CAMILO PALACIO MESA

PROGRAMACION ORIENTADA A OBJETOS

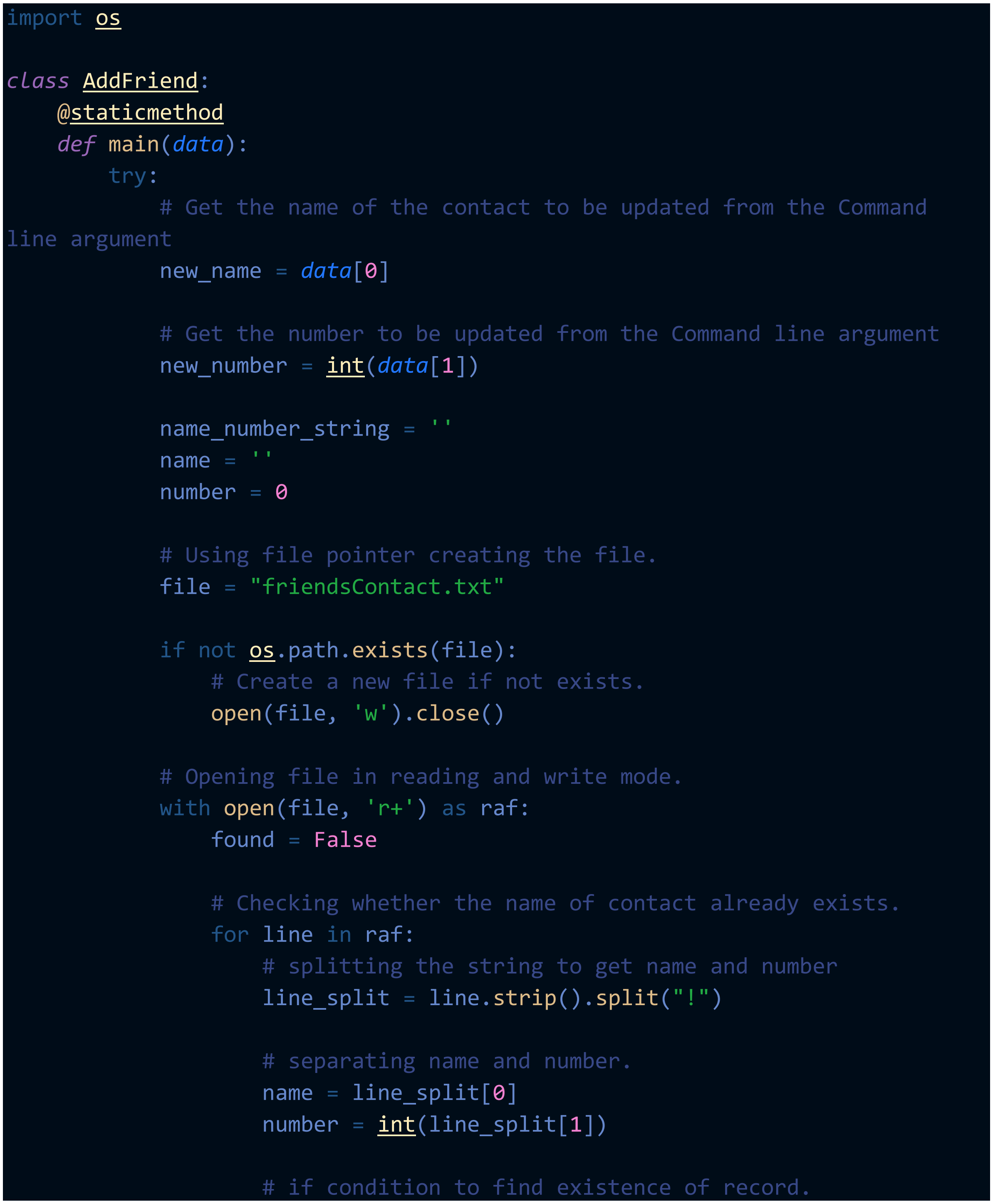
PROFESOR: WALTER HUGO ARBOLEDA MAZO

CRUD

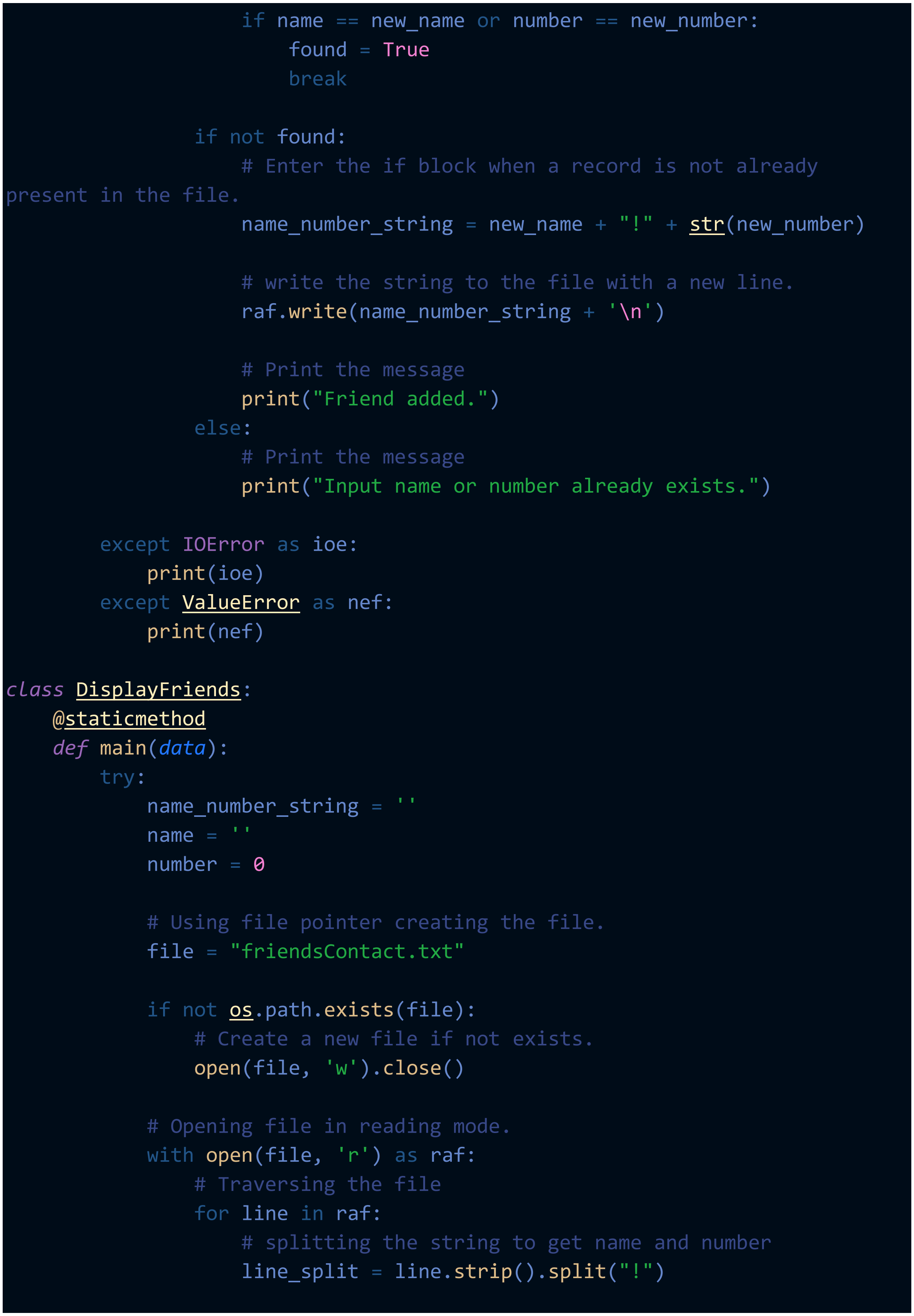
UNIVERSIDAD NACIONAL DE COLOMBIA

MEDELLIN

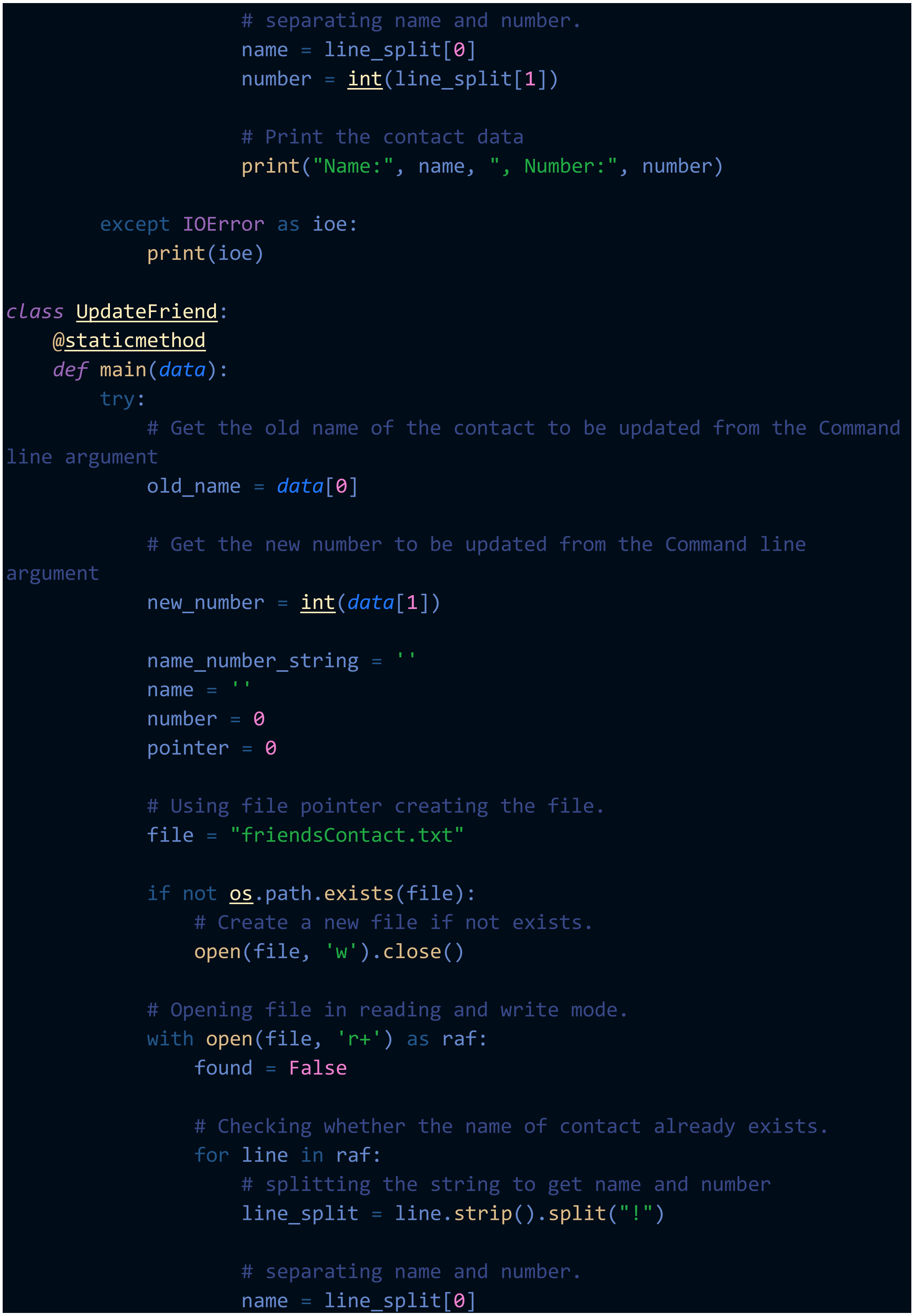
NOTA: Primeramente, el código proporcionado en 'Material guía: https://www.geeksforgeeks.org/file-handling-in-java-with-crudoperations/' fue traducido tal cual está a