



ELECTRICAL&COMPUTER ENGINEERING DEPARTMENT

LINUX LABORATORY ENCS313

Python - Project 2

Medical Test Management System

Prepared by: Joud Thaher, Ibrahim Salahat

Id numbers: 1221381, 1221701

Instructor: Dr. Mohammad Jubran

Assistant: Eng. Ahed Mafarjeh

Section: 1

Date: 22.8.2024

Test Cases

Please note that we might give a different file name because we edited the file multiple times each one alone, in different times.

Now let's go through the test cases in our medical test management system!

First, we created the two files MedicalTest and medicalRecord as requested in the project.

Below is what it looks like after running the program:

```
Menu:
1.Add new medical test.
2.Add new medical test record.
3.Update patient record.
4.Update medical tests.
5.Delete patient record.
6.Filter medical tests.
7.Generate textual summary reports.
8.Export medical records.
9.Import medical records.
10.Print all records.
11.Exit.
Choose an option:
```

It gives the user a menu to choose the operation he wants. Let's go through the menu

1. Add a New Medical Test

- *Adding a Valid Input Test:*

```
Choose an option: 1
Enter test name: LDL
Enter range values (e.g., '> 13.8, < 17.2'): >14.8, <12.7
Enter unit: MG
Enter turnaround time (format DD-hh-mm): 00-07-08
Test added successfully.
```

- *Adding Invalid Input Tests*

```
Choose an option: 1
Enter test name: Hgb
Enter range values (e.g., '> 13.8, < 17.2'): >14.a
Invalid range value: '14.a' is not a number.
```

```
Choose an option: 1
Enter test name: LDL
Enter range values (e.g., '> 13.8, < 17.2'): >13.8
Enter unit: mg
Enter turnaround time (format DD-hh-mm): 12-99-04
Invalid turnaround time values. Please ensure hours are between 0-23 and minutes between 0-59.
```

2. Add a New Medical Test Record

- *Adding a Valid Input Test:*

```
Choose an option: 2
Adding a new test record:
Patient ID (7 digits): 1234567
Test Name (fixed length): LDL
Test Date and Time (YYYY-MM-DD HH:MM): 2020-01-04 01:03
Result Value (numeric): 14.9
Results Unit (fixed length): mg
Status (Pending, Completed, Reviewed): Pending
Test record added successfully.
```

- *Invalid Patient ID:*

```
Choose an option: 2
Adding a new test record:
Patient ID (7 digits): 1234
Invalid Patient ID. It should be exactly 7 digits.
Patient ID (7 digits): 12a3456
Invalid Patient ID. It should consist of digits only.
Patient ID (7 digits): 123a4567
Invalid Patient ID. It should consist of digits only.
Patient ID (7 digits): 12,34566
Invalid Patient ID. It should consist of digits only.
Patient ID (7 digits): 1234567
```

