

Debugging dan Testing

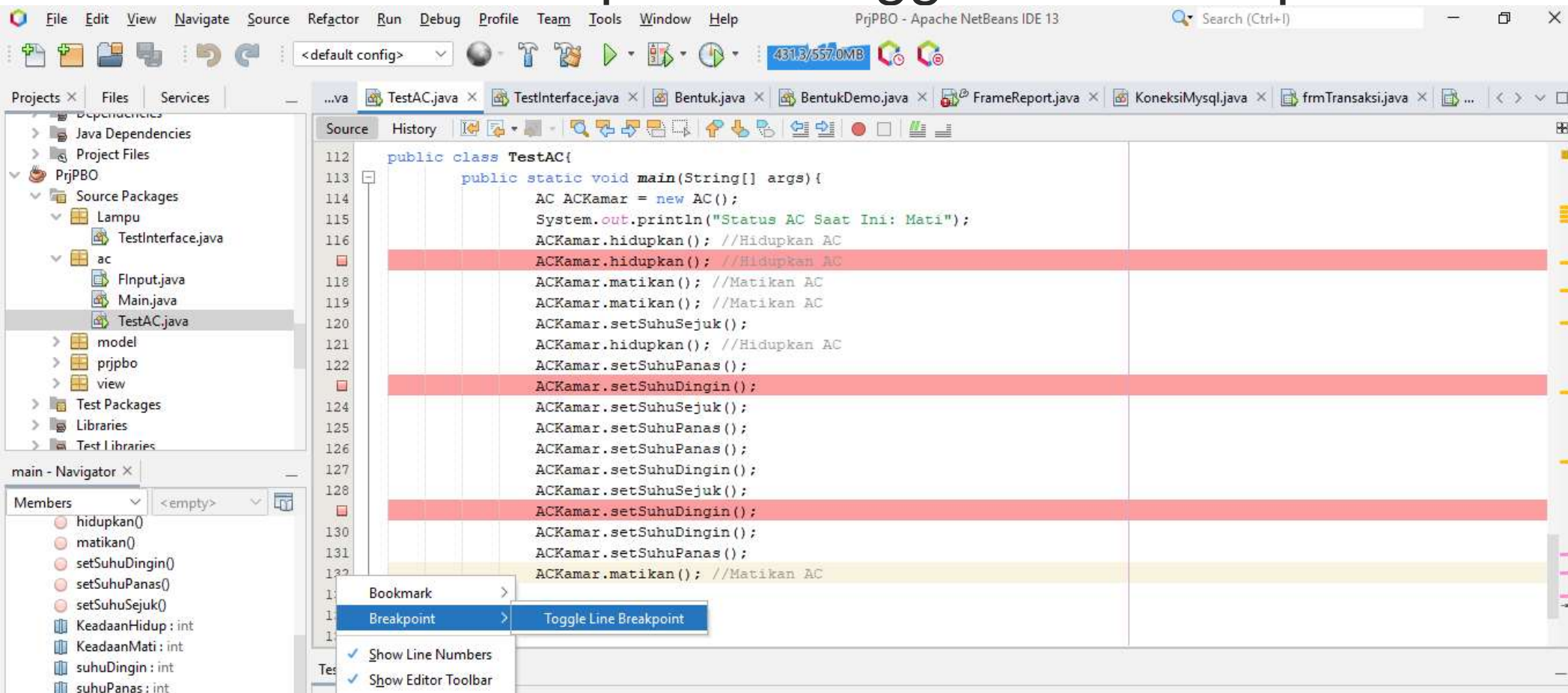


Debugging

- Proses memperbaiki bug dalam perangkat lunak
- Proses mengidentifikasi, menganalisis dan menghapus kesalahan.
- Kegiatan ini dimulai setelah perangkat lunak gagal dijalankan dengan benar dan diakhiri dengan pemecahan masalah dan berhasil menguji perangkat lunak