Python a través de la pagina: https://you.com/code?utm_source=youtube&utm_medium=video&utm_content=linktree&utm_campaign=channel&cfr=codeanything



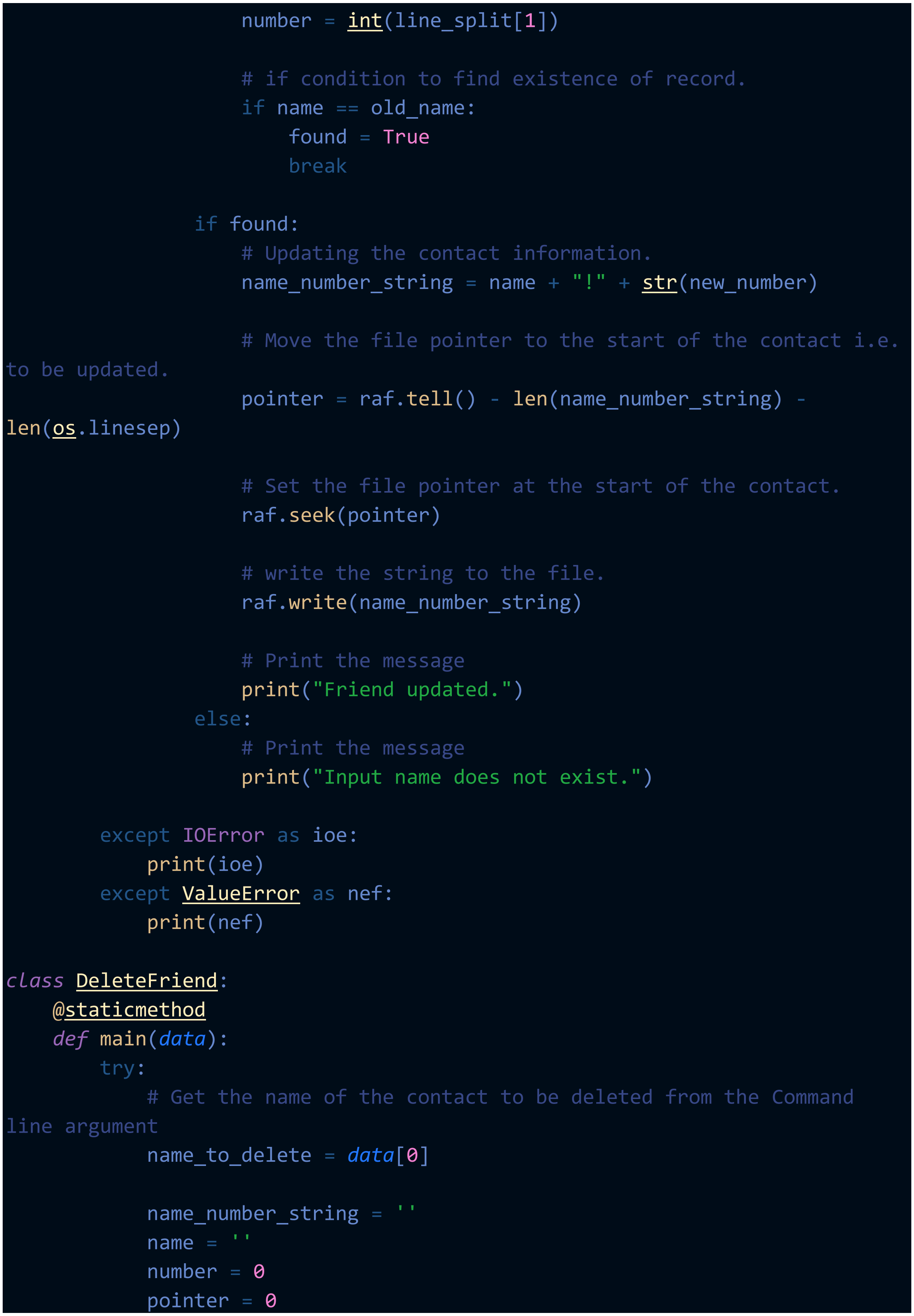
```
import os

class AddFriend:
    @staticmethod
    def main(data):
        try:
            # Get the name of the contact to be updated from the Command
line argument
            new_name = data[0]

            # Get the number to be updated from the Command line argument
            new_number = int(data[1])

            name_number_string = ''
            name = ''
            number = 0

            # Using file pointer creating the file.
            file = "friendsContact.txt"

            if not os.path.exists(file):
                # Create a new file if not exists.
                open(file, 'w').close()

            # Opening file in reading and write mode.
            with open(file, 'r+') as raf:
                found = False

                # Checking whether the name of contact already exists.
                for line in raf:
                    # splitting the string to get name and number
                    line_split = line.strip().split("!")

                    # separating name and number.
                    name = line_split[0]
                    number = int(line_split[1])

