto
2. Autos disponibles
3. Comprar autos
4. Autos vendidos
5. Cantidad total en S/ de autos vendidos
6. Salir

opcion: |
```

Cuando presionamos salir el programa finaliza

5.-Conclusiones y recomendaciones:

- El software diseñado es ideal para una compañía tipo MYPE (micro y pequeñas empresas) que se dedique al rubro de la compra y venta de automóviles. Puesto que el sistema cuenta con una interfaz minimalista y efectiva. Se recomienda al usuario no borrar ni modificar el formato predeterminado del *archivo.txt*, ya que es donde todos los datos se encuentran almacenados.
- Con el uso de las librerías **pandas** y **csv** podríamos hacer que el código funcione de una manera más eficiente. Se recomienda elaborar una función que filtre o vea de una manera de contar y registrar si se tiene la existencia de un auto con las misma características.

- Se podría utilizar una base de datos actualizada o una base de datos en la nube para evitar pérdidas locales y optimizar la velocidad de ejecución del software. Guardar los datos en un .txt los hace inseguros ya que no hay una copia de seguridad en caso de borre.
- Se podría hacer una interfaz gráfica para mejorar la experiencia con el usuario. Aún así la interfaz mostrada no es mala en lo absoluto, sino que se puede mejorar la experiencia usuario a computador.

1. Referencias bibliográficas:

- Gerencia de Estudios Económicos de la AAP. (2022, mayo). INFORME DEL SECTOR AUTOMOTOR ABRIL 2022.
<https://aap.org.pe/informes-estadisticos/abril-2022/Informe-Abril-2022.pdf>
- Anónimo. (2022). Flujo vehicular por unidades de peaje. 12/06/22, de INEI Sitio web:
https://www.inei.gob.pe/media/MenuRecursivo/boletines/informe-flujo-vehicular_1.pdf
- Anónimo. (2022). Venta e Inmatriculación de Vehículos Nuevos 2022. 12/06/22, de AAP Sitio web:
https://aap.org.pe/estadisticas/ventas_inmatriculaciones_vehiculos_nuevos/venta-e-inmatriculacion-de-vehiculos-nuevos-2022/