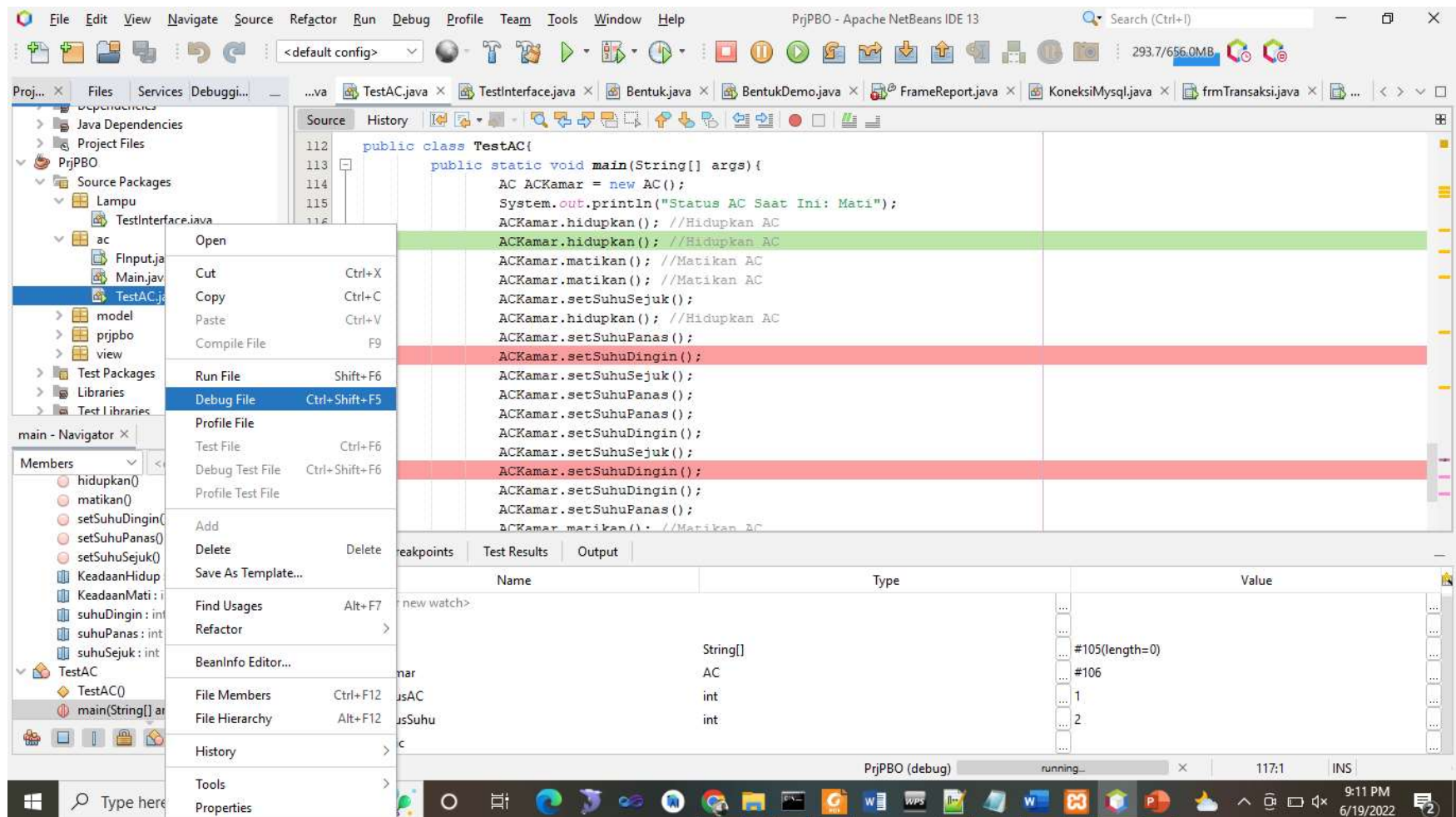
Setting Breakpoint

- Double klik baris atau sourcode yang akan diberi breakpoint atau
- klik kanan breakpoint -> toggle line breakpoint



Memulai Debug

- memulai debug klik kanan di file kemudian debug file



Hasil Debug

The screenshot displays the Apache NetBeans IDE 13 interface during a debug session. The top menu bar includes File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, and Help. The toolbar contains various icons for file operations, running, and debugging. The Projects window on the left shows the 'Common-Cleaner' running and the 'main' method at line 117 of TestAC.java. The Source window displays the following code:

```
112 public class TestAC{
113     public static void main(String[] args){
114         AC ACKamar = new AC();
115         System.out.println("Status AC Saat Ini: Mati");
116         ACKamar.hidupkan(); //Hidupkan AC
117         ACKamar.hidupkan(); //Hidupkan AC
118         ACKamar.matikan(); //Matikan AC
119         ACKamar.matikan(); //Matikan AC
120         ACKamar.setSuhuSejuk();
121         ACKamar.hidupkan(); //Hidupkan AC
122         ACKamar.setSuhuPanas();
123         ACKamar.setSuhuDingin();
```

The Variables window shows the state of the program at the breakpoint. The variables are listed in the following table:

| Name | Type | Value |
|-------------------|----------|----------------|
| <Enter new watch> | | |
| Static | | |
| args | String[] | #105(length=0) |
| ACKamar | AC | #106 |
| statusAC | int | 1 |
| statusSuhu | int | 2 |
| Static | | |
| KeadaanHidup | int | 1 |
| KeadaanMati | int | 0 |
| suhuDingin | int | 0 |
| suhuSejuk | int | 1 |
| suhuPanas | int | 2 |

The bottom status bar shows the current project is 'PrjPBO (debug)' and the status is 'running...'. The line number is 117:1 and the cursor is at the INS position.