- *Invalid Test Date Format:*

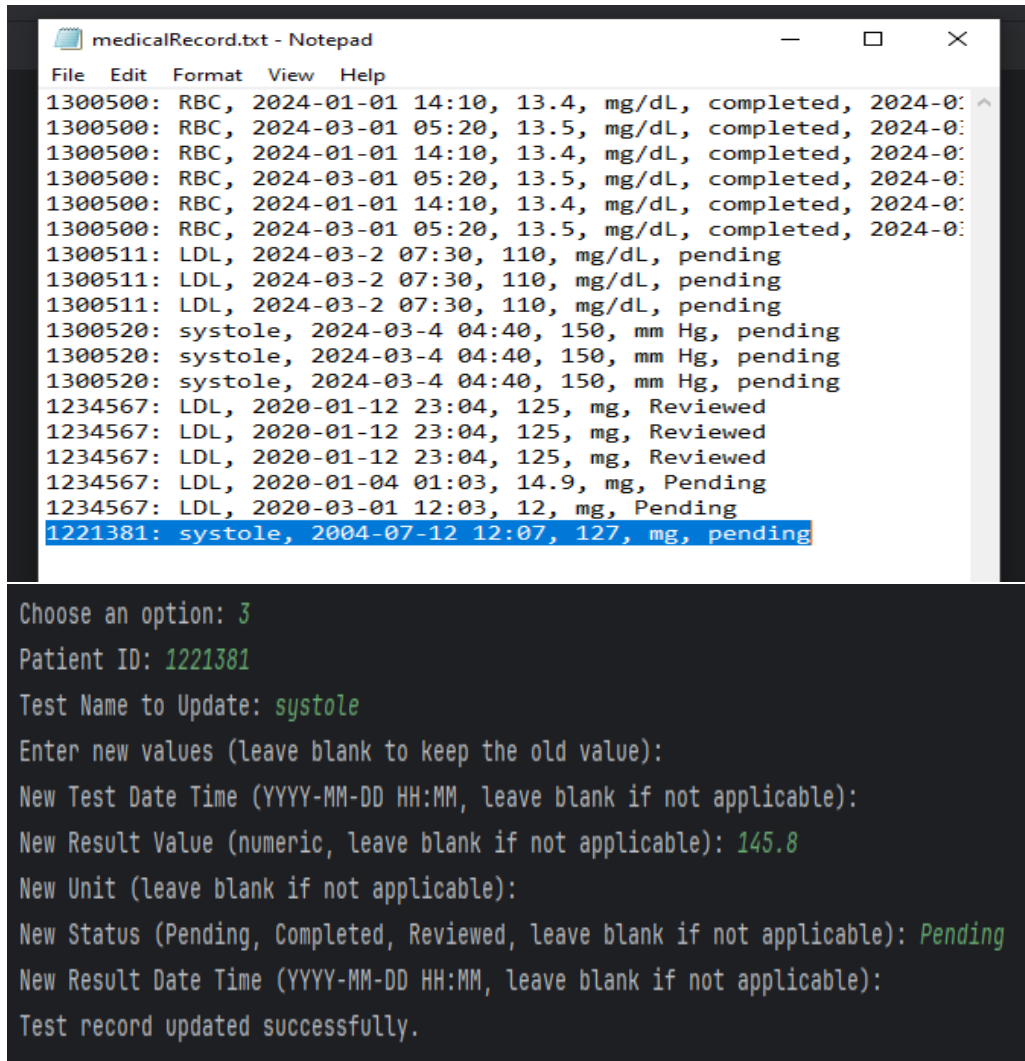
```
Choose an option: 2
Adding a new test record:
Patient ID (7 digits): 1234567
Test Name (fixed length): LDL
Test Date and Time (YYYY-MM-DD HH:MM): 2020-99-99 12:05
Invalid Date and Time. The month must be between 01 and 12.
Test Date and Time (YYYY-MM-DD HH:MM): 2020-01-99 12:05
Invalid Date and Time. The day must be valid for the given month.
Test Date and Time (YYYY-MM-DD HH:MM): 2020-01-a2 12:05
Invalid Date and Time format. Please use YYYY-MM-DD HH:MM.
```

- *Invalid Status:*

```
Status (Pending, Completed, Reviewed): joud
Invalid Status. It should be one of 'Pending', 'Completed', 'Reviewed'.
Status (Pending, Completed, Reviewed): Pending
```

