CLIENT DISPATCHER DESIGN PATTERN

Chethan kumar K M

PES1PG23CA329

1. Dispatcher Class

- The `Dispatcher` class is responsible for managing handlers and dispatching commands to appropriate handlers based on user input.

- It has methods `add_handler()` to add handlers for different commands and `dispatch()` to dispatch commands to the corresponding handler.

2. Handler Class

- The `Handler` class is an abstract class defining the interface for handling different commands. It includes a `handle()` method which is overridden by concrete handler classes.

3. Concrete Handler Classes

- Three concrete handler classes are implemented: `BookCabHandler`, `CancelBookingHandler`, and `ViewBookingHandler`.

- Each handler class inherits from the `Handler` class and implements the `handle()` method according to its functionality.

Code

```
class Dispatcher:
    def __init__(self):
        self.handlers = {}

    def add_handler(self, command, handler):
        self.handlers[command] = handler

    def dispatch(self, command):
        handler = self.handlers.get(command)
        if handler:
            handler.handle()
        else:
            print("Invalid option. Please try again.")


class Handler:
    def handle(self):
```

```
        raise NotImplementedError("Subclasses must implement handle
method.")



class BookCabHandler(Handler):
    def handle(self):
        print("Booking a cab...")



class CancelBookingHandler(Handler):
    def handle(self):
        print("Canceling cab booking...")



class ViewBookingHandler(Handler):
    def handle(self):
        print("Viewing cab booking...")



def main():
    dispatcher = Dispatcher()
    dispatcher.add_handler("1", BookCabHandler())
    dispatcher.add_handler("2", CancelBookingHandler())
    dispatcher.add_handler("3", ViewBookingHandler())
```

```
    print("Welcome to the Cab Booking System!")
    while True:
        print("\nOptions:")
        print("1. Book a cab")
        print("2. Cancel cab booking")
        print("3. View cab booking")
        print("4. Exit")
        choice = input("Enter your choice: ").strip()

        if choice == "4":
            print("Thank you for using the Cab Booking System.
Goodbye!")
            break
        dispatcher.dispatch(choice)


if __name__ == "__main__":
    main()
```

output

```
Welcome to the Cab Booking System!

Options:
1. Book a cab
2. Cancel cab booking
3. View cab booking
4. Exit
Enter your choice: 1
Booking a cab...

Options:
1. Book a cab
2. Cancel cab booking
3. View cab booking
4. Exit
Enter your choice: 3
Viewing cab booking...

Options:
1. Book a cab
2. Cancel cab booking
3. View cab booking
4. Exit
Enter your choice: 2
Canceling cab booking...

Options:
1. Book a cab
2. Cancel cab booking
3. View cab booking
4. Exit
Enter your choice: 4
Thank you for using the Cab Booking System.
Goodbye!
```