Kontrol Key Debug

| Key | Kegunaan key pada saat debug |
|----------|--|
| Shift+F5 | Mengakhiri debug |
| F5 | Melanjutkan debug, sampai menemui breakpoint selanjutnya |
| F8 | Debug perbaris/ change row |
| F7 | Debug eksekusi method, dan masuk kedalam method tersebut |
| Shift+F7 | Keluar dari method, dan kembali pada pemanggil eksekusi method tersebut. |

Kontrol Key Debug

- F7

The screenshot shows the Apache NetBeans IDE 13 interface. The 'Source' window displays the code for `BufferedWriter.java`, which is currently paused at a breakpoint. The code is as follows:

```
while (b < t) {  
    int d = min(nChars - nextChar, t - b);  
    s.getChars(b, b + d, cb, nextChar);  
    b += d;  
    nextChar += d;  
    if (nextChar >= nChars)  
        flushBuffer();  
}
```

The 'Variables' window shows the state of the `BufferedWriter` object and its fields:

| Name | Type | Value |
|-------------------|-----------------------------|---|
| <code>this</code> | <code>BufferedWriter</code> | <code>#132</code> |
| <code>s</code> | <code>String</code> | <code>"Hidupkan AC! --> AC Sudah Hidup Kok"</code> |
| <code>off</code> | <code>int</code> | <code>0</code> |
| <code>len</code> | <code>int</code> | <code>35</code> |
| <code>b</code> | <code>int</code> | <code>35</code> |
| <code>t</code> | <code>int</code> | <code>35</code> |

The 'Output' window shows the program's output:

```
write - Navigator x  
Members <empty>  
BufferedWriter :: Writer  
  BufferedWriter(Writer out)  
  BufferedWriter(Writer out, int sz)  
  close() ↑ Writer  
  ensureOpen()  
  flush() ↑ Writer  
  flushBuffer()  
  min(int a, int b) : int  
  newLine()  
  write(int c) ↑ Writer  
  write(char[] cbuf, int off, int len) ↑ W  
  write(String s, int off, int len) ↑ Write
```

The status bar at the bottom indicates the program is running in the 'PrjPBO (debug)' window, with the current line being 227:1 in the 'INS Unix (LF)' file.

Kontrol Key Debug

- F8

Apache NetBeans IDE 13

Search (Ctrl+I)

410.4/692.0MB

Projects Files Services Deb... x

...va TestAC.java x TestInterface.java x Bentuk.java x BentukDemo.java x FrameReport.java x KoneksiMysql.java x frmTransaksi.java x ...

Source History

```
113 public static void main(String[] args){
114     AC ACKamar = new AC();
115     System.out.println("Status AC Saat Ini: Mati");
116     ACKamar.hidupkan(); //Hidupkan AC
117     ACKamar.hidupkan(); //Hidupkan AC
118     ACKamar.matikan(); //Matikan AC
119     ACKamar.matikan(); //Matikan AC
120     ACKamar.setSuhuSejuk();
121     ACKamar.hidupkan(); //Hidupkan AC
122     ACKamar.setSuhuPanas();
123     ACKamar.setSuhuDingin();
124     ACKamar.setSuhuSejuk();
```

Variables Breakpoints Test Results Output

| Name | Type | Value |
|--------------------|------|-------|
| <Enter new watch> | | |
| Static | | |
| args | | |
| String[] | | |
| ... #105(length=0) | | |
| ACKamar | | |
| AC | | |
| statusAC | | |
| int | | |
| statusSuhu | | |
| int | | |
| Static | | |
| KeadaanHidup | | |
| int | | |
| KeadaanMati | | |
| int | | |
| suhuDingin | | |
| int | | |
| suhuSejuk | | |
| int | | |
| suhuPanas | | |
| int | | |

main - Navigator x

Members <empty>

- hidupkan()
- matikan()
- setSuhuDingin()
- setSuhuPanas()
- setSuhuSejuk()
- KeadaanHidup : int
- KeadaanMati : int
- suhuDingin : int
- suhuPanas : int
- suhuSejuk : int
- TestAC
- TestAC()
- main(String[] args)