3. Update Patient Record

- *Found*



The image shows a Notepad window titled "medicalRecord.txt - Notepad" with a menu bar (File, Edit, Format, View, Help). The text file contains a list of medical records. The last record, "1221381: systole, 2004-07-12 12:07, 127, mg, pending", is highlighted in blue. Below the Notepad window is a terminal window with the following text:

```
Choose an option: 3
Patient ID: 1221381
Test Name to Update: systole
Enter new values (leave blank to keep the old value):
New Test Date Time (YYYY-MM-DD HH:MM, leave blank if not applicable):
New Result Value (numeric, leave blank if not applicable): 145.8
New Unit (leave blank if not applicable):
New Status (Pending, Completed, Reviewed, leave blank if not applicable): Pending
New Result Date Time (YYYY-MM-DD HH:MM, leave blank if not applicable):
Test record updated successfully.
```

```
medicalRecord.txt - Notepad
File Edit Format View Help
1300500: RBC, 2024-01-01 14:10, 13.4, mg/dL, completed, 2024-01-01
1300500: RBC, 2024-03-01 05:20, 13.5, mg/dL, completed, 2024-03-01
1300500: RBC, 2024-01-01 14:10, 13.4, mg/dL, completed, 2024-01-01
1300500: RBC, 2024-03-01 05:20, 13.5, mg/dL, completed, 2024-03-01
1300500: RBC, 2024-01-01 14:10, 13.4, mg/dL, completed, 2024-01-01
1300500: RBC, 2024-03-01 05:20, 13.5, mg/dL, completed, 2024-03-01
1300511: LDL, 2024-03-2 07:30, 110, mg/dL, pending
1300511: LDL, 2024-03-2 07:30, 110, mg/dL, pending
1300511: LDL, 2024-03-2 07:30, 110, mg/dL, pending
1300520: systole, 2024-03-4 04:40, 150, mm Hg, pending
1300520: systole, 2024-03-4 04:40, 150, mm Hg, pending
1300520: systole, 2024-03-4 04:40, 150, mm Hg, pending
1234567: LDL, 2020-01-12 23:04, 125, mg, Reviewed
1234567: LDL, 2020-01-12 23:04, 125, mg, Reviewed
1234567: LDL, 2020-01-12 23:04, 125, mg, Reviewed
1234567: LDL, 2020-01-04 01:03, 14.9, mg, Pending
1234567: LDL, 2020-03-01 12:03, 12, mg, Pending
1221381: systole, 2004-07-12 12:07, 145.8, mg, Pending
```

- *Not Found*

```
Choose an option: 3
Patient ID: 1221701
Patient ID does not exist. Please enter a valid Patient ID.
Patient ID: |
```

4. Update Medical Tests

- *Found*

Menu:

- 1.Add new medical test.
- 2.Add new medical test record.
- 3.Update patient record.
- 4.Update medical tests.
- 5.Delete patient record.
- 6.Filter medical tests.
- 7.Generate textual summary reports.
- 8.Export medical records.
- 9.Import medical records.
- 10.Print all records.
- 11.Exit.

Choose an option:

```
medicalTest.txt - Notepad
File Edit Format View Help
Hemoglobin (Hgb); > 13.8, < 17.2; g/dL; 00-03-04
Blood Glucose Test (BGT); > 70, < 99; mg/dL; 00-12-06
LDL Cholesterol Low-Density Lipoprotein (LDL); < 100; mg/dL; 00-08-04
Systolic Blood Pressure (systole); < 120; mm Hg; 00-08-04
Diastolic Blood Pressure (diastole); < 80; mm Hg; 00-10-00
```

```
Choose an option: 4
Enter the old test name to update: Diastolic Blood Pressure (diastole)
Enter the new test name: Diastolic Blood Pressure (DBP)
Enter the new range (e.g., > 13.8, < 17.2): >90, <75
Enter the new unit: mm Hg
Enter the new turnaround time (e.g., 00-03-04): 00-01-03
Medical test updated successfully.
```

```
File Edit Format View Help
Hemoglobin (Hgb); > 13.8, < 17.2; g/dL; 00-03-04
Blood Glucose Test (BGT); > 70, < 99; mg/dL; 00-12-06
LDL Cholesterol Low-Density Lipoprotein (LDL); < 100; mg/dL; 00-08-04
Systolic Blood Pressure (systole); < 120; mm Hg; 00-08-04
Diastolic Blood Pressure (DBP); >90, <75; mm Hg; 00-01-03
```

- **Not Found**

```
Choose an option: 4
Enter the old test name to update: hemo
Enter the new test name: hemoo
Enter the new range (e.g., > 13.8, < 17.2): >90
Enter the new unit: mg
Enter the new turnaround time (e.g., 00-03-04): 00-90-1
Invalid turnaround time values. Please ensure hours are between 0-23 and minutes between 0-59.
Invalid new turnaround time. Update aborted.
```

