Laboratory Activity 3	
Polymorphism	
Course Code: CPE009B	**Program**: BSCPE
Course Title: Object-Oriented Programming 2	**Date Performed:** 9/30/24
Section: CPE21S4	**Date Submitted:** 9/30/24
Name(s): Santos, Andrei R.	**Instructor:** Professor Maria Rizette Sayo

5. Procedure

```python
class FileReaderWriter():
    def read(self):
        print("This is the default read method")

    def write(self):
        print("This is the default write method")
```

```python
from FileReaderWriter import FileReaderWriter
import csv

class CSVFileReaderWriter(FileReaderWriter):
    def read(self, filepath):
        with open(filepath, newline = '') as csvfile:
            data = csv.reader(csvfile, delimiter = ',', quotechar = '|')
            for row in data:
                print(row)
            return data

    def write(self, filepath, data):
        with open(filepath, 'w', newline = '') as csvfile:
            writer = csv.writer(csvfile, delimiter = ',',
                                quotechar = '|', quoting = csv.QUOTE_MINIMAL)
            writer.writerow(data)
```

```python
from FileReaderWriter import FileReaderWriter
import json

class JSONFileReaderWriter(FileReaderWriter):
    def read(self, filepath):
        with open(filepath, "r") as read_file:
            data = json.load(read_file)
            print(data)
            return data

    def write(self, filepath, data):
        with open(filepath, "w") as write_file:
            json.dump(obj = data, fp = write_file)
```

```
1. Apple, Banana, Mango, Orange, Cherry
```

```
{
    "description" : "This is a JSON Sample",
    "accounts" :
    [
        {"id" : 1, "name":"Jack"},
        {"id" : 2, "name":"Rose"}
    ]
}

```

```
from FileReaderWriter import FileReaderWriter
from CSVFileReaderWriter import CSVFileReaderWriter
from JSONFileReaderWriter import JSONFileReaderWriter

#Test the default class
df = FileReaderWriter()
df.read()
df.write()

#Test the polymorphed methods
c = CSVFileReaderWriter()
c.read("sample.csv")
c.write(filepath = "sample2.csv", data = ["Hello", "World"])

j = JSONFileReaderWriter()
j.read("sample.json")
j.write(data = ['foo', {'bar' : ('baz', None, 1.0, 2)}], filepath = "sample2.json")
```

Output:

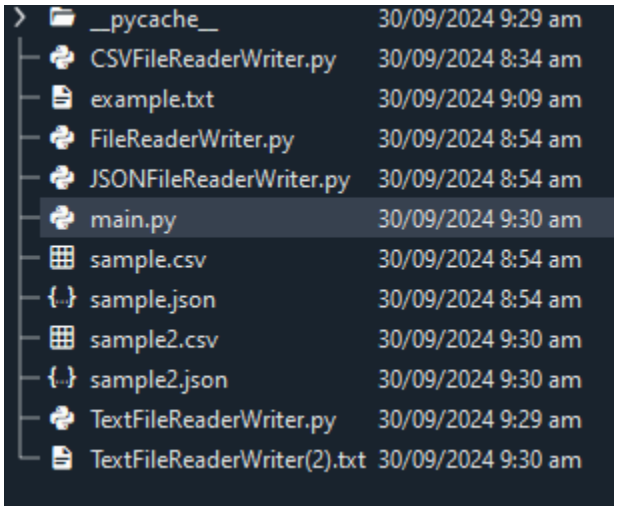
```
This is the default read method
This is the default write method
['1. Apple', ' Banana', ' Mango', ' Orange', ' Cherry']
[]
{'description': 'This is a JSON Sample', 'accounts': [{'id': 1, 'name': 'Jack'}, {'id': 2,
'name': 'Rose'}]}
```

6. Supplementary Activity

```python
from FileReaderWriter import FileReaderWriter

class TextFileReaderWriter(FileReaderWriter):
    def read(self, filepath):
        with open(filepath, newline = '') as read_file:
            read_file.read()

    def write(self, filepath, data):
        with open(filepath, 'w', newline = '') as write_file:
            write_file.write(data)
```




```
This is the default read method
This is the default write method
['1. Apple', ' Banana', ' Mango', ' Orange', ' Cherry']
[]
{'description': 'This is a JSON Sample', 'accounts': [{'id': 1, 'name': 'Jack'}, {'id': 2,
'name': 'Rose'}]}
```

(Note: I didn't add the print() that is why it has the same output.)
(Note: The new file was created, shown in github)

Questions

1. **Why is polymorphism important?**

Polymorphism is essential in object-oriented programming as it allows methods on objects of different classes through a unified interface like having different forms into one. This shows more flexibility and reusability, enabling programmers to implement new functionalities without altering existing or previous code. With this consistent program, polymorphism enhances and makes it easy to understand and maintain complex codes.

2. **Explain the advantages and disadvantages of using applying Polymorphism in an Object-Oriented Programming.**

The advantage of polymorphism is the increased code flexibility, allowing for different implementations to be used without changing the code that uses them. This shows code reuse, making it easier to add new codes or modify existing codes without too many changes. Inshort, polymorphism simplifies its maintenance and errors. While the disadvantages are the potential for increased complexity, which can make the code harder to read and understand, especially for new programmers. Runtime errors may occur if method calls are wrong called, that is why we need to call each of them to let the code run smoothly.

3. **What maybe the advantages and disadvantages of the program we wrote to read and write csv and json files?**

The advantages of a program designed to read and write CSV and JSON files include flexibility in data interchange formats and ease of use, as both formats are known for its widely usability. But potential disadvantages may involve differences in data types, which can complicate the implementation. Also, performance needs to be considered since it requires reading or run complex codes.

4. **What maybe considered if Polymorphism is to be implemented in an Object-Oriented Program?**

Considerations for Implementing Polymorphism when implementing polymorphism in an object-oriented program, to make clear, neat, and easy to understand code that all subclasses will follow. We should also consider how different classes will interact and ensure that their implementations align with the intended, needed, or required polymorphic methods.

5. **How do you think Polymorphism is used in an actual programs that we use today**

We can use polymorphism in graphical user interfaces (GUIs), where various components can be used to show the user its output design for better understanding for the user to interact.. Like creating a single GUI that can manage clicks on different buttons while allowing each button to execute its unique behavior. We can also say that, polymorphism is used in frameworks and libraries, enabling developers to extend functionalities without changing the main code.

7. Conclusion

In conclusion, I have learned how to use Polymorphism wherein it can form in different forms and also to apply it in different classes or codes. By running the procedural activity and also the supplementary activity that we apply the codes can be reused and also flexible when collecting data and making it a little easy, while there is a disadvantage is when the code becomes complex, it will be harder to understand if we are maintaining the base code. Lastly, we can apply this learning in the future code like the GUI wherein we will tackle it for upcoming lessons.

8. Assessment Rubric