PrjPBO (debug) running_ 120:1 INS

Testing / Pengujian

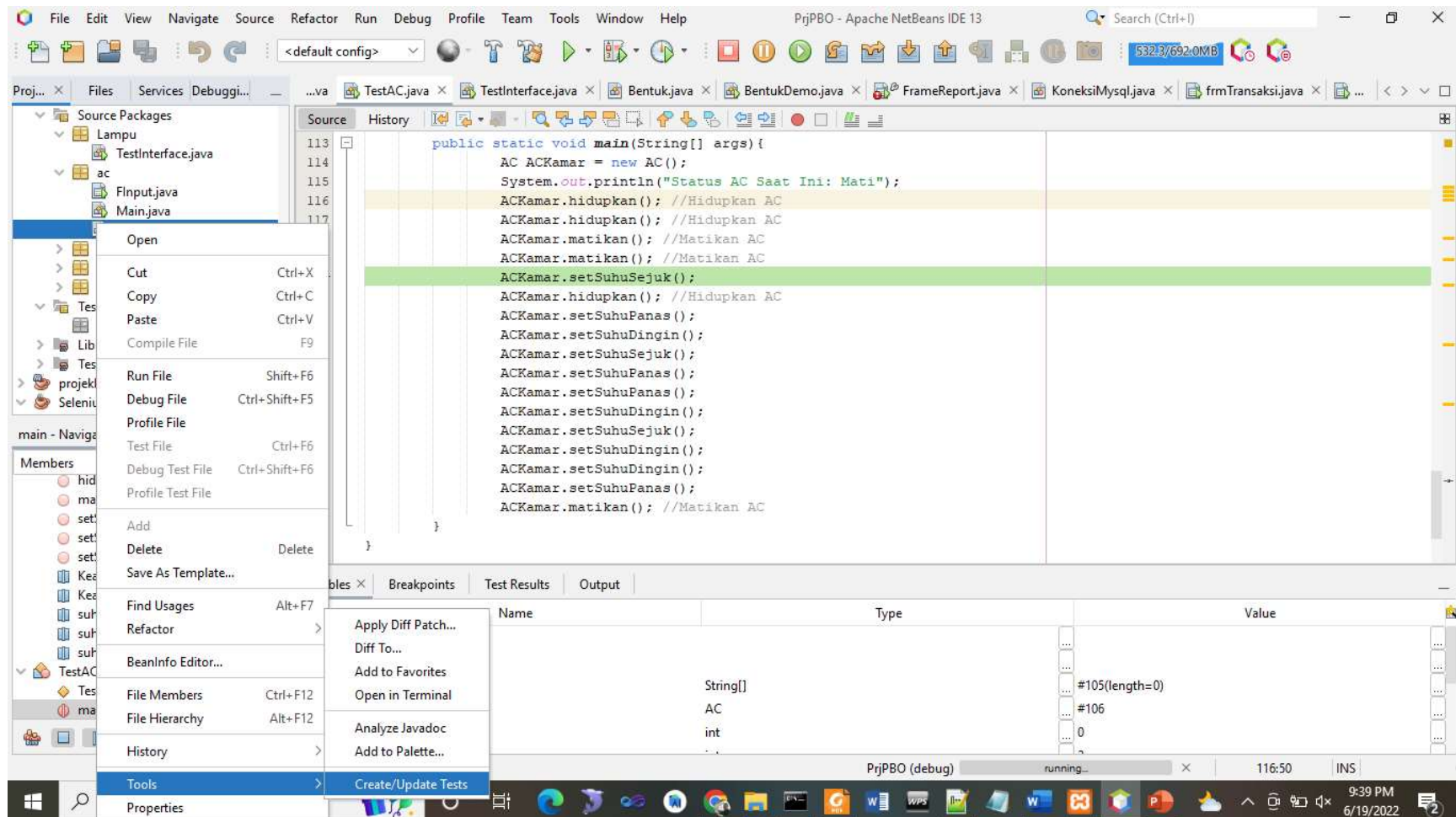
- Proses memverifikasi dan memvalidasi bahwa perangkat lunak atau aplikasi bebas bug,
- Memenuhi persyaratan teknis sesuai dengan desain dan pengembangannya serta
- Memenuhi persyaratan pengguna secara efektif dan efisien dengan menangani semua kasus yang ada.

Unit Testing

- Jenis *software testing* yang dilakukan untuk menguji suatu bagian atau komponen *software*
- Unit yang dimaksud bisa berupa kode, fungsi, metode, prosedur, modul, atau objek tersendiri
- *Unit testing* termasuk dalam tahapan *software development*. Biasanya, pengujian unit ini dilakukan sebelum *system integration testing*.