5. Delete Patient Records

- *Delete an existing test record*

The screenshot displays the PyCharm IDE interface during the execution of a Python program. The top editor pane shows a list of medical records in `medicalRecord.txt`. The sixth record, `1221381: systole, 2004-07-12 12:07, 127, mg, pending`, is selected. The bottom editor pane shows the program's output, which includes a menu of options and the user's input to delete a record. The output shows the user selecting option 5, entering the patient ID `1221381`, the test name `systole`, the test date and time `2004-07-12 12:07`, the result value `127`, the unit `mg`, and the status `pending`. The final output message is `Record deleted successfully.`

```
1 1300500: RBC, 2024-01-01 14:10, 13.4, mg/dL, completed, 2024-01-01 15:30
2 1300500: RBC, 2024-03-01 05:20, 13.5, mg/dL, completed, 2024-03-01 05:30
3 1300511: LDL, 2024-03-2 07:30, 110, mg/dL, pending
4 1300520: systole, 2024-03-4 04:40, 150, mm Hg, pending
5 1234567: LDL, 2020-01-12 23:04, 125, mg, Reviewed
6 1221381: systole, 2004-07-12 12:07, 127, mg, pending
7
```

Run `hi` ×

C:\Users\USER\PycharmProjects\hhh\ver

Menu:

- 1.Add new medical test.
- 2.Add new medical test record.
- 3.Update patient record.
- 4.Update medical tests.
- 5.Delete patient record.
- 6.Filter medical tests.
- 7.Generate textual summary reports.
- 8.Export medical records.
- 9.Import medical records.
- 10.Print all records.
- 11.Exit.

Choose an option: 5

Enter the Patient ID: 1221381

Enter the Test Name: systole

Enter the Test Date and Time (YYYY-MM-DD HH:MM:SS): 2004-07-12 12:07

Enter the Result Value: 127

Enter the Unit: mg

Enter the Status: pending

Enter the Result Date and Time (YYYY-MM-DD HH:MM:SS) or leave blank if not applicable:

Record deleted successfully.

- *Delete a test that does not exist*

```
hi.py  medicalRecord.txt  medicalTest.txt
1 1300500: RBC, 2024-01-01 14:10, 13.4, mg/dL, completed, 2024-01-01 15:30
2 1300500: RBC, 2024-03-01 05:20, 13.5, mg/dL, completed, 2024-03-01 05:30
3 1300511: LDL, 2024-03-2 07:30, 110, mg/dL, pending
4 1300520: systole, 2024-03-4 04:40, 150, mm Hg, pending
5 1234567: LDL, 2020-01-12 23:04, 125, mg, Reviewed
6

Run hi x
Menu:
1.Add new medical test.
2.Add new medical test record.
3.Update patient record.
4.Update medical tests.
5.Delete patient record.
6.Filter medical tests.
7.Generate textual summary reports.
8.Export medical records.
9.Import medical records.
10.Print all records.
11.Exit.
Choose an option: 5
Enter the Patient ID: 1234567
Enter the Test Name: hemo
Enter the Test Date and Time (YYYY-MM-DD HH:MM:SS): 2020-01-12 23:04
Enter the Result Value: 125
Enter the Unit: mg
Enter the Status: Reviewed
Enter the Result Date and Time (YYYY-MM-DD HH:MM:SS) or leave blank if not applicable:
No matching record found.
```