                    # if condition to find existence of record.
```



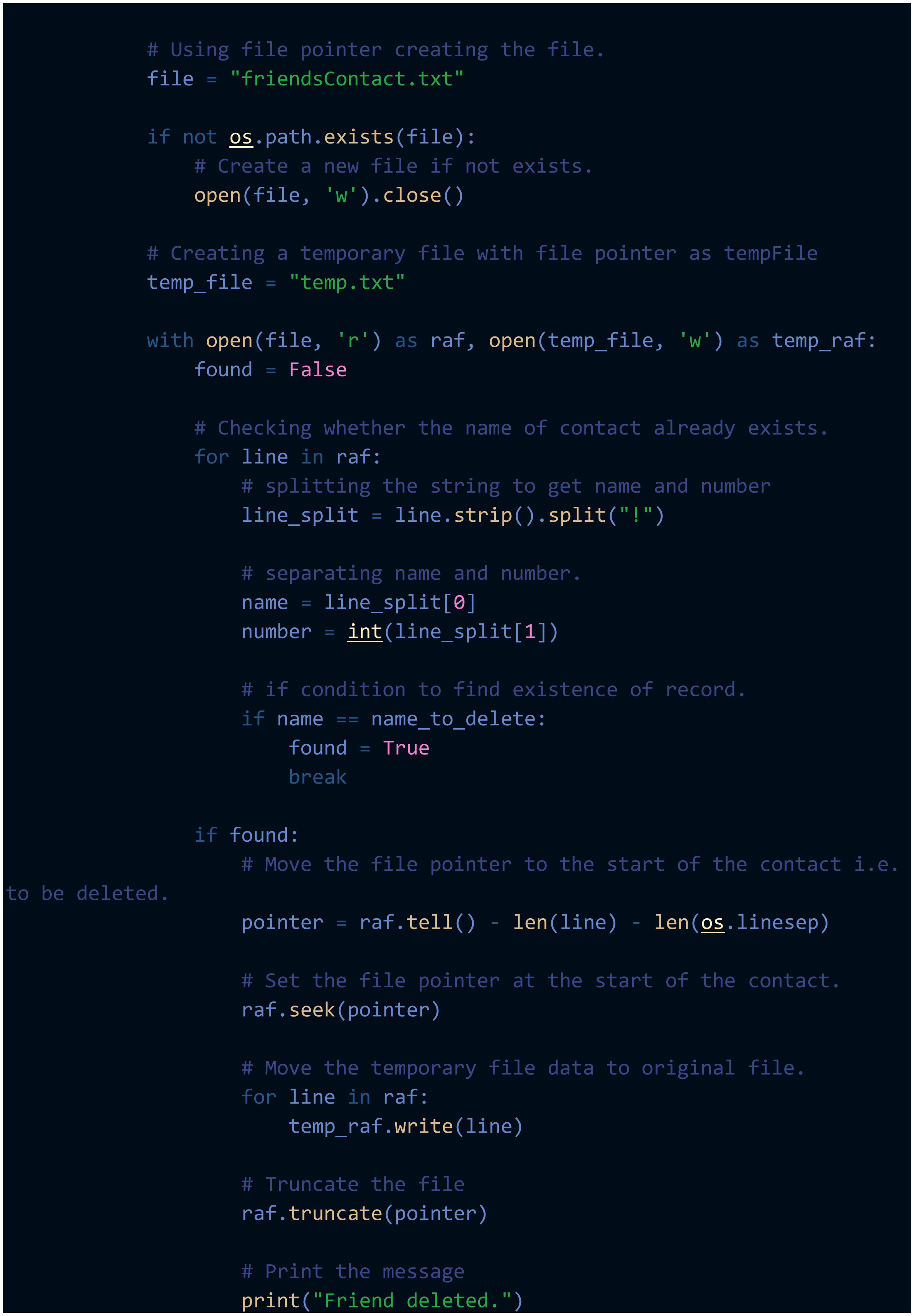
```
                    if name == new_name or number == new_number:
                        found = True
                        break

                if not found:
                    # Enter the if block when a record is not already
present in the file.
                    name_number_string = new_name + "!" + str(new_number)

                    # write the string to the file with a new line.
                    raf.write(name_number_string + '\n')

                    # Print the message
                    print("Friend added.")
                else:
                    # Print the message
                    print("Input name or number already exists.")

        except IOError as ioe:
            print(ioe)
        except ValueError as nef:
            print(nef)

class DisplayFriends:
    @staticmethod
    def main(data):
        try:
            name_number_string = ''
            name = ''
            number = 0

            # Using file pointer creating the file.
            file = "friendsContact.txt"

            if not os.path.exists(file):
                # Create a new file if not exists.
                open(file, 'w').close()

            # Opening file in reading mode.
            with open(file, 'r') as raf:
                # Traversing the file
                for line in raf:
                    # splitting the string to get name and number
                    line_split = line.strip().split("!")