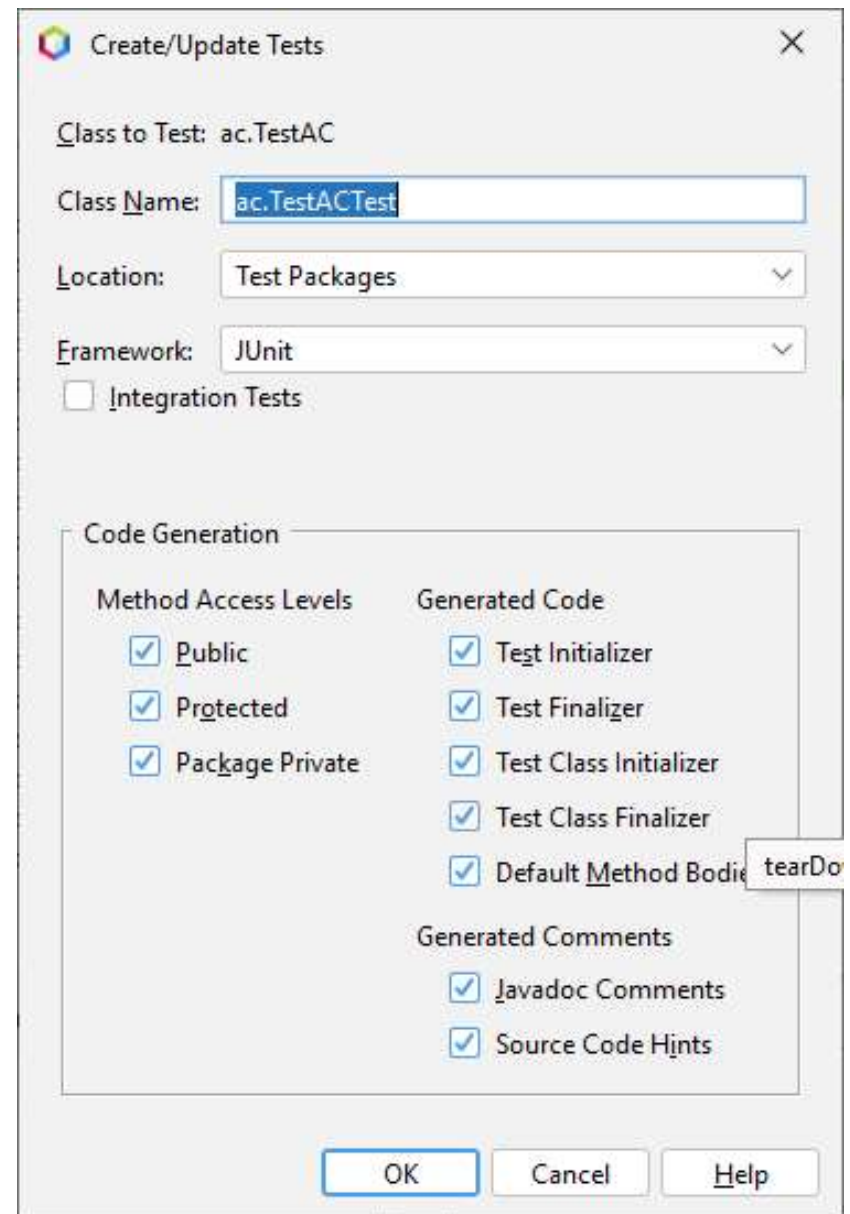
JUnit Netbeans

- Buat Project baru atau gunakan project yang sudah ada, pilih file yang akan di test
- Klik kanan pilih tools, create/update tests