6. Filter Medical Tests

```
hi.py  medicalRecord.txt  medicalTest.txt

1 1300500: RBC, 2024-01-01 14:10, 13.4, mg/dL, completed, 2024-01-01 15:30
2 1300500: RBC, 2024-03-01 05:20, 13.5, mg/dL, completed, 2024-03-01 05:30
3 1300511: LDL, 2024-03-2 07:30, 110, mg/dL, pending
4 1300520: systole, 2024-03-4 04:40, 150, mm Hg, pending
5 1234567: LDL, 2020-01-12 23:04, 125, mg, Reviewed
6

Run hi x

3. Filter by Abnormal Tests
4. Filter by Date Range
5. Filter by Test Status
6. Filter by Turnaround Time Range
Apply Filter by Patient ID? (1/0): 1
Apply Filter by Test Name? (1/0): 1
Apply Filter by Abnormal Tests? (1/0): 0
Apply Filter by Date Range? (1/0): 0
Apply Filter by Test Status? (1/0): 0
Apply Filter by Turnaround Time Range? (1/0): 0
Enter Patient ID: 1234567
Enter Test Name: LDL
Patient ID: 1234567, Test Name: LDL, Date/Time: 2020-01-12 23:04, Result: 125, Unit: mg, Status: Reviewed, Result Date/Time: None
```

```
hi.py  medicalRecord.txt  medicalTest.txt

1 1300500: RBC, 2024-01-01 14:10, 13.4, mg/dL, completed, 2024-01-01 15:30
2 1300500: RBC, 2024-03-01 05:20, 13.5, mg/dL, completed, 2024-03-01 05:30
3 1300511: LDL, 2024-03-2 07:30, 110, mg/dL, pending
4 1300520: systole, 2024-03-4 04:40, 150, mm Hg, pending
5 1234567: LDL, 2020-01-12 23:04, 125, mg, Reviewed
6

Run hi x

Choose an option: 6

Filter Medical Tests - Options:
1. Filter by Patient ID
2. Filter by Test Name
3. Filter by Abnormal Tests
4. Filter by Date Range
5. Filter by Test Status
6. Filter by Turnaround Time Range
Apply Filter by Patient ID? (1/0): 0
Apply Filter by Test Name? (1/0): 1
Apply Filter by Abnormal Tests? (1/0): 0
Apply Filter by Date Range? (1/0): 1
Apply Filter by Test Status? (1/0): 0
Apply Filter by Turnaround Time Range? (1/0): 0
Enter Test Name: LDL
Enter start date (YYYY-MM-DD): 2019-04-02
Enter end date (YYYY-MM-DD): 2021-05-06
Patient ID: 1234567, Test Name: LDL, Date/Time: 2020-01-12 23:04, Result: 125, Unit: mg, Status: Reviewed, Result Date/Time: None
```

hi.py

medicalRecord.txt

medicalTest.txt

1	1300500: RBC, 2024-01-01 14:10, 13.4, mg/dL, completed, 2024-01-01 15:30	✓
2	1300500: RBC, 2024-03-01 05:20, 13.5, mg/dL, completed, 2024-03-01 05:30	
3	1300511: LDL, 2024-03-2 07:30, 110, mg/dL, pending	
4	1300520: systole, 2024-03-4 04:40, 150, mm Hg, pending	
5	1234567: LDL, 2020-01-12 23:04, 125, mg, Reviewed	
6		

Run

↑

↓

↕

↕

🗑

Filter Medical Tests - Options:
1. Filter by Patient ID
2. Filter by Test Name
3. Filter by Abnormal Tests
4. Filter by Date Range
5. Filter by Test Status
6. Filter by Turnaround Time Range
Apply Filter by Patient ID? (1/0): 0
Apply Filter by Test Name? (1/0): 0
Apply Filter by Abnormal Tests? (1/0): 0
Apply Filter by Date Range? (1/0): 0
Apply Filter by Test Status? (1/0): 1
Apply Filter by Turnaround Time Range? (1/0): 0
Enter Test Status: pending
Patient ID: 1300511, Test Name: LDL, Date/Time: 2024-03-2 07:30, Result: 110, Unit: mg/dL, Status: pending, Result Date/Time: None
Patient ID: 1300520, Test Name: systole, Date/Time: 2024-03-4 04:40, Result: 150, Unit: mm Hg, Status: pending, Result Date/Time: None

7. Generate Textual Summary Reports

hi.py

medicalRecord.txt

medicalTest.txt

1	1300500: RBC, 2024-01-01 14:10, 13.4, mg/dL, completed, 2024-01-01 15:30	✓
2	1300500: RBC, 2024-03-01 05:20, 13.5, mg/dL, completed, 2024-03-01 05:30	
3	1300511: LDL, 2024-03-2 07:30, 110, mg/dL, pending	
4	1300520: systole, 2024-03-4 04:40, 150, mm Hg, pending	
5	1234567: LDL, 2020-01-12 23:04, 125, mg, Reviewed	
6		

