Lengkapi Code dibawah ini

Tugas Mg-09 PBO

Student   : Reza Chairul Manam, `reza.120140086@student.itera.ac.id`
Student ID Number : 120140086
Class   : PBO-RC
Lecturer   : I Wayan Wiprayoga Wisesa S.Kom., M.Kom, `wayan.wisesa@if.itera.ac.id`

Listing 1: Tugas-1 Mg-09 PBO-RC — Lengkapi Script dibawah ini

```python
#Tugas-1 Mg-09
from abc import _____, _____

# --- Abstract Base Class ---
class LibraryItem(_____):
    def __init__(self, title: str, item_id: str):
        self._____ = title  # Private attribute
        self._____ = item_id  # Private attribute
        self.checked_out = False

    @property
    def title(self):
        return self._____

    @property
    def item_id(self):
        return self._____

    @abstractmethod
    def _____(self):  # Abstract method for checkout
        pass

    @abstractmethod
    def return_item(self):
        pass

    def _____(self) -> str:  # String representation
        return f"{self.__class__.__name__}: {self.title} (ID: {self.
            item_id})"

# --- Book Subclass ---
class Book(LibraryItem):
    def __init__(self, title: str, item_id: str, author: str):
        _____._____(self, title, item_id)
        self._____ = author  # Private attribute

    @property
    def author(self):
        return self._____

    def check_out(self):
        if not self.checked_out:
            self.checked_out = _____
```

```python
                print(f"Book '{self.title}' checked out.")
            else:
                print("Book already checked out.")

    def return_item(self):
        if self.checked_out:
            self.checked_out = _____
            print(f"Book '{self.title}' returned.")

# --- DVD Subclass ---
class DVD(_____):
    def __init__(self, title: str, item_id: str, duration: int):
        super().__init__(title, item_id)
        self._____ = duration   # Private attribute

    def check_out(self):
        if not self.checked_out:
            print(f"DVD '{self.title}' checked out for {self.__duration}
                minutes.")
            self.checked_out = True
        else:
            print("DVD already checked out.")

    def _____(self):
        return f"DVD: {self.title} (Duration: {self.__duration} mins)"

# --- Library Class (Composition) ---
class Library:
    def __init__(self):
        self._____ = []   # List of LibraryItem objects

    def _____(self, item: LibraryItem):
        self.items.append(item)
        print(f"Added {item.title} to the library.")

    def search_by_title(self, title: str) -> list:
        return [item for item in self.items if _____.lower() in item.title.
            lower()]

    def _____(self) -> int:   # Total items in library
        return len(_____)

    def __add__(self, other: 'Library') -> 'Library':
        new_library = Library()
        new_library.items = _____ + _____
        return new_library

# --- Test Code ---
if __name__ == "__main__":
    book1 = Book("The Python Guide", "BK001", "Alice
        Smith")
    dvd1 = DVD("Learn OOP in 30 Days", "DVD001", 120)
```

```
    library = Library()
    library.add_item(book1)
    library._____(dvd1)

    book1.check_out()
    dvd1.check_out()
    dvd1._____()

    print(f"Total items: {library.total_items()}")
```

Jawaban Isian Tugas-1 Mg-09 PBO-RC

1. `ABC, abstractmethod`

2. `ABC`

3. `__title`

4. `__item_id`

5. `__title`

6. `__item_id`

7. `check_out`

8. `__str__`

9. `super().__init__`

10. `__author`

11. `__author`

12. `True`

13. `False`

14. `LibraryItem`

15. `__duration`

16. `__str__`

17. `items`

18. `add_item`

19. `title`

20. `self.items`

21. `self.items, other.items`

22. `add_item`

23. `__str__`

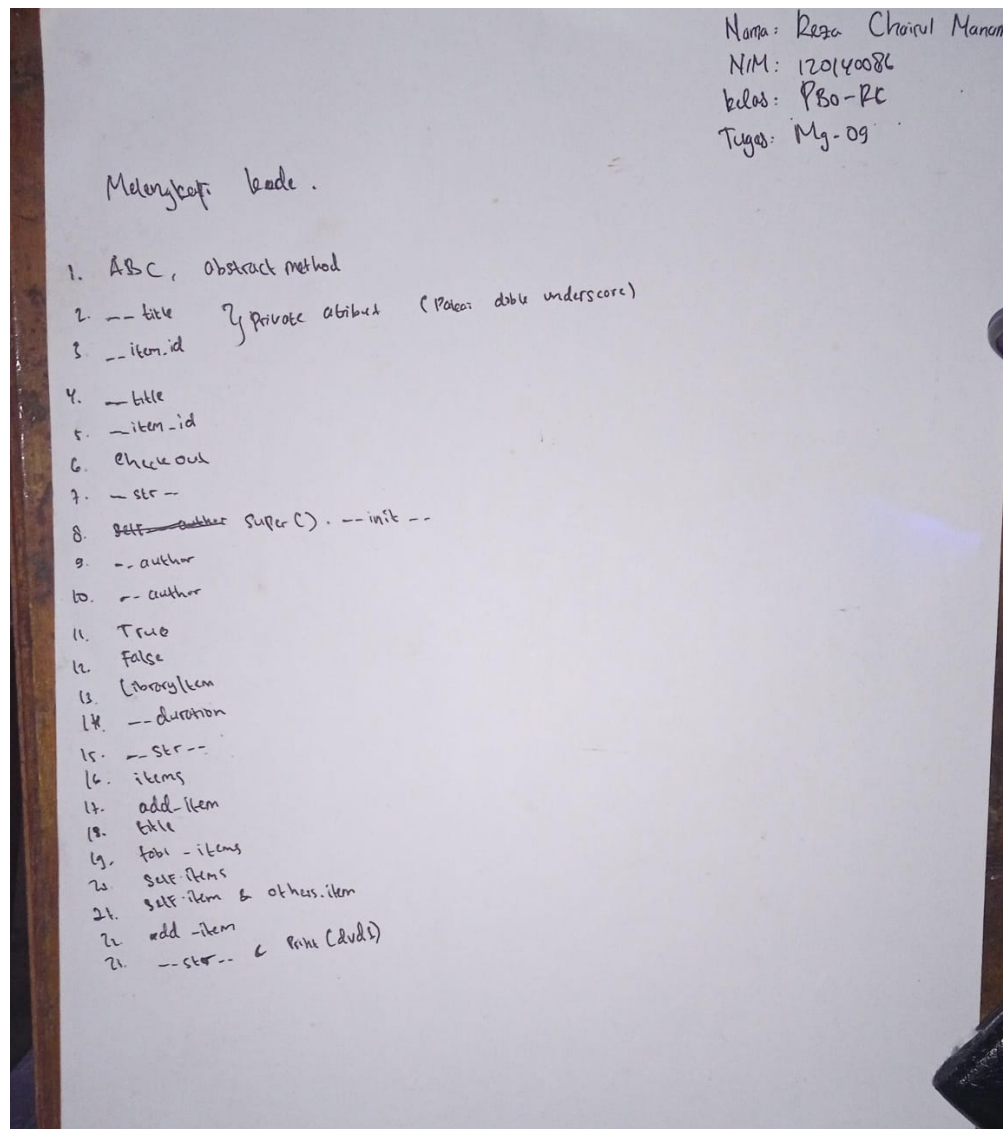## Foto Tulisan Tangan di Kertas

Figure 1: Jawaban ditulis tangan di kertas

Link Source Code

Source code lengkap di tautan berikut: github.com/rezachairul/Latihan-PBO-RC-2025