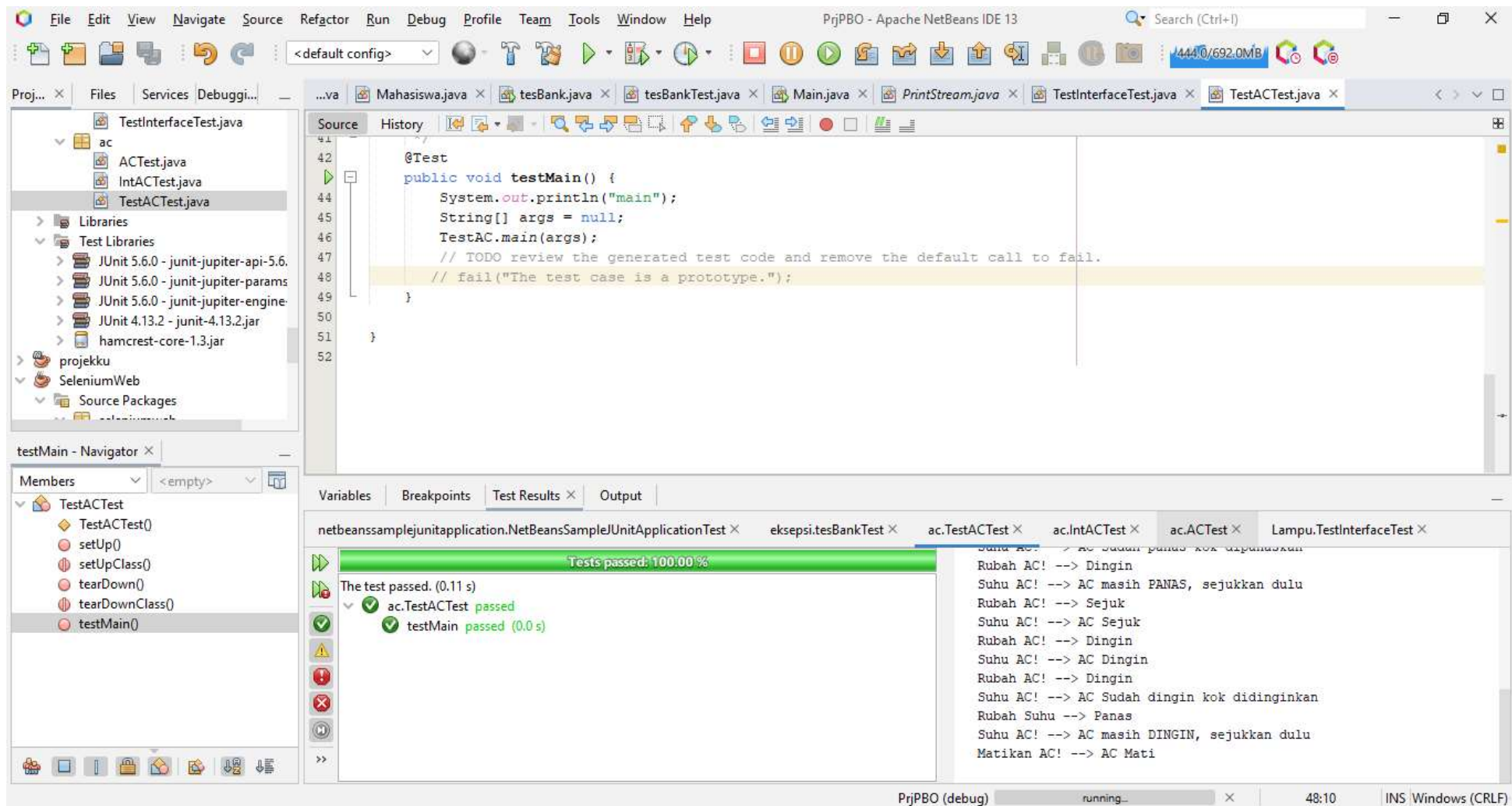
JUnit Netbeans

- Create/update tests
- Klik Ok



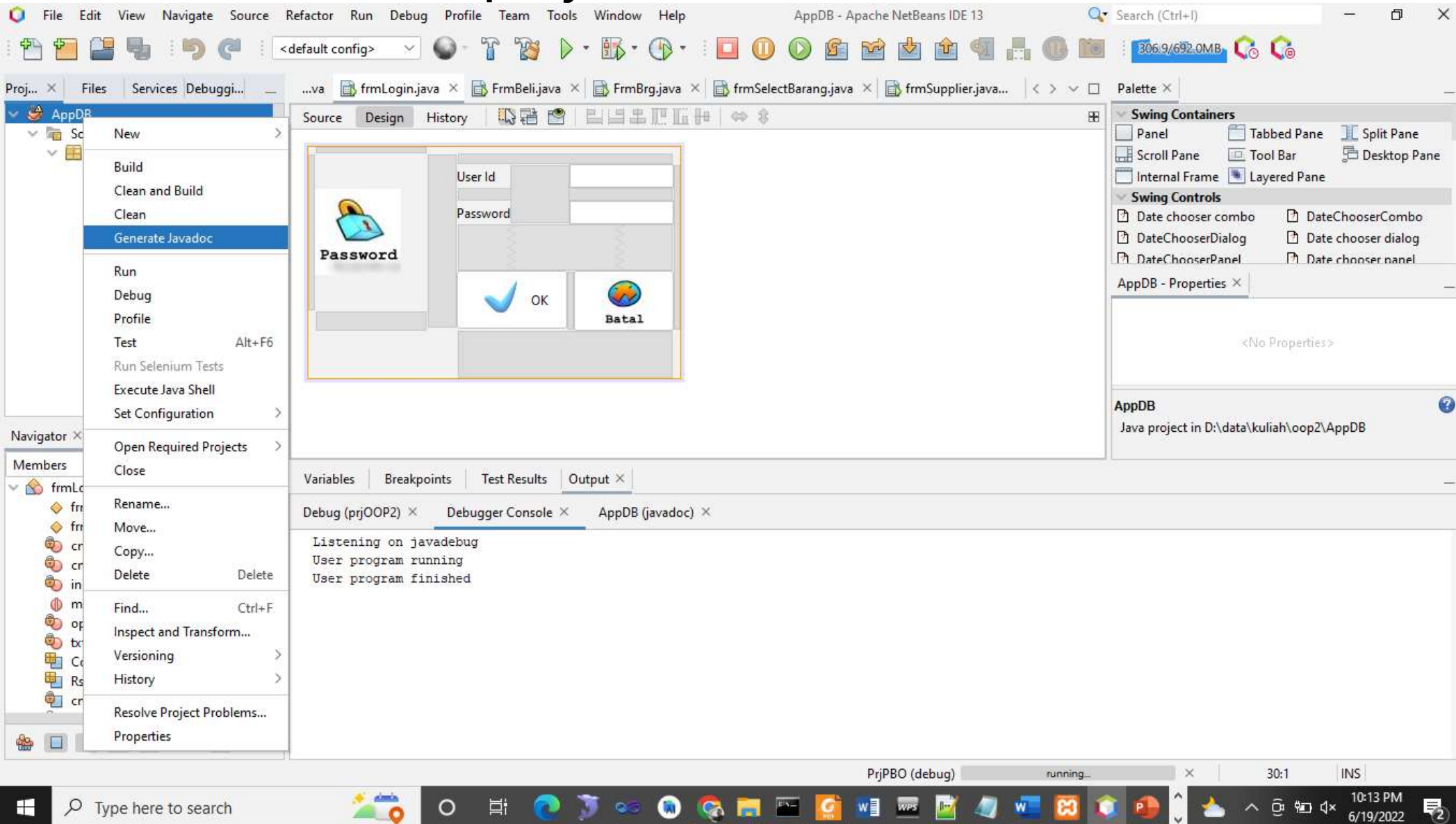
Eksekusi File Testing

- Jalankan hasil file testing, pastikan library JUnit & hamcrest-core-1.3.jar tersedia