Run

↑

↓

↕

↕

🗑

3. Filter by Abnormal Tests
4. Filter by Date Range
5. Filter by Test Status
6. Filter by Turnaround Time Range
Apply Filter by Patient ID? (1/0): 0
Apply Filter by Test Name? (1/0): 0
Apply Filter by Abnormal Tests? (1/0): 0
Apply Filter by Date Range? (1/0): 0
Apply Filter by Test Status? (1/0): 1
Apply Filter by Turnaround Time Range? (1/0): 0
Enter Test Status: pending
Skipping record due to missing date information: {'test_name': 'LDL', 'test_date_time': '2024-03-2 07:30', 'result_value': '110', 'unit': 'mg/dL', 'status': 'pending', 'result_date_time': None}
Skipping record due to missing date information: {'test_name': 'systole', 'test_date_time': '2024-03-4 04:40', 'result_value': '150', 'unit': 'mm Hg', 'status': 'pending', 'result_date_time': None}

Test Value Statistics:
Minimum Value: 110.0
Maximum Value: 150.0
Average Value: 130.00
No valid turnaround times found.

hi.py medicalRecord.txt x medicalTest.txt

1

1300500: RBC, 2024-01-01 14:10, 13.4, mg/dL, completed, 2024-01-

2

1300500: RBC, 2024-03-01 05:20, 13.5, mg/dL, completed, 2024-03-

3

1300511: LDL, 2024-03-2 07:30, 110, mg/dL, pending

4

1300520: systole, 2024-03-4 04:40, 150, mm Hg, pending

5

1234567: LDL, 2020-01-12 23:04, 125, mg, Reviewed

6

Run hi x

↺

⏏

:

↑

↓

⇌

⇌

🖨

🗑

6. Filter by Turnaround Time Range

Apply Filter by Patient ID? (1/0): 1

Apply Filter by Test Name? (1/0): 1

Apply Filter by Abnormal Tests? (1/0): 0

Apply Filter by Date Range? (1/0): 0

Apply Filter by Test Status? (1/0): 0

Apply Filter by Turnaround Time Range? (1/0): 0

Enter Patient ID: 1300500

Enter Test Name: RBC

Test Value Statistics:

Minimum Value: 13.4

Maximum Value: 13.5

Average Value: 13.45

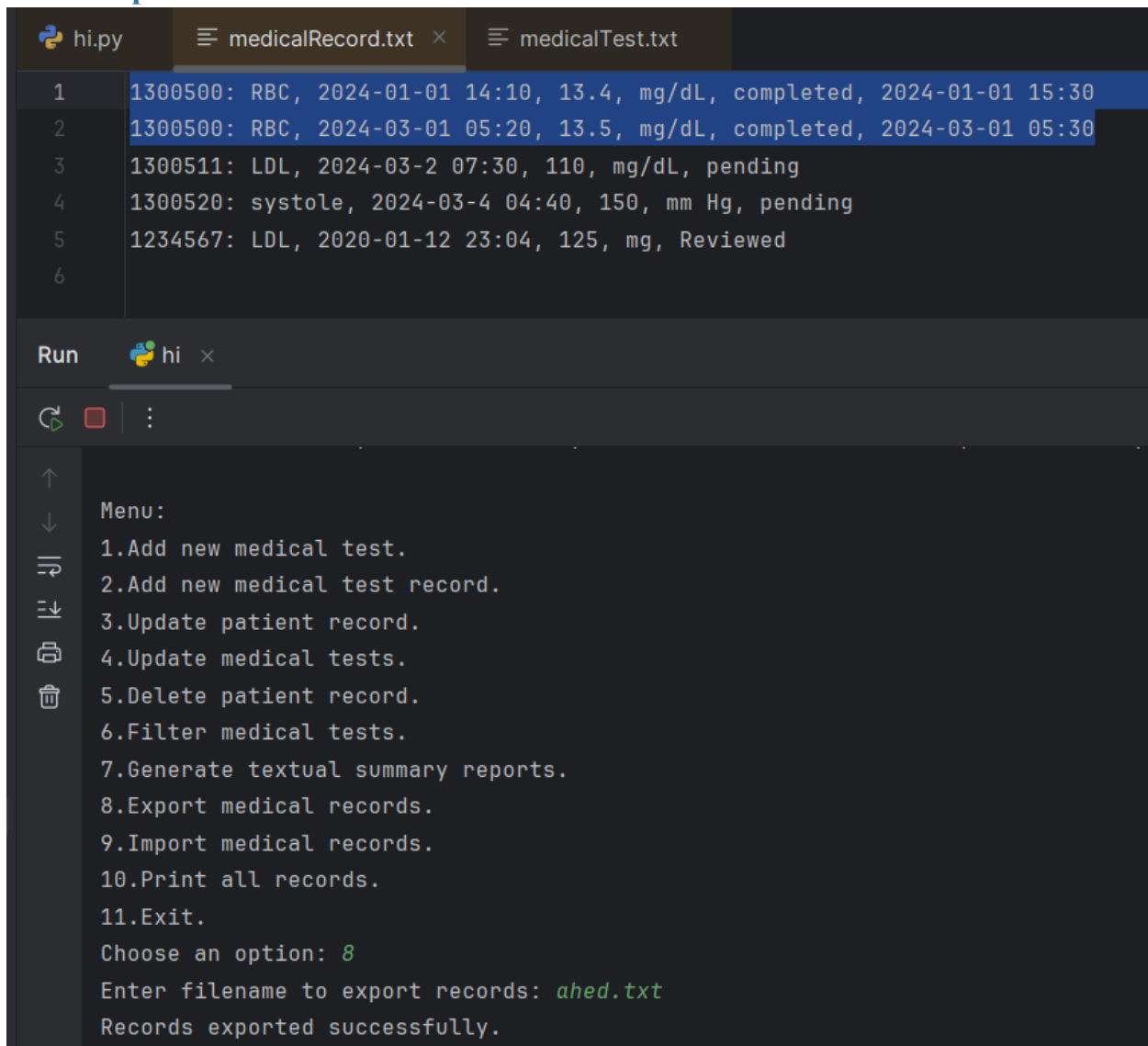
Turnaround Time Statistics (in minutes):

