```
import datetime

class ExpenseTracker:
    def __init__(self):
        self.expenses = []

    def add_expense(self, amount, category, description=""):
        # Validation: amount must be positive
        if amount <= 0:
            print("❌ Invalid amount. Please enter a positive number.")
            return

        # Prevent duplicate category + description
        for exp in self.expenses:
            if exp["category"].lower() == category.lower() and exp["description"].lower() ==
description.lower():
                print(f"❌ Expense with category '{category}' and description '{description}' already
exists.")
                return

        expense = {
            "amount": amount,
            "category": category,
            "description": description,
            "date": datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")
        }
        self.expenses.append(expense)
        print(f"✅ Added: {amount} in {category} - {description}")

    def view_expenses(self):
        if not self.expenses:
            print("No expenses recorded yet.")
            return
        print("\n--- Expense List ---")
        for i, exp in enumerate(self.expenses, start=1):
            print(f"{i}. {exp['date']} | {exp['category']} | ₱{exp['amount']} | {exp['description']}")

    def total_expenses(self):
        total = sum(exp["amount"] for exp in self.expenses)
        print(f"\n💰 Total Expenses: ₱{total}")
        return total

    def filter_by_category(self, category):
        filtered = [exp for exp in self.expenses if exp["category"].lower() == category.lower()]
        if not filtered:
            print(f"No expenses found in category: {category}")
            return
        print(f"\n--- Expenses in {category} ---")
        for exp in filtered:
            print(f"{exp['date']} | ₱{exp['amount']} | {exp['description']}")

    def update_expense(self, category, description, new_amount):
        """Update the amount of an existing expense by category + description."""
        for exp in self.expenses:
            if exp["category"].lower() == category.lower() and exp["description"].lower() == description.lower():
                if new_amount <= 0:
                    print("❌ Invalid amount. Please enter a positive number.")
                    return
                old_amount = exp["amount"]
                exp["amount"] = new_amount
                print(f"🔄 Updated: {category} - {description} | ₱{old_amount} → ₱{new_amount}")
                return
        print("❌ Expense not found. Please check category and description.")

# User Input
if __name__ == "__main__":
    tracker = ExpenseTracker()

    while True:
        print("\n📊 Expense Tracker Menu")
        print("1. Add Expense")
        print("2. View Expenses")
        print("3. Show Total / Update Expense")
        print("4. Filter by Category")
        print("5. Exit")

        choice = input("Choose an option (1-5): ")

        if choice == "1":
            try:
                amount = float(input("Enter amount: "))
            except ValueError:
                print("❌ Invalid input. Please enter a number.")
                continue
            category = input("Enter category: ")
            description = input("Enter description: ")
            tracker.add_expense(amount, category, description)

        elif choice == "2":
            tracker.view_expenses()

        elif choice == "3":
            tracker.total_expenses()
            update_choice = input("Do you want to update an expense? (y/n): ").lower()
            if update_choice == "y":
                category = input("Enter category of expense to update: ")
                description = input("Enter description of expense to update: ")
                try:
                    new_amount = float(input("Enter new amount: "))
                except ValueError:
                    print("❌ Invalid input. Please enter a number.")
                    continue
                tracker.update_expense(category, description, new_amount)

        elif choice == "4":
            category = input("Enter category to filter: ")
            tracker.filter_by_category(category)

        elif choice == "5":
            print("👋 Goodbye!")
            break
```

```
else:
    print("❌ Invalid choice, try again.")
```