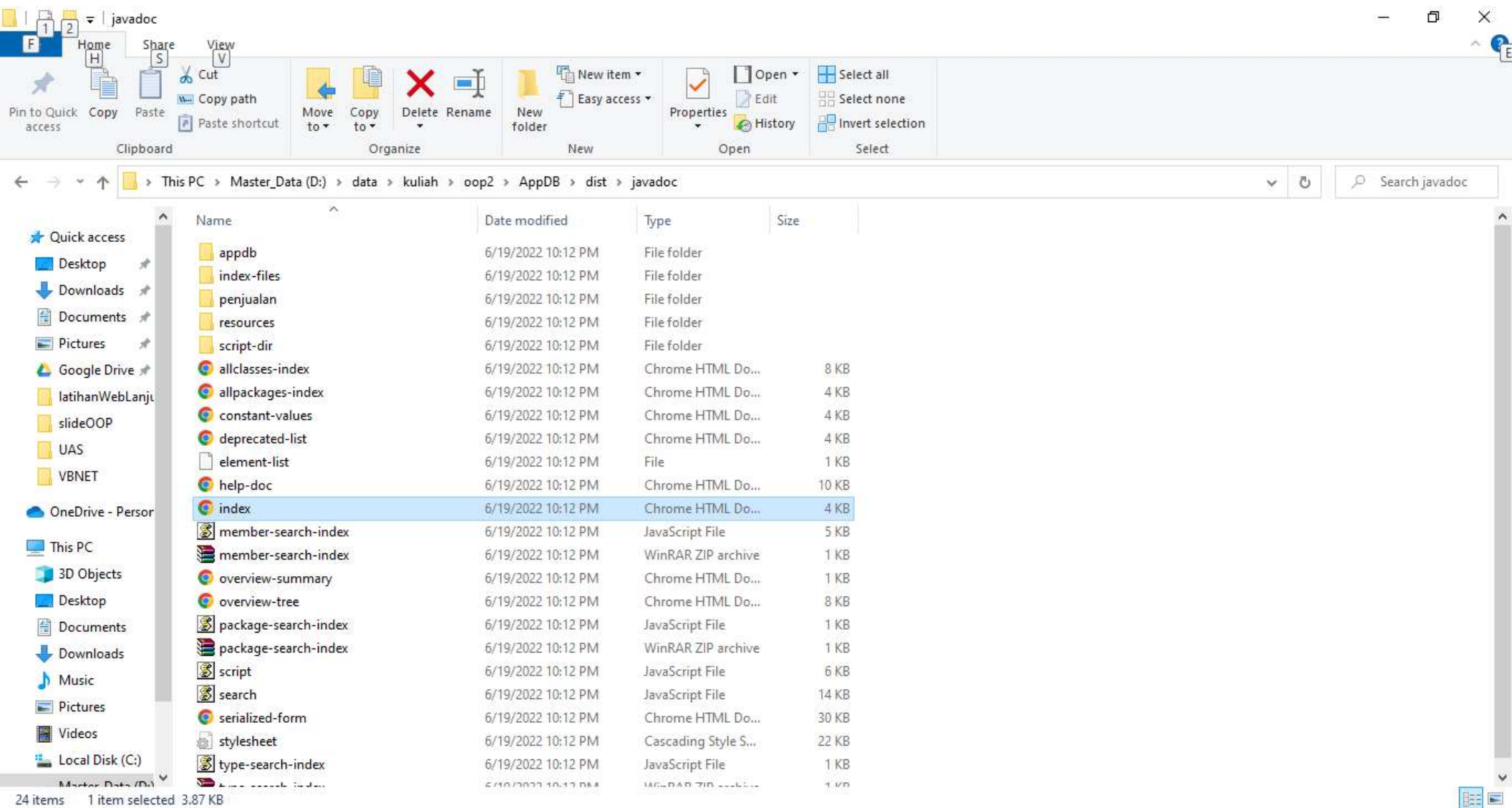
JavaDoc

- Klik kanan di project → Generate JavaDoc



Hasil JavaDoc

- Buka folder project → dist → javadoc



Buka di Browser

- File Index.html

The screenshot shows a web browser window displaying the Javadoc index.html file for AppDB. The browser's address bar shows the file path: D:/data/kuliah/oop2/AppDB/dist/javadoc/index.html. The page has a navigation bar with the following tabs: OVERVIEW, PACKAGE, CLASS, USE, TREE, DEPRECATED, INDEX, and HELP. A search bar is located on the right side of the navigation bar. The main content area shows a table with the following data:

| Package | Description |
|-----------|-------------|
| appdb | |
| penjualan | |

Class

- Daftar class di package

The screenshot shows a web browser window displaying the Javadoc package summary for 'appdb'. The browser's address bar shows the URL 'D:/data/kuliah/oop2/AppDB/dist/javadoc/appdb/package-summary.html'. The browser's taskbar at the bottom shows several open applications, including 'Semuanya Ajib', 'Semunyaajib', '4: Data Augmentati...', 'RecyclerView CRUD...', 'Keyword Tool (FREE...', 'Dashboard < CV. D...', 'Belajar Membuat A...', and 'Ajib Susanto - Peng...'. The Javadoc page has a navigation bar with tabs: 'OVERVIEW', 'PACKAGE' (selected), 'CLASS', 'USE', 'TREE', 'DEPRECATED', 'INDEX', and 'HELP'. A search bar is located on the right side of the navigation bar. The main content area is titled 'Package appdb' and contains a table with two columns: 'Class' and 'Description'. The table lists the following classes: 'AppDB', 'DBAccess', 'frmBackup', 'frmBarang', 'FrmBeli', 'FrmBrg', 'frmJual', 'frmKonsumen', 'frmLapKonsumen', 'frmLaporanBarang', 'frmLogin', 'FrmMenu', 'frmSelectBarang', and 'frmSupplier'.

| Class | Description |
|------------------|-------------|
| AppDB | |
| DBAccess | |
| frmBackup | |
| frmBarang | |
| FrmBeli | |
| FrmBrg | |
| frmJual | |
| frmKonsumen | |
| frmLapKonsumen | |
| frmLaporanBarang | |
| frmLogin | |
| FrmMenu | |
| frmSelectBarang | |
| frmSupplier | |



ajibsusanto@gmail.com

ajib.susanto@dsn.dinus.ac.id

@ajibsusanto / 085876247118



Referensi

- Object First With Java, Fifth edition, David J. Barnes & Michael Kölling, Prentice Hall / Pearson Education, 2012.
- The Java™ Tutorial,
<http://docs.oracle.com/javase/tutorial/java/nutsandbolts/>, Oracle, 1995–2022.
- Java SE Tutorial,
<https://docs.oracle.com/javase/tutorial/>, Oracle, 2022.
- Java Platform, SE Documentation,
<https://docs.oracle.com/en/java/javase/index.html>
- Testing & Debugging, GeeksforGeeks,
<https://www.geeksforgeeks.org/differences-between-testing-and-debugging/>, 2022 .
- Unit Testing, Guru99, <https://www.guru99.com/unit-testing-guide.html>, 2022
- JUnit NetBeans Example, Alvin Reyes,
https://examples.javacodegeeks.com/core-java/junit/junit-netbeans-example/#google_vignette, 2022.