Minimum Turnaround Time: 10.0

Maximum Turnaround Time: 80.0

Average Turnaround Time: 45.00

8. Export Medical Records



The screenshot shows a Python IDE with two tabs: `hi.py` and `medicalRecord.txt`. The `medicalRecord.txt` tab is active, displaying a list of medical records. Below the list is a `Run` button and a `hi` icon. To the left of the main text area is a vertical toolbar with icons for undo, redo, and other editing functions. The main text area contains a menu with 11 options, where option 8, "Export medical records.", is selected. Below the menu, the user is prompted to enter a filename, and the output shows that the records were exported successfully.

```
1 1300500: RBC, 2024-01-01 14:10, 13.4, mg/dL, completed, 2024-01-01 15:30
2 1300500: RBC, 2024-03-01 05:20, 13.5, mg/dL, completed, 2024-03-01 05:30
3 1300511: LDL, 2024-03-2 07:30, 110, mg/dL, pending
4 1300520: systole, 2024-03-4 04:40, 150, mm Hg, pending
5 1234567: LDL, 2020-01-12 23:04, 125, mg, Reviewed
6
```

Run `hi`

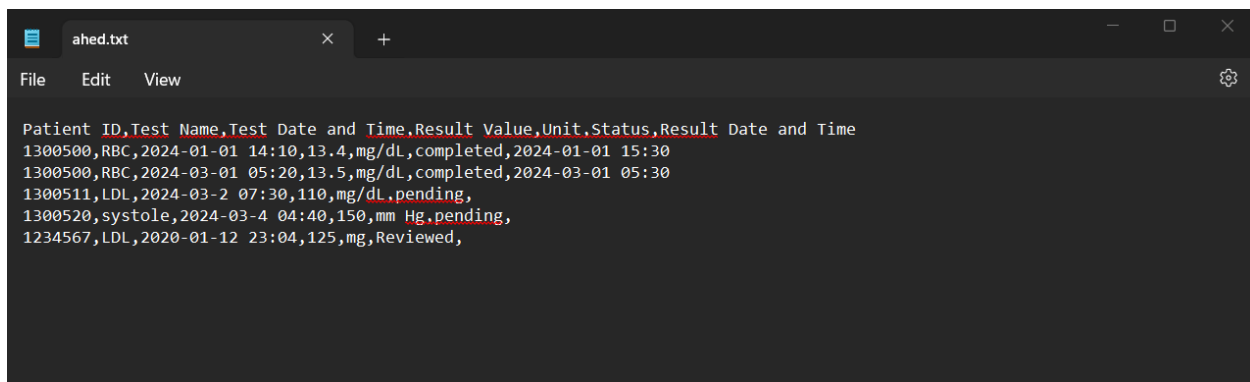
Menu:

- 1.Add new medical test.
- 2.Add new medical test record.
- 3.Update patient record.
- 4.Update medical tests.
- 5.Delete patient record.
- 6.Filter medical tests.
- 7.Generate textual summary reports.
- 8.Export medical records.
- 9.Import medical records.
- 10.Print all records.
- 11.Exit.

