```
# This line imports a library that helps us work with CSV files (like spreadsheets).
import csv

# These are global variables.  They store information that the whole program can use.
expenses = []  # This is a list where we'll store all the expenses.
monthly_budget = 0  # This is where we'll store the user's monthly budget.
filename = "expenses.csv"  # This is the name of the file where we'll save the expenses.

# This function lets the user add a new expense.
def add_expense():
    # Ask the user for the date of the expense.
    date = input("Enter date (YYYY-MM-DD): ")
    # Ask the user for the category of the expense.
    category = input("Enter category: ")
    # Ask the user for the amount of the expense.
    amount = float(input("Enter amount: "))  # The `float()`  function converts the input to a number with decimals.
    # Ask the user for a short description of the expense.
    description = input("Enter description: ")

    # Create a dictionary to store the expense information.  A dictionary is like a list, but it uses names to access the values.
    expense = {
        'date': date,
        'category': category,
        'amount': amount,
        'description': description
    }

    # Add the expense to our list of expenses.
    expenses.append(expense)
    # Tell the user that the expense was added.
    print("Expense added!")

# This function lets the user see all the expenses they've added.
def view_expenses():
    # First, check if there are any expenses in the list.
    if not expenses:
        # If there are no expenses, tell the user.
        print("No expenses yet.")
    else:
        # If there are expenses, loop through each one...
        for expense in expenses:
            # ...and print out the information for each expense.
            print(f"Date: {expense['date']}, Category: {expense['category']}, "
                  f"Amount: ${expense['amount']:.2f}, Description: {expense['description']}") # the :.2f is the decimal places


# This function lets the user set their monthly budget.
def set_budget():
    # We need to tell Python that we want to change the global variable `monthly_budget`.
    global monthly_budget
    # Ask the user for their monthly budget.
    monthly_budget = float(input("Enter monthly budget: $"))
    # Tell the user what their budget is.
    print(f"Budget set to ${monthly_budget:.2f}")


# This function lets the user track their budget.
def track_budget():
    # First, check if the user has set a budget yet.
    if monthly_budget == 0:
        # If the budget is 0, tell the user to set a budget first.
        print("Please set a budget first.")
    else:
        # If the budget is set, calculate the total amount spent.
        total_spent = 0
        for expense in expenses:
            total_spent = total_spent + expense['amount']
        # Calculate the remaining balance.
        remaining = monthly_budget - total_spent
        # Print out the total amount spent, the budget, and the remaining balance.
        print(f"Total spent: ${total_spent:.2f}")
        print(f"Budget: ${monthly_budget:.2f}")
        print(f"Remaining: ${remaining:.2f}")
        # Check if the user is over budget.
        if remaining < 0:
            # If the user is over budget, tell them.
            print("Warning: Over budget!")

# This function saves the expenses to a file.
def save_expenses():
    # Open the file in "write" mode ("w").  This will create the file if it doesn't exist, or overwrite it if it does.
    with open(filename, 'w', newline='') as file:
        # Create a CSV writer object.  This helps us write data to the CSV file.
        writer = csv.DictWriter(file, fieldnames=['date', 'category', 'amount', 'description'])
        # Write the header row (the names of the columns).
        writer.writeheader()
        # Write all the expenses to the file.
        writer.writerows(expenses)
    # Tell the user that the expenses were saved.
    print("Expenses saved!")

# This function loads the expenses from the file.
def load_expenses():
    # We need to tell Python that we want to change the global variable `expenses`.
    global expenses
    # Try to open the file in "read" mode ("r").
    try:
        with open(filename, 'r') as file:
            # Create a CSV reader object.  This helps us read data from the CSV file.
            reader = csv.DictReader(file)
            # Read all the rows from the file and store them in the `expenses` list.
            expenses = list(reader)
        #Loop through each row and turn it into a float
        for expense in expenses:
            expense['amount'] = float(expense['amount'])
        # Tell the user that the expenses were loaded.
        print("Expenses loaded!")
    # If the file doesn't exist, we'll get a `FileNotFoundError`.
    except FileNotFoundError:
        # If the file doesn't exist, tell the user.
        print("No previous data found.")

# This function shows the menu and gets the user's choice.
def show_menu():
    # Print the menu options.
    print("\n--- Expense Tracker Menu ---")
    print("1. Add expense")
    print("2. View expenses")
    print("3. Set budget")
    print("4. Track budget")
    print("5. Save expenses")
    print("6. Exit")
    # Ask the user to choose an option.
    return input("Choose an option (1-6): ")

# This is the main function of the program.
def main():
    # First, load the expenses from the file.
    load_expenses()

    # Then, loop forever...
    while True:
        # ...show the menu and get the user's choice.
        choice = show_menu()

        # If the user chose option 1...
        if choice == '1':
            # ...add an expense.
            add_expense()
        # If the user chose option 2...
        elif choice == '2':
            # ...view the expenses.
            view_expenses()
        # If the user chose option 3...
        elif choice == '3':
            # ...set the budget.
            set_budget()
        # If the user chose option 4...
        elif choice == '4':
            # ...track the budget.
            track_budget()
        # If the user chose option 5...
        elif choice == '5':
            # ...save the expenses.
            save_expenses()
        # If the user chose option 6...
        elif choice == '6':
            # ...save the expenses and exit the program.
            save_expenses()
            print("Goodbye!")
            break
        # If the user chose an invalid option...
        else:
            # ...tell them to try again.
            print("Invalid choice. Try again.")

# This line tells Python to run the `main()` function when the program starts.
if __name__ == "__main__":
    main()
```