```



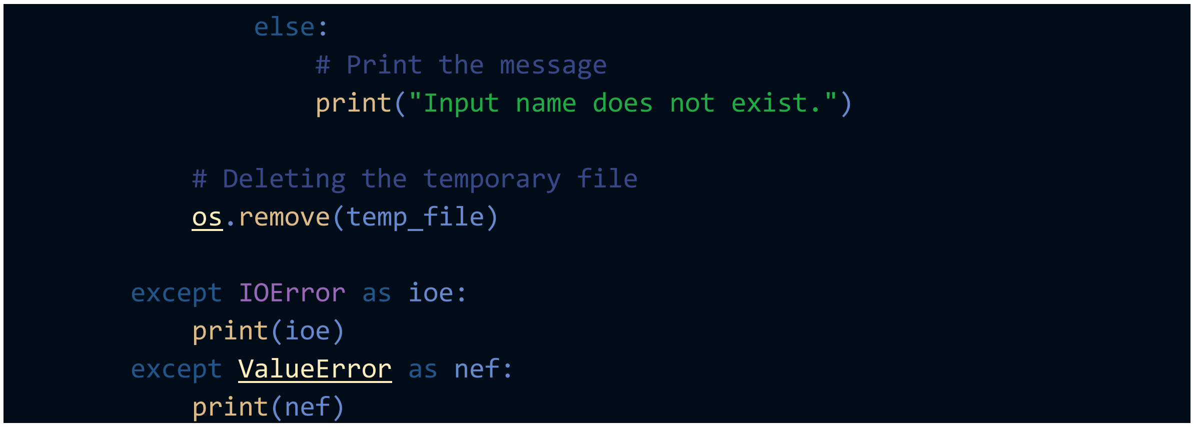
```
                    # separating name and number.
                    name = line_split[0]
                    number = int(line_split[1])

                    # Print the contact data
                    print("Name:", name, ", Number:", number)

        except IOError as ioe:
            print(ioe)

class UpdateFriend:
    @staticmethod
    def main(data):
        try:
            # Get the old name of the contact to be updated from the Command
line argument
            old_name = data[0]

            # Get the new number to be updated from the Command line
argument
            new_number = int(data[1])

            name_number_string = ''
            name = ''
            number = 0
            pointer = 0

            # Using file pointer creating the file.
            file = "friendsContact.txt"

            if not os.path.exists(file):
                # Create a new file if not exists.
                open(file, 'w').close()

            # Opening file in reading and write mode.
            with open(file, 'r+') as raf:
                found = False

                # Checking whether the name of contact already exists.
                for line in raf:
                    # splitting the string to get name and number
                    line_split = line.strip().split("!")

                    # separating name and number.
                    name = line_split[0]
```



```
                    number = int(line_split[1])

                    # if condition to find existence of record.
                    if name == old_name:
                        found = True
                        break

                if found:
                    # Updating the contact information.
                    name_number_string = name + "!" + str(new_number)

                    # Move the file pointer to the start of the contact i.e.
to be updated.
                    pointer = raf.tell() - len(name_number_string) -
len(os.linesep)

                    # Set the file pointer at the start of the contact.
                    raf.seek(pointer)

                    # write the string to the file.
                    raf.write(name_number_string)

                    # Print the message
                    print("Friend updated.")
                else:
                    # Print the message
                    print("Input name does not exist.")

        except IOError as ioe:
            print(ioe)
        except ValueError as nef:
            print(nef)

class DeleteFriend:
    @staticmethod
    def main(data):
        try:
            # Get the name of the contact to be deleted from the Command
line argument
            name_to_delete = data[0]

            name_number_string = ''
            name = ''
            number = 0
            pointer = 0
```



```
            # Using file pointer creating the file.
            file = "friendsContact.txt"

            if not os.path.exists(file):
                # Create a new file if not exists.
                open(file, 'w').close()

            # Creating a temporary file with file pointer as tempFile
            temp_file = "temp.txt"

            with open(file, 'r') as raf, open(temp_file, 'w') as temp_raf:
                found = False

                # Checking whether the name of contact already exists.
                for line in raf:
                    # splitting the string to get name and number
                    line_split = line.strip().split("!")

                    # separating name and number.
                    name = line_split[0]
                    number = int(line_split[1])

                    # if condition to find existence of record.
                    if name == name_to_delete:
                        found = True
                        break

                if found:
                    # Move the file pointer to the start of the contact i.e.
to be deleted.
                    pointer = raf.tell() - len(line) - len(os.linesep)

                    # Set the file pointer at the start of the contact.
                    raf.seek(pointer)

                    # Move the temporary file data to original file.
                    for line in raf:
                        temp_raf.write(line)

                    # Truncate the file
                    raf.truncate(pointer)

                    # Print the message
                    print("Friend deleted.")
```



```
                else:
                    # Print the message
                    print("Input name does not exist.")

            # Deleting the temporary file
            os.remove(temp_file)

        except IOError as ioe:
            print(ioe)
        except ValueError as nef:
            print(nef)
```

Luego, será subido a este mismo brach en github el código con interfaz grafica en formato .py