Choose an option: 8

Enter filename to export records: `ahed.txt`

Records exported successfully.



The screenshot shows a text editor window with a single tab named `ahed.txt`. The editor has a menu bar with `File`, `Edit`, and `View` options. The text area contains the exported medical records, which are formatted as a table with columns: Patient ID, Test Name, Test Date and Time, Result Value, Unit, Status, and Result Date and Time. The records are listed in a single line, separated by commas.

```
ahed.txt
File Edit View
Patient ID,Test Name,Test Date and Time,Result Value,Unit,Status,Result Date and Time
1300500,RBC,2024-01-01 14:10,13.4,mg/dL,completed,2024-01-01 15:30
1300500,RBC,2024-03-01 05:20,13.5,mg/dL,completed,2024-03-01 05:30
1300511,LDL,2024-03-2 07:30,110,mg/dL,pending,
1300520,systole,2024-03-4 04:40,150,mm Hg,pending,
1234567,LDL,2020-01-12 23:04,125,mg,Reviewed,
```

9. Import Medical Records

```
h.txt
File Edit View
2222222, hemo, 2022-03 02:03, 233, kilo, complete,
1111111, systole, 2004-07:12 12:07, 127, mg, pending,
```

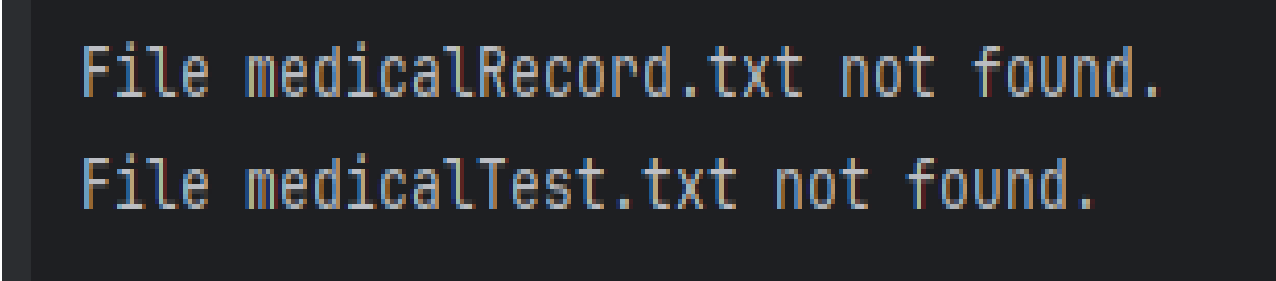
```
hi.py x medicalRecord.txt x medicalTest.txt
1 1300500: RBC, 2024-01-01 14:10, 13.4, mg/dL, completed, 2024-01-01 15:30
2 1300500: RBC, 2024-03-01 05:20, 13.5, mg/dL, completed, 2024-03-01 05:30
3 1300511: LDL, 2024-03-2 07:30, 110, mg/dL, pending
4 1300520: systole, 2024-03-4 04:40, 150, mm Hg, pending
5 1234567: LDL, 2020-01-12 23:04, 125, mg, Reviewed
6 2222222: hemo, 2022-03 02:03, 233, kilo, complete
7 1111111: systole, 2004-07:12 12:07, 127, mg, pending
8

Run hi x
5.Delete patient record.
6.Filter medical tests.
7.Generate textual summary reports.
8.Export medical records.
9.Import medical records.
10.Print all records.
11.Exit.
Choose an option: 9
Enter filename to import records: h.txt
Records imported successfully.
```

10. Error Handling

- *Invalid File Name*

The program handles it when files don't exist by giving the user a message indicating the error as seen in the figure below:



```
File medicalRecord.txt not found.  
File medicalTest.